



THÈSE

Pour l'obtention du grade de

DOCTEUR DE L'UNIVERSITE DE POITIERS

Faculté des Sciences Fondamentales et Appliquées

Diplôme National - Arrêté du 7 août 2006

ÉCOLE DOCTORALE SCIENCES POUR L'INGENIEUR & AERONAUTIQUE

DOMAINE DE RECHERCHE : MILIEUX DENSES, MATERIAUX ET COMPOSANTS

Présentée par

Prénom Nom

Titre de la thèse

Directeur de thèse : M. xxxxxx

Co-direction : M. xxxxxx, Mme xxxxxx

Soutenue le xx xxxxx xxxx
Devant la Commission d'Examen

JURY

M. xx xxxxx	Professeur d'Université	Rapporteur
M. xx xxxxx	Directeur de Recherche au CNRS	Rapporteur
M. xx xxxxx	Directeur de Recherche au CNRS	Examineur
M. xx xxxxx	Professeur d'Université	Examineur
M. xx xxxxx	Chargé de Recherche au CNRS	Examineur

Remerciement

Nous adressons nos remerciements tout d'abord à :

- Tout le cadre de l'Académie Militaire
- Tous les professeurs de l'Académie Militaire
- Tous mes collègues

pour leur soutien et leurs encouragements durant toute l'année qu'on a passée au sein de cet établissement.

Nous remercions tout particulièrement M.Sidhom Walid qui nous a honoré par son encadrement, son soutien, sa disponibilité et surtout sa patience tout le long de la période du travail.

Nos mains se lèvent également vers nos amis de la classe Génie Informatique 21, pour leurs critiques constructives à l'égard de notre petite mémoire et pour leur encouragement durant les deux années qu'on a passées ensemble.

Toute notre gratitude et nos salutations les plus chaleureux à nos familles, spécialement nos parents qui n'ont pas cessé de nous pousser depuis la naissance.

Nous dédions enfin ce travail à toutes les personnes qui nous ont gracieusement accorder de leur temps, pour atteindre nos modestes buts et parvenir à ce que nous le somme maintenant...

Table des matières

Table des figures	vii
Liste des tableaux	ix
Introduction générale	1
I Introduction à la sécurité	3
1 La sécurité informatique	3
1.1 Domaines de la sécurité informatique	3
1.2 Objectifs généraux de la sécurité informatique	3
1.2.1 L'intégrité	3
1.2.2 La confidentialité	3
1.2.3 La disponibilité	4
1.2.4 La non-répudiation	4
1.2.5 L'authentification	4
2 Vulnérabilité	4
2.1 Définition	4
2.2 Causes	4
2.3 Identification et correction des vulnérabilités	5
3 Les types d'attaques	5
3.1 Anatomie d'une attaque	5
3.2 Les attaques réseaux	6
3.2.1 Les techniques du scan	6

3.2.2	IP Spoofing	6
3.2.3	ARP Spoofing (ou ARP Redirect)	7
3.3	Les attaques applicatives	7
3.3.1	Les problèmes de configuration	8
3.3.2	Les buffer overflows	8
3.3.3	Les scripts	8
3.3.4	Les injections SQL	9
3.4	Le Dénî de service	9
3.4.1	SYN Flooding	9
3.4.2	UDP Flooding	9
3.4.3	Packet Fragment	9
3.4.4	Smurfing	9
4	Conclusion	10
II Analyse réseau		11
1	Définition	11
1.1	Qu'est ce qu'un Analyseur réseau ?	11
1.2	Utilisation du sniffer	11
1.3	Technique de fonctionnement	12
2	Outils d'analyse réseau	12
2.1	Wireshark	12
2.2	Tcpdump	12
2.3	Ettercap	13
2.4	Dsniff	13
2.5	SoftPerfect Analyser	13
2.6	Kismet	14
3	Tableau comparatif des outils	14
III Tests d'Intrusion		17
1	Définition	17
1.1	Qu'est ce qu'un test d'intrusion ?	17
1.2	Stratégies des tests	17
1.3	Types des tests	18
1.4	Leurs limites	18
1.5	La démarche utilisée dans les tests d'intrusion	18

2	L'audit de vulnérabilité	20
3	Outils liés aux tests d'intrusion	20
3.1	Divers outils	21
3.1.1	Etape de collecte d'informations publiques (A.1.)	21
3.1.2	Etape de cartographie du réseau cible (A.2.)	21
3.1.3	Etape d'identification des vulnérabilités (A.3.)	21
3.1.4	Etape d'exécution des scénarii (B.2.)	22
3.2	Outils d'audit	22
3.2.1	Internet Scanner	22
3.2.2	SATAN, SAINT, SARA	22
3.2.3	Retina	22
3.2.4	Nessus, NeWT	22

IV Réalisation **25**

1	Introduction	25
2	Environnement de travail	25
2.1	Environnement matériel	25
2.2	Environnement logiciel	25
2.3	Back Track	26
2.3.1	L'outil	26
2.3.2	Installation	26
2.3.3	Tests qu'on peut réaliser	26
2.3.4	Evolutions	28
3	Les tests réalisés	28
3.1	Topologie	28
4	Les tests réalisés	30
4.1	Test 1 :	30
4.2	Test 2 : Sniffer le trafic réseau	30
4.2.1	Test 2-A : En utilisant Wireshark	30
4.2.2	Test 2-B : En utilisant Ettercap	31
4.3	Test 3 : Crackage d'un clé WEP dans un réseau sans fil (Wi-Fi)	32
4.3.1	Paramétrage	33
4.3.2	Exploitation	34
5	Conclusion	37

Conclusion générale et perspectives	39
Annexe A : Glossaire	41
Bibliographie	45

Table des figures

I.1	Typologie des faiblesses de sécurité.	6
I.2	L'attaque IP spoofing.	7
I.3	L'attaque ARP spoofing.	8
III.1	La démarche utilisée dans les tests d'intrusion.	19
III.2	Exemples d'outils utilisés lors d'une intrusion.	20
IV.1	Le réseau ETHERNET utilisé pour les tests	29
IV.2	Capture du trafic entre le client FTP et le serveur FTP avec Wireshark	31
IV.3	Capture du trafic entre le client FTP et le serveur FTP avec Ettercap	32
IV.4	le processus de crackage du clé Wi-Fi	33
IV.5	Le réseau du test 3	34
IV.6	le processus de crackage du clé Wi-Fi	35
IV.7	Résultat obtenu par Aircrack-ng	37

Liste des tableaux

II.1	Tableau comparatif de quelques outils d'analyse réseau	15
IV.1	Les configurations des machines virtuelles	30

Convention

Les différentes typographies utilisées dans ce document sont les suivantes :

- une typographie ordinaire pour le texte,
- **Une mise en gras** pour les termes figurant dans le glossaire (Annexe A).

Introduction générale

De nos jours, l'informatique est devenue une pièce maîtresse dans l'acquisition et le traitement de l'information sous toutes ses formes. De ce fait les systèmes informatiques sont omniprésents. Ainsi, dans notre société ce domaine attire de plus en plus des personnes malintentionnées qui tentent de compromettre leur intégrité, confidentialité ou disponibilité. En effet, les entreprises et les particuliers se voient donc confrontés de façon quotidienne à des virus, des attaques de tout type de tentatives d'intrusions. La sécurité est plus que jamais une problématique d'actualité et nous pouvons facilement le constater en parcourant les journaux de la presse spécialisée. Un moyen rapide de connaître l'étendue de la fragilité d'un environnement, vis à vis des attaques diverses et variées, est d'effectuer des tests d'intrusions qui permettent d'avoir une liste de failles de vulnérabilités potentielles. Dans ce cadre se situe notre projet, intitulé « Etude d'environnement d'analyse réseau et test d'intrusion », qui se propose de présenter les tests d'intrusion dans les réseaux. Ce document est le rapport des travaux entrepris durant ce projet de fin d'année, réalisé au sein de l'Académie Militaire. L'introduction est suivie d'un premier chapitre consacré à l'introduction de la sécurité et ses objectifs et aux divers types d'attaques. Par la suite, le second chapitre abordera l'analyse réseau et ses différents outils. Le troisième chapitre présentera les tests d'intrusion. Lors du quatrième et dernier chapitre, nous présenterons les différentes étapes de réalisation de nos tests, avant d'aborder la conclusion générale.

Chapitre I

Introduction à la sécurité

1 La sécurité informatique

1.1 Domaines de la sécurité informatique

La sécurité informatique c'est l'ensemble des moyens mis en œuvre pour minimiser la vulnérabilité d'un système contre des menaces accidentelles ou in accidentelles. Elle consiste avant tout à connaître et prévenir. Connaître les dangers auxquels il convient de parer, et connaître les moyens de se prémunir contre eux. La sécurité exige de se montrer systématique, mais elle implique aussi un effort de coordination, de technologie, d'information, de discussion, de comparaison et de contrôle.

1.2 Objectifs généraux de la sécurité informatique

La sécurité informatique[1], 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 :

1.2.1 L'intégrité

la vérification de l'intégrité des données consiste à déterminer si les données n'ont pas été modifiées durant la communication (de manière fortuite ou intentionnelle).

1.2.2 La confidentialité

La confidentialité consiste à rendre l'information inintelligible(indéchiffrable) à d'autres personnes que les seuls acteurs de la transaction.

1.2.3 La disponibilité

Son objectif est de garantir l'accès à un service ou à des ressources.

1.2.4 La non-répudiation

La non-répudiation de l'information est la garantie qu'aucun des correspondants ne pourra nier la transaction.

1.2.5 L'authentification

L'authentification consiste à assurer l'identité d'un utilisateur, c'est-à-dire de 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.

2 Vulnérabilité

2.1 Définition

Dans le domaine de la sécurité informatique[2], une vulnérabilité est une faiblesse dans un système informatique permettant à un attaquant de porter atteinte à l'intégrité de ce système, c'est-à-dire à son fonctionnement normal, à la confidentialité et l'intégrité des données qu'il contient. On parle aussi de faille de sécurité informatique.

2.2 Causes

La cause des vulnérabilités informatiques est souvent la négligence ou l'inexpérimentation d'un programmeur. Une vulnérabilité permet généralement à l'attaquant de tromper l'application, par exemple en dépassant les vérifications de contrôle d'accès ou en exécutant des commandes sur le système hébergeant l'application.

Exemples de vulnérabilités

- Dépassement de tampon
- Injection **SQL**
- Cross site scripting

Quelques vulnérabilités surviennent lorsque l'entrée d'un utilisateur n'est pas contrôlée, permettant l'exécution de commandes ou de requêtes **SQL** (connues sous le nom d'injection

SQL). D'autres proviennent d'erreurs d'un programmeur lors de la vérification des buffers de données (qui peuvent alors être dépassés), causant ainsi une corruption de la pile mémoire (et ainsi permettre l'exécution de code fourni par l'attaquant).

2.3 Identification et correction des vulnérabilités

Il existe de nombreux outils qui peuvent faciliter la découverte de vulnérabilités sur un système information, certains permettant leur suppression. Mais, bien que ces outils puissent fournir à un auditeur une bonne vision d'ensemble des vulnérabilités potentiellement présentes, ils ne peuvent pas remplacer le jugement humain. Se reposer uniquement sur des scanners automatiques de vulnérabilité rapportera de nombreux faux positifs et une vue limitée des problèmes présents dans le système.

Exemples de ces outils

Nessus, OpenVAS scanners de vulnérabilités

Wapiti, Nikto analyseurs de Site Web

Absynthe, SQLNinja, SQLInject analyseurs de Base de données

3 Les types d'attaques

les attaques[3], qui visent ces failles, (voir fig I.1) peuvent être à la fois très variées et très dangereuses. C'est pourquoi nous allons dans tout d'abord analyser ce que nous appellerons « l'anatomie d'une attaque », ensuite, nous caractériserons ces attaques et observerons leur déroulement.

3.1 Anatomie d'une attaque

Fréquemment appelés « les 5 P » dans la littérature, ces cinq verbes anglophones constituent le squelette de toute attaque informatique : Probe, Penetrate, Persist, Propagate, Paralyze.

Probe consiste en la collecte d'informations par le biais d'outils

Penetrate utilisation des informations récoltées pour pénétrer un réseau.

Persist création d'un compte avec des droits de super utilisateur pour pouvoir se réinfiltrer ultérieurement.

Propagate cette étape consiste à observer ce qui est accessible et disponible sur le réseau local.

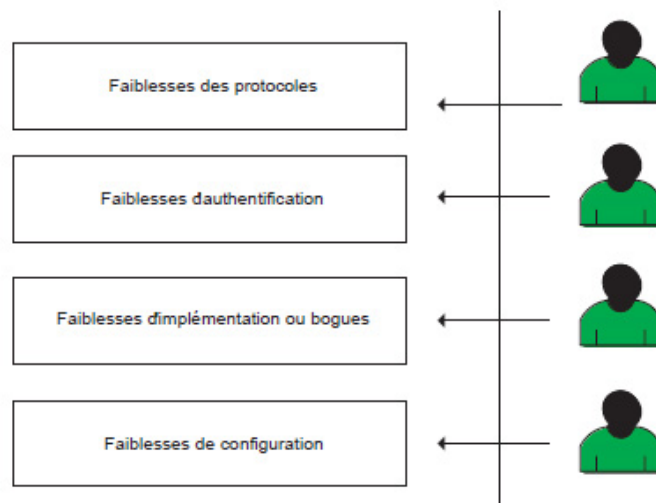


Figure I.1 – Typologie des faiblesses de sécurité.

Paralyze cette étape peut consister en plusieurs actions. Le pirate peut utiliser le serveur pour mener une attaque sur une autre machine, détruire des données ou encore endommager le système d'exploitation dans le but de planter le serveur.

Après ces cinq étapes, le pirate peut éventuellement tenter d'effacer ses traces, bien que cela ne soit rarement utile.

3.2 Les attaques réseaux

Les attaques[4] réseaux se basent principalement sur des failles liées aux protocoles ou à leur implémentation. Les **RFC** ne sont parfois pas assez spécifiques, et un choix particulier d'implémentation dans les différents services ou clients peut entraîner un problème de sécurité.

Observons quelques attaques bien connues.

3.2.1 Les techniques du scan

Les scans de ports ne sont pas des attaques à proprement parler. Son but est de déterminer quels sont les ports ouverts, et donc en déduire les services qui sont exécutés sur la machine cible (ex : port 80/**TCP** pour un service **HTTP**).

3.2.2 IP Spoofing

But Il s'agit d'usurper l'adresse **IP** d'une autre machine.

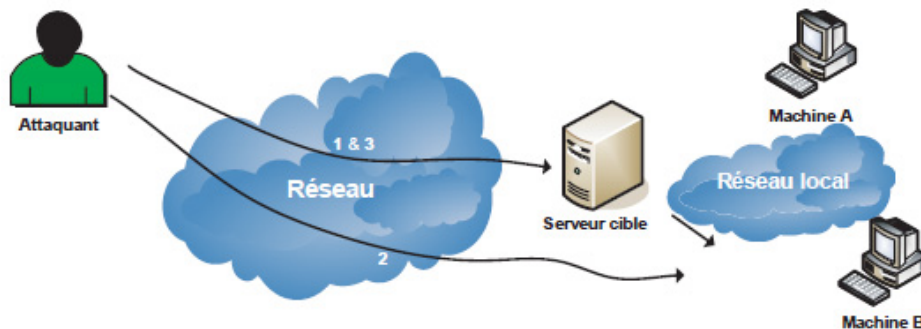


Figure I.2 – L'attaque IP spoofing.

Finalité Il s'agit de pouvoir passer à une autre machine en truquant les paquets **IP**. Cette technique peut être utile dans le cas d'authentifications basées sur une adresse **IP** (services tels que ssh par exemple).

Déroulement En effet, il existe des utilitaires qui permettent de modifier les paquets IP ou de créer ses propres paquets. Grâce à ces utilitaires, il est possible de spécifier une adresse IP différente de celle que l'on possède, d'où se faire passer pour une autre machine.

3.2.3 ARP Spoofing (ou ARP Redirect)

But Il s'agit de rediriger le trafic d'une machine vers une autre.

Finalité On peut se faire passer pour une autre machine. De plus, on peut rerouter les paquets qu'on reçoit vers le véritable destinataire, ainsi l'utilisateur usurpé ne se rendra compte de rien. La finalité est la même que l'**IP** spoofing mais on travaille ici au niveau de la couche liaison de données.

Déroulement pour effectuer cette usurpation, il faut corrompre le cache **ARP** de la victime. Ce qui signifie qu'il faut lui envoyer des trames **ARP** en lui indiquant que l'adresse **IP** d'une autre machine est la sienne. Les caches **ARP** étant régulièrement vidés, il faudra veiller à maintenir l'usurpation.

3.3 Les attaques applicatives

Les attaques applicatives se basent sur des failles dans les programmes utilisés, ou encore des erreurs de configuration. On peut classer ces attaques selon leur provenance.

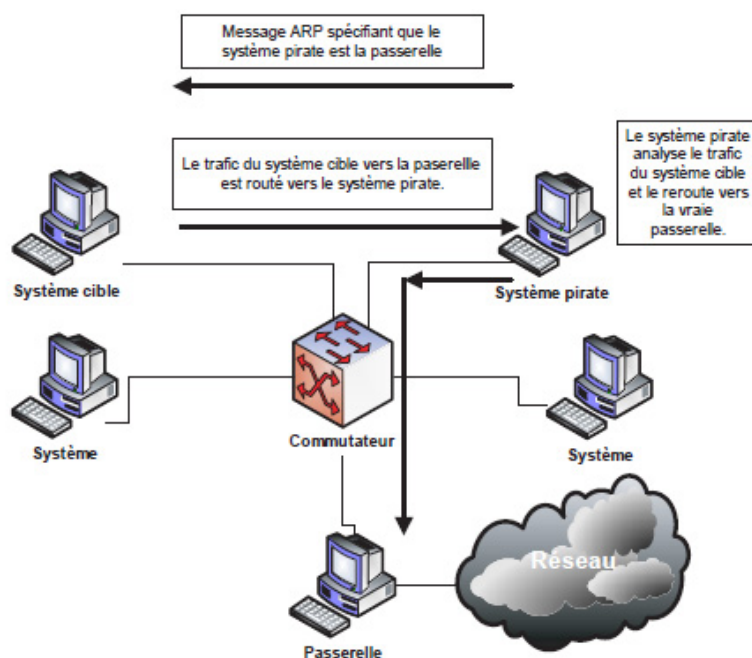


Figure I.3 – L’attaque ARP spoofing.

3.3.1 Les problèmes de configuration

Il est très rare que les administrateurs réseaux configurent correctement un programme. En général, ils se contentent d’utiliser les configurations par défaut. Celles-ci sont souvent non sécurisées afin de faciliter l’exploitation du logiciel (ex : login/mdp par défaut d’un serveur de base de données).

3.3.2 Les buffer overflows

Les buffer overflows, ou dépassement de la pile, sont une catégorie de bug particulière. Issus d’une erreur de programmation, ils permettent l’exploitation d’un shellcode à distance. Ce shellcode permettra à une personne mal intentionnée d’exécuter des commandes sur le système distant, pouvant aller jusqu’à sa destruction.

3.3.3 Les scripts

Principalement web (ex : Perl, PHP, ASP), ils s’exécutent sur un serveur et renvoie un résultat au client.

3.3.4 Les injections SQL

Tout comme les attaques de scripts, les injections *SQL* profitent de paramètres d'entrée non vérifiés. Comme leur nom l'indique, le but des injections *SQL* est d'injecter du code *SQL* dans une requête de base de données. Ainsi, il est possible de récupérer des informations se trouvant dans la base (exemple : des mots de passe) ou encore de détruire des données.

3.4 Le Déni de service

Le déni de service, ou DoS (Denial of Service), est une attaque qui vise bloquer un service, un système ou un réseau. Ces attaques s'appuient généralement sur une faiblesse d'implémentation, ou bogue, ou sur une faiblesse d'un protocole.

Citons quelques attaques réseaux connues permettant de rendre indisponible un service :

3.4.1 SYN Flooding

exploite la connexion en 3 phases de TCP (Three Way Handshake : **SYN** / **SYN-ACK** / **ACK**). Le principe est de laisser un grand nombre de connexions **TCP** en attente. Le pirate envoie de nombreuses demandes de connexion (**SYN**), reçoit les **SYN-ACK** mais ne répond jamais avec **ACK**.

3.4.2 UDP Flooding

le trafic **UDP** est prioritaire sur **TCP**. Le but est donc d'envoyer un grand nombre de paquets **UDP**, ce qui va occuper toute la bande passante et ainsi rendre indisponible toutes les connexions **TCP**.

3.4.3 Packet Fragment

utilise une mauvaise gestion de la défragmentation au niveau **ICMP**.

Exemple : ping of death. La quantité de données est supérieure à la taille maximum d'un paquet **IP**.

3.4.4 Smurfing

le pirate fait des requêtes **ICMP ECHO** à des adresses de broadcast en spoofant l'adresse source (en indiquant l'adresse de la machine cible). Cette machine cible va recevoir un nombre énorme de réponses.

4 Conclusion

De plus en plus la sécurité contre les attaques distantes se renforce, notamment par le biais d'équipements réseaux plus puissants (comme des firewalls plus intelligents), mais les attaques locales restent encore fort efficaces. Vue que l'informatique évolue alors les applications sont de plus en plus complexes et les délais laissés aux programmeurs et administrateurs sont souvent trop courts. Les risques de failles applicatives sont, de ce fait, très grands et peuvent s'avérer dangereux pour des applications largement répandues.

Chapitre II

Analyse réseau

1 Définition

1.1 Qu'est ce qu'un Analyseur réseau ?

Un « analyseur réseau »^[4] (appelé également analyseur de trames ou en anglais *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.).

1.2 Utilisation du sniffer

Un sniffer est un 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**, pour intrusion detection system) sont basés sur un sniffeur pour la capture des trames, et utilisent une base de données de règles pour détecter des trames suspectes. le sniffer peut aussi 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 fils 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.

1.3 Technique de fonctionnement

Pour pouvoir écouter tout le trafic sur une interface réseau, celle-ci doit être configurée dans un mode spécifique, le « **mode promiscuous** ». Ce mode permet d'écouter tous les paquets passant par l'interface, alors que dans le mode normal, le matériel servant d'interface réseau élimine les paquets n'étant pas à destination de l'hôte.

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 messagerie via le protocole **POP** ou **IMAP**, ou bien surfe sur internet sur des sites dont l'adresse ne commence pas par **HTTPS**, toutes les informations envoyées ou reçues peuvent être interceptées. 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.

Le packet sniffer décompose ces messages et les rassemble, ainsi les informations peuvent être analysées à des fins frauduleuses (détecter des logins, des mots de passe, des emails), analyser un problème réseau, superviser un trafic ou encore faire de la **rétro-ingénierie**.

2 Outils d'analyse réseau

2.1 Wireshark

Wireshark[5] (connu comme Eathreal jusqu'à une discussion de la marque en été 2006) est un analyseur libre du protocole du réseau pour Unix et Windows. Il nous permet d'examiner des données d'un réseau en direct ou d'un dossier de la capture sur disque. On conserve les données de la capture interactivement, en creusant en bas dans seulement l'égal de détail du paquet qu'on a besoin.

Wireshark a des plusieurs traits puissants, y compris une langue du filtre de l'exposition riche et la capacité de regarder le ruisseau reconstruit d'une session **TCP**. Il supporte aussi des centaines de protocoles et types de média. Mais cet Ethereal a souffert de douzaines de trous de sécurité exploitables de façon distante, donc il faut être à jour et prudent de le faire tourner sur les réseaux hostiles (tel que conférences de la sécurité).

2.2 Tcpcmdump

Tcpcmdump est le renifleur classique de l'**IP** qui était utilisé avant qu'Ethereal (Wireshark) est venu sur la scène, et une majorité continu à l'utiliser fréquemment. Il ne peut pas avoir

les cloches et les sifflements comme Wireshark , mais il travail bien et avec moins de trous de sécurité.

Il exige aussi moins de ressources du système. Pendant qu'il ne reçoit pas souvent de nouveaux traits, il est maintenu active pour arrange des insectes et des problèmes de la transférabilité. Il y a une version Windows séparé nommé WinDump.

2.3 Ettercap

C'est un renifleur du réseau terminal-basé pour les réseaux Ethernet **LAN**. Il supporte des examinateurs active et passive de beaucoup de protocoles (même chiffrés, comme **ssh** et **http**). Une injection des données dans un rapport peut être établie même un filtrage rapide est aussi possible, en gardant le rapport synchronisé. Beaucoup de modes reniflant ont été rendues effectif pour vous donner une suite reniflant puissante et complète. Il a la capacité de vérifier si vous êtes dans un **LAN** switcher ou pas, et utilise des empreintes digitales du Système d'exploitation (actif ou passif) pour nous faire savoir la géométrie du **LAN**.

2.4 Dsniff

C'est à la fois une suite d'utilitaire et un utilitaire lui-même d'audit réseau permettant aisément de sniffer les passwords circulants en clair, dans des protocoles non sécurisés.

Ainsi, Il détecte automatiquement les protocoles d'application, en capturant seulement les données intéressantes (Des mots clefs : pass, user,...). Elles concernent les sessions **FTP**, **telnet**, **SMTP**, **HTTP**, **POP**, **NNTP**, **IMAP**, **SNMP**, **LDAP**, **rlogin**, **RIP**, **OSPF**, **PPTP**, **AIM**, **IRC**,... Chacun de ses programmes a un but précis (capture de mail, de fichier, d'url...).

2.5 SoftPerfect Analyser

C'est est un outil avancé, professionnel pour analyser, en dépannant, en maintenant et dirigeant des réseaux locaux et des rapports Internet.

Il capture les données qui passent à travers la carte Ethernet du réseau, analyse ces données puis le représente dans une forme lisible. Il est un outil utile pour un administrateurs du réseau, spécialistes de la sécurité, promoteurs de l'application du réseau et n'importe qui a besoin d'une image complète de la circulation qui traverse leur rapport du réseau ou segment d'un réseau de région local.

2.6 Kismet

Kismet est un sniffer et un détecteur d'intrusion. Il fonctionne avec toutes les cartes supportant le monitoring (**rfmon**) et peut sniffer le trafic 802.11b, 802.11a, et 802.11g. Il est composé d'un serveur et d'un client (par défaut l'accès est restreint à l'adresse local) ce qui permet à plusieurs utilisateurs de voir un seul serveur Kismet simultanément. il peut détecter automatiquement les bloques IP du réseau en reniflant TCP, UDP, ARP et paquets DHCP, circulant dans le réseau estimé.

3 Tableau comparatif des outils

<i>Critères</i>	Wireshark	Tcpdump	Cain and Abel	Ettercap	Dsniff	SoftPerfect An.	Kismet
<i>Matériel et Système</i>							
<i>Les plate formes d'utilisation</i>	Linux	Oui	Non	Oui	Oui	Non	Oui
	BSD	Oui	Non	Oui	Oui	Non	Oui
	Mac OS X	Oui	Non	Oui	Oui	Non	Oui
	Solaris	Oui	Non	Oui	Oui	Non	Non
	Microsoft Windows	Oui	Oui	Oui	Oui	Oui	Oui
<i>Type de réseau analysé</i>	Ethernet	Oui	Oui	Oui	Oui	Oui	Non
	Réseaux sans fil	Non	Oui	Non	Non	Oui	Oui
<i>Interfaces et Utilisation</i>							
<i>Interfaces offert</i>	Console	Oui	Non	Oui	Oui	Non	Oui
	GUI	Non	Oui	Oui	Non	Oui	Non
<i>Facilité d'utilisation</i>	*****	***	*****	*****	**	*****	**
<i>Outils</i>							
<i>Type d'outil</i>	Open source	Open source	Open source	Open source	Open source	Payant	Open source
<i>Capture du trafic</i>	Oui	Oui	Oui	Non	Non	Oui	Oui

Tableau II.1 – Tableau comparatif de quelques outils d’analyse réseau

Chapitre III

Tests d’Intrusion

1 Définition

1.1 Qu’est ce qu’un test d’intrusion ?

Les tests d’intrusion[6] constituent une tentative autorisée de simuler les activités d’un pirate qui veut s’approprier des ressources qui ne sont pas les siennes, ou nuire au bon fonctionnement d’un système d’informations, par exemple en le rendant indisponible.

Ces tests permettent d’avoir une image claire de la sécurité globale d’une entreprise ou d’un accès Internet chez un particulier. Ils correspondent à des attaques simulées d’un réseau. Ils permettent de tester la robustesse de la sécurité, d’apprécier l’efficacité des mécanismes mis en oeuvre. Il est ainsi possible de savoir si les mécanismes mis en place permettent de stopper ou non un attaquant malintentionné.

Les tests d’intrusion ne peuvent pas se réduire à la simple utilisation d’un logiciel de détection automatique de vulnérabilités par balayage. En particulier ils nécessitent l’intervention d’une équipe de professionnels compétents qui eux vont identifier et qualifier les failles de manière plus réfléchie et auront à l’esprit les conséquences des tests qu’ils effectueront. Néanmoins, les scanners de vulnérabilité présentent un certain intérêt dans leur caractère automatique mais ils ne suffisent pas à eux seuls à obtenir une bonne détermination des failles de vulnérabilité que présente un réseau.

1.2 Stratégies des tests

Il existe plusieurs stratégies de tests :

- Les tests externes qui correspondent à un examen des services disponibles via Internet.
- Les tests internes qui exploitent les failles de vulnérabilités qui pourraient être disponibles à un attaquant en provenance d’Internet ayant réussi à s’introduire dans le

réseau ou à un employé malveillant.

Les méthodes et les techniques utilisés dans les tests internes ou externes sont identiques. La seule différence notable est l’étendue des connaissances relatives au réseau, en possession des attaquants.

Pour simuler ce degré de connaissance du système, les tests d’intrusion peuvent se faire de plusieurs façons :

Test en aveugle : les équipes en charge du test ont un accès limité aux renseignements relatifs à la configuration du système d’information

Test en double aveugle : seule la personne qui est à l’initiative du test est au courant, la personne en charge de la sécurité ne l’est pas.

Test ciblé : l’équipe de sécurité est au courant et a des connaissances sur le réseau et sur la cible visée.

1.3 Types des tests

Il existe différents types de tests parmi lesquels nous pouvons noter ceux relatifs :

- la sécurité des applications Web
- Les dénis de service (DoS)
- le scannage de numéros de téléphone (War dialing)
- Au réseau sans fil
- l’ingénierie sociale

1.4 Leurs limites

Un test d’intrusion peut échouer mais ça ne signifie pas que le système ne présente pas de faille de vulnérabilité.

Il est impossible de tester toutes les failles de vulnérabilité présentes dans un réseau. Par exemple, Les scanners de vulnérabilité ne simulent pas toutes les nouvelles failles.

De plus, il est nécessaire de répéter de façon régulière ces tests. Tout ajout de matériel, l’apparition de nouveaux outils de piratage ou de nouvelles technologies remettent en cause les résultats des tests d’intrusion.

1.5 La démarche utilisée dans les tests d’intrusion

Nous nous intéressons ici à la description de la démarche employée dans les tests d’intrusion :

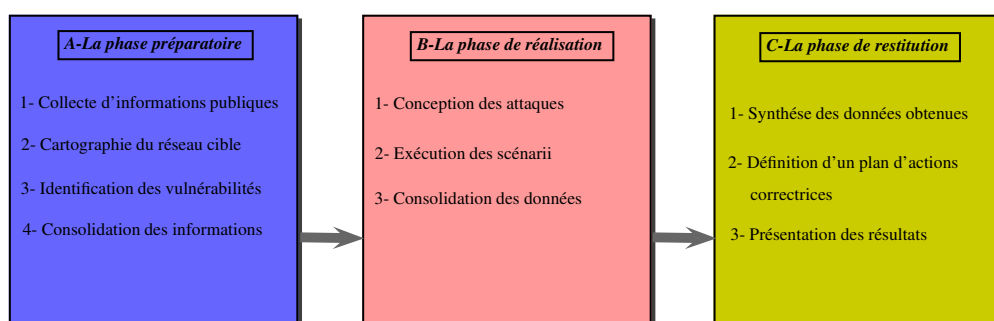


Figure III.1 – La démarche utilisée dans les tests d'intrusion.

A. La phase préparatoire

1. collection d'informations publiques (DNS, WhoIs, ...)
2. cartographie réseau de la cible (ping, traceroute, nmap)
3. identification de vulnérabilité (Nessus)
4. consolidation de données obtenues

B. La phase de réalisation

1. conception des scénarii d'attaques à évaluer
2. exécution des scénarii
3. consolidation des données obtenues

Dans le cas des **hackers**, l'intrusion s'arrête ici. Ils nettoient généralement les traces laissées par leur passage.

Dans le cas d'un audit de sécurité, il faut alors fermer les brèches ouvertes, puis passer à l'étape suivante.

C. La phase de restitution

1. synthèse des données obtenues lors des phases préparatoires puis de réalisation
2. définition d'un plan d'actions correctrices
3. présentation des résultats au commanditaire du test

Nous illustrerons dans la suite cette démarche en indiquant les différents outils utilisés agrémentés d'exemples.

2 L’audit de vulnérabilité

Nous parlons ici de l’audit[7] en tant qu’étape de la phase préparatoire de la démarche utilisée dans les tests d’intrusion. Dans le but d’identifier les failles de vulnérabilité, elle consiste à récupérer des informations relatives aux réseaux et aux systèmes présents sur ces derniers. Les failles de vulnérabilité résultent en général de limites jointes à la conception des technologies ou découlent de mauvaises configurations ou utilisations. Les tests d’intrusions donnent des indications sur la facilité ou à l’inverse la difficulté d’accéder à l’information et au système d’informations en exploitant les vulnérabilités de sécurité. Les scanners de vulnérabilité correspondent à une façon automatisée de mise en évidence de ces failles. Ils indiquent la façon dont il est possible d’exploiter ces vulnérabilités et les méthodes permettant de résoudre les problèmes. Ils couvrent, en général, un large éventail de vulnérabilités connues. Tandis que les tests d’intrusion ciblent certaines vulnérabilités.

3 Outils liés aux tests d’intrusion

Nous allons reprendre dans cette partie les différentes étapes de la démarche utilisée dans les tests d’intrusion que nous avons présentées précédemment, dans lesquelles sont utilisés des outils spécifiques. Nous donnerons également des exemples afin de mieux illustrer la démarche.

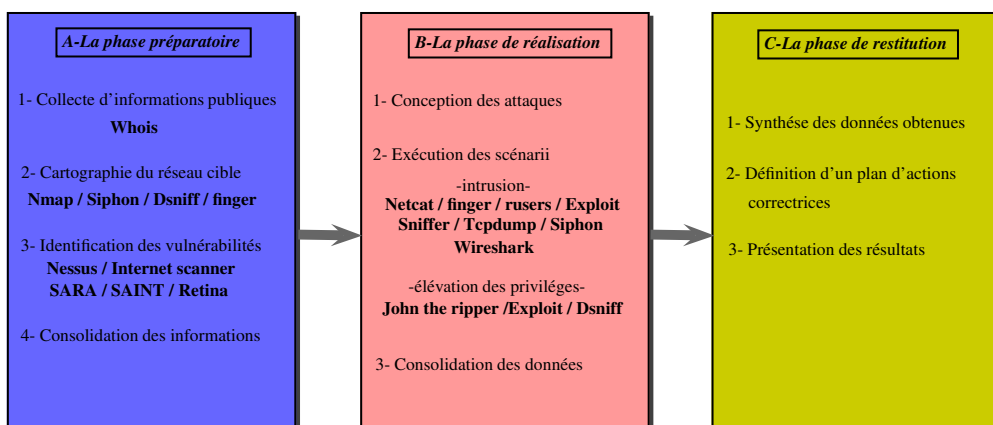


Figure III.2 – Exemples d’outils utilisés lors d’une intrusion.

3.1 Divers outils

3.1.1 Etape de collecte d'informations publiques (A.1.)

La commande **Whois** permet d'obtenir des informations publiques correspondant au réseau cible : nom du serveur **DNS**, nom du responsable, numéro de téléphone, adresse e-mail, description du réseau etc.

3.1.2 Etape de cartographie du réseau cible (A.2.)

Les outils utilisés dans cette étape permettent de récolter des informations relatives à la topologie du réseau afin de déterminer sa structuration. Parmi ces outils, nous pouvons trouver :

Nmap qui est un scanner de réseau. Il permet de savoir quels sont les ports ouverts, fermés ou filtrés , ainsi que le système d'exploitation autorisé et sa version. Il permet par exemple de scanner un ensemble d'**adresses IP** en précisant la méthode de scan utilisée, les types de ports tels que les ports **UDP**, en tentant d'identifier la machine cible et en sauvegardant le résultat dans un fichier.

Siphon permet de découvrir la topologie de la portion de réseau sur laquelle se trouve la machine où nous le lançons. Il indique les systèmes d'exploitation présents sur les machines , les ports ouverts , les machines qui ont le droit de se connecter au réseau. Il est ainsi possible de savoir pour quelle machine nous devons nous faire passer, afin de contourner les Firewalls.

Dsniff permet de visualiser les paquets présents sur le réseau et ainsi de récupérer des clefs en sniffant (**sniffer**).

Finger permet d'obtenir des comptes valides. En général, le **démon** correspondant est désactivé.

3.1.3 Etape d'identification des vulnérabilités (A.3.)

Nous voulons identifier les failles potentielles présentes sur le réseau en utilisant des scanners de vulnérabilité tels que **Nessus**. L'utilisation de ces logiciels n'est pas très discrète. En effet, étant donné que ces logiciels testent des failles bien connues des **NIDS**, ils sont facilement repérables. Les **NIDS** sont des systèmes de détection d'intrusion basés au niveau d'un réseau.

Une fois un certain nombre de vulnérabilités est identifiées, il est alors nécessaire d'éliminer les failles non fondées.

3.1.4 Etape d’exécution des scénarii (B.2.)

Dans cette étape, nous pouvons distinguer deux ensembles de produits utilisés ceux qui permettent de s’introduire sur un ordinateur ou un serveur et ceux qui permettent de se procurer des privilèges auxquels nous n’avons normalement pas accès.

3.2 Outils d’audit

Il ya de nombreux logiciels qui permettent d’automatiser la découverte de vulnérabilités, ils sont appelés des scanners. Ils permettent d’évaluer les vulnérabilités présentes sur les réseaux. Ils se déclinent sous plusieurs formes et donnent des résultats avec des précisions variables.

Parmi ces logiciels, nous pouvons citer :

3.2.1 Internet Scanner

Il peut s’intégrer au produit **ISS** Décisions pour être utilisé avec d’autres produits de sécurité tels que les systèmes de détection d’intrusions et les **firewall** (pare-feu).

3.2.2 SATAN, SAINT, SARA

SATAN avait la particularité d’être **open source**. Il n’est plus mis à jour depuis plusieurs années. Mais, il existe une myriade d’outils similaires tel que l’outil **open source SARA** qui est la troisième génération d’outil d’analyse basé sur **SATAN**, ou la solution commerciale **SAINT**.

3.2.3 Retina

Nous pouvons également citer le logiciel **Retina** de la société eEye, qui est rapidement devenu populaire. Il analyse le trafic sur chaque port afin de déterminer le service utilisé. Il existe une fonction nommée **CHAM** permettant de découvrir de nouvelles failles de vulnérabilité. Cette méthode repose sur un moteur d’intelligence artificiel. **Retina** est une solution commerciale.

3.2.4 Nessus, NeWT

Enfin, il existe une autre offre dont nous allons parler de façon plus approfondie dans ce document, l’outil Nessus, et sa version Windows appelée **NeWT**. **Nessus** est un outil **open**

source. Un français, renaud Deraison, est l'auteur et l'animateur de se projet. La version Windows est disponible sur le site de la société Tenable Network Security en version d'évaluation. Nessus semble être l'un des outils les plus populaires du moment.

Chapitre IV

Réalisation

1 Introduction

Suite à l'étude élaborée dans les trois chapitres précédents on a atteint la dernière phase de la réalisation de nos tests. Ce chapitre est destiné pour décrire cette phase et présenter les résultats aux quels on a aboutit. Nous commencerons d'abord par présenter les différents outils techniques à utiliser pour réaliser le travail.

2 Environnement de travail

2.1 Environnement matériel

La plate-forme de travail est une machine de caractéristique suivantes :

- Processeur : Intel core 2 duo 2.00 GHz
- RAM : 2043 Mo

2.2 Environnement logiciel

Système d'exploitation :

- Windows vista Edition familial premium
- Windows Xp sp2
- Mandriva Linux 2009 spring
- Linux Backtrack 4 beta

Machine virtuelle :

- VMware Workstation v6.0.3
- Sun VirtualBox v3.1.6

Et LATEX comme environnement de rédaction du rapport.

2.3 Back|Track

2.3.1 L'outil

BackTrack [8] est une distribution GNU/Linux basée sur la distribution *Slackware* jusqu'à la version 3 et *Ubuntu* depuis la version 4 apparue en 11 Janvier 2010, elle est née de la fusion de **Whax** et **Auditor**. Son objectif est de fournir une distribution regroupant l'ensemble des outils nécessaires aux tests de sécurité d'un réseau.

Avec ces 300 outils, BackTrack aborde tous les domaines liés aux sécurités modernes allant de l'audit réseau à l'analyse et l'identification de vulnérabilités en passant par divers outils de récupération d'informations. (**Fuzzers** / Testeurs de sécurité des réseaux filaires / Testeur des réseaux **wifi** ...)

BackTrack est principalement connu et utilisé à des fins d'audit de réseaux sans fil wifi. Son développement est axé sur la prise en charge de cartes wifi supportant le mode Monitoring, ce qui permet la capture de paquets, nécessaire pour le crack de clé **WEP**/ **WPA** et autres test (suite de logiciel aircrack-ng par exemple)

BackTrack contient aussi des applications basiques comme un lecteur multimédia, traitement de texte ... ce qui en fait un système d'exploitation polyvalent.

2.3.2 Installation

L'un des principaux intérêts de BackTrack est d'être disponible sous forme d'un Livecd, c'est à dire qu'un ordinateur peut booter directement sur le cd sans avoir à se préoccuper d'une quelconque installation avec la possibilité d'exécuter chaque outil immédiatement.

Ainsi, tout se passe dans la mémoire RAM de l'ordinateur n'entraînant aucune intervention sur le disque dur permettant ainsi de l'utiliser sans risque de perte de données ou autre. Ce Livecd permet d'avoir tous les outils indispensables à la sécurité informatique sans laisser aucune trace.

2.3.3 Tests qu'on peut réaliser

Comme nous avons pu le dire précédemment, l'objectif de BackTrack est de fournir une distribution compacte regroupant le maximum d'outils nécessaire aux tests de sécurité d'un réseau ou d'application.

Le nombre d'outils liés à la sécurité informatique ne cesse de croître avec les versions de BackTrack.

Dans la version 4.0 beta, nous disposons d'approximativement 300 outils décomposés en 16 catégories. Voici les parties qu'il nous est possible d'utiliser dans de nombreux cas :

Analyse de réseau sans fil

Aircrack-ng, wesside-ng pour l'analyse et le crackage de Wifi.

BTscanner, BlueSnarfer pour l'analyse et le crackage de Bluetooth.

Anonymat

TOR permet d'être complètement invisible sur le net sans laisser la moindre trace à fin d'éviter de nombreux démêlés judiciaires, il peut être une bonne chose de ne pas laisser de traces sur les différents systèmes comme dans les fichiers de **logs** par exemple.

Attaque de mot de passe BackTrack dispose d'un large panel de possibilité d'attaque contre les mots de passe que ce soit des attaques << en ligne >> ou bien << locale >>.

Hydra concernant les attaques en lignes, il est actuellement considéré comme l'un des meilleurs brute forceur en ligne.

RainbowCrack Concernant les attaques hors lignes, il est un casseur de mot de passe basé sur les rainbows tables ce qui le rend excessivement puissant.

Collecte d'informations Il faut toujours prendre très au sérieux cette partie qui est la base de toute attaques de petite ou de grande envergure. Suivant le type d'informations que nous désirons rechercher, nous avons la possibilité de choisir parmi les nombreux outils que nous propose BackTrack.

Wapiti nous permettre de récupérer des informations sur les failles Web

Nessus nous permettre de visualiser les vulnérabilités d'un système d'exploitation ainsi que ces services

Nmap simplement nous renseigner sur les ports ouverts

Pénétration l'étape suivant la récupération d'information est en règle générale la pénétration de la cible.

Métasploit Il contient une base de données d'environ 300 exploits, capable de simplifier les tests d'intrusion sur des failles importantes.

Reverse engineering C'est l'activité qui consiste à étudier un programme pour en déterminer le fonctionnement interne ou sa méthode de fabrication afin de pouvoir par exemple s'octroyer des accès ou des fonctionnalités ne nous étant pas autorisé initialement.

Ollydbg outils très performant en matière de cracking pour débogueur/désassembleur

Hexedit comme éditeur hexadécimal

Sniffers les sniffers sont des logiciels qui peuvent récupérer les données transitant par le biais d'un réseau local.

- Dsniff
- Ettercap-ng(anciennement Ettercap)
- Wireshark(anciennement Ethereal)

2.3.4 Evolutions

BackTrack est à l'heure actuelle la distribution la plus aboutie en matière de sécurité informatique. BackTrack se qualifie tant par le nombre impressionnant d'outils que leur par qualité reconnu par les professionnels. Ces nombreux développeurs et sa large communauté permettent d'avoir une distribution de plus en plus stable avec une compatibilité accrue avec les différents constructeurs de matériels. BackTrack a gagné une grande notoriété et c'est pourquoi, en 2006, BackTrack a été élu comme étant la première distribution de sécurité par insecure.org.

3 Les tests réalisés

Après la définition de différents outils technique et environnements de tests pour l'achèvement de ce travail nous allons dans ce qui suit présenter les tâches et les tests réalisés.

Nous allons dans cette partie effectuer une série de tests à l'aide de Backtrack. Suite à des contraintes de type matériel, la stratégie adoptée est une stratégie de type interne. Le client et le serveur sont tous deux situés à l'intérieur d'un réseau local **Ethernet**. L'objectif est de mieux appréhender ce qui sont les failles de sécurité dans un tel réseau au travers de tests simples.

3.1 Topologie

Nous avons créé 3 machines virtuelles sur VMware, pour qu'elles puissent communiquer directement sur le réseau physique. Pour ce là, nous avons utilisés le mode **Host-Only** qui permet de connecter les cartes réseau (virtuelles) des VM directement à la carte physique.

Cette connexion permet entre autres d'activer le service **DHCP-Server**, afin de délivrer des adresses **IP** aux machines virtuelle du réseau **Host-Only**. Du point de vue de la **VM**, c'est comme si elle été directement connectée au réseau physique. Techniquement, un **vSwitch** est créé, et est connecté à la carte réseau physique, via l'interface associée. Ceci explique pourquoi il est possible d'utiliser des sniffers de paquets directement sur l'interface dans le but d'écouter le trafic réseau des machines virtuelles.

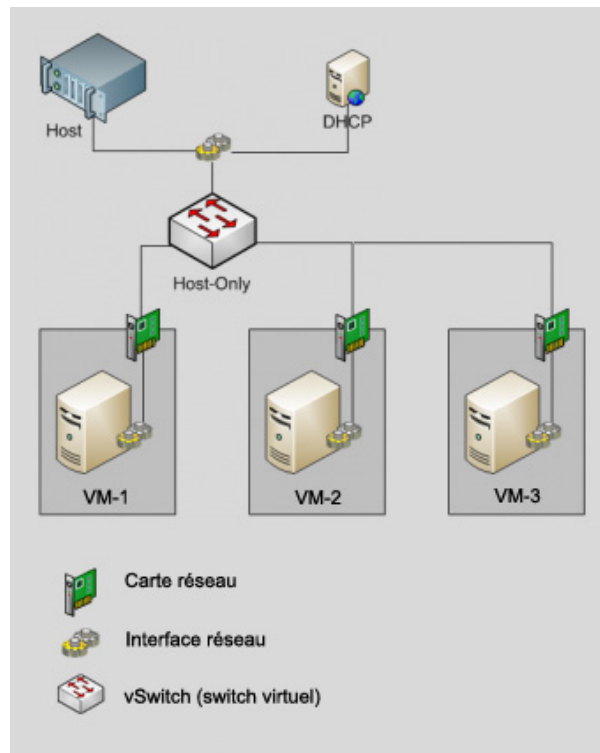


Figure IV.1 – Le réseau ETHERNET utilisé pour les tests

Différents systèmes ont été installés sur ces machines virtuelles. Les configurations de ces derniers ont été indiquées dans le tableau

NB : cette topologie n'est valable que pour les deux premiers test (test 1 et test 2) qu'on a pu réaliser.

V.M.		VM-1	VM-2	VM-3
Adresse IP		192.168.56.102	192.168.56.104	192.168.56.101
Carte réseau	Type de carte	PCnet-FAST III (Am79C793)	PCnet-FAST III (Am79C793)	PCnet-FAST III (Am79C793)
	Adresse MAC	08 00 27 1F 04 70	08 00 27 1C 73 44	08 00 27 2B 50 FD
Mémoire vive		512 Mo	256 Mo	256 Mo
OS		Linux Backtrack 4 beta	Linux Mandriva	Windows XP-sp2

Tableau IV.1 – Les configurations des machines virtuelles

4 Les tests réalisés

4.1 Test 1 :

4.2 Test 2 : Sniffer le trafic réseau

À fin de récupérer les mots de passe circulant dans le flux de notre réseau ou' la grande majorité des protocoles font transiter les informations en clair, c'est-à-dire de manière non chiffrée. Ainsi, lorsqu'un utilisateur du réseau consulte un serveur via le protocole FTP ou sFTP, ou accéder sur des sites dont l'adresse commence par HTTP ou HTTPS, toutes les informations envoyées ou reçues peuvent être interceptées.

On va utilisée :

- Wireshark pour récupérer les mot de passe circulant dans les paquets FTP et HTTP
- Ettercap pour récupérer les mot de passe circulant dans les paquets sFTP et HTTPS

Suite au résultat obtenue par le Test 1, on a constaté que le port FTP est ouvert ce qui nous a donné la possibilité de récupérer un login et son mot de passe pour accéder au serveur FTP.

4.2.1 Test 2-A : En utilisant Wireshark

Paramétrage

- le sniffeur est installer dans VM-1 à fin de capturer le trafic entre VM-2 et VM-1 .
- le serveur FTP (ou Web) est installer dans le VM-2.
- le client FTP (ou Web) est le VM-3

Exploitation

1. On lance la capture des trame dans l'analyseur Wireshark dans VM-1.

2. On fait un test de connexion, par login et mot de passe, du client VM-3 au serveur VM-2(l'adresse est ftp ://192.168.56.104/).
3. On fait le filtrage des paquets en capture et impression (avec personnalisation de l'utilisation de la couleur)

voici le détail des échanges entre le client et le serveur :

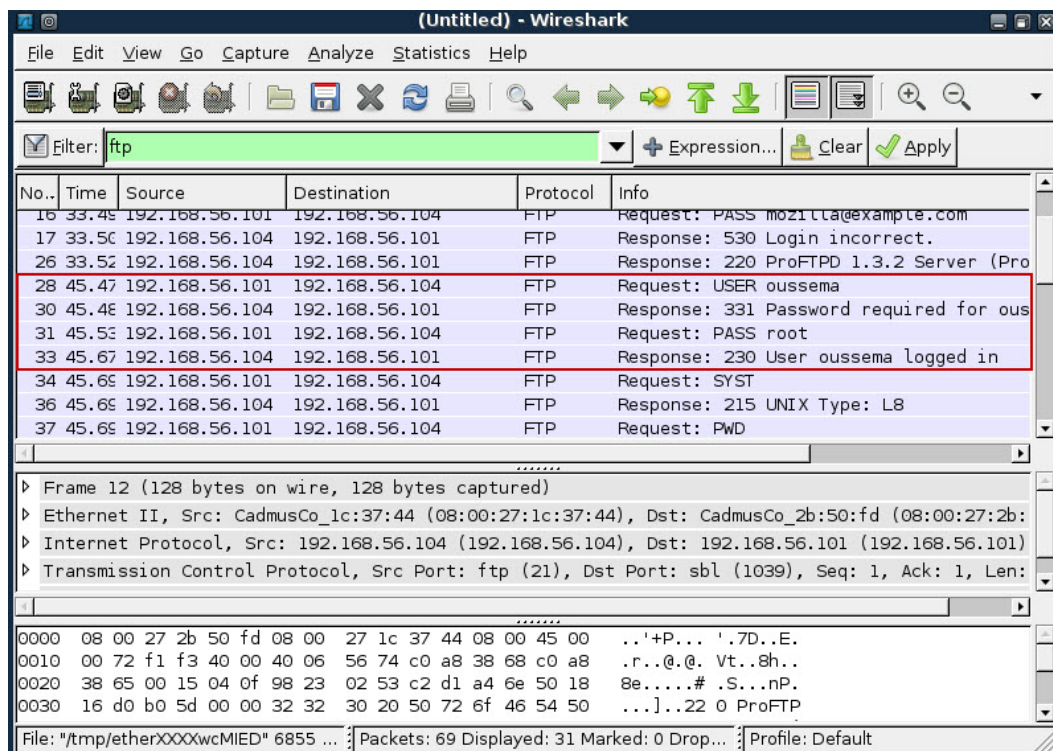


Figure IV.2 – Capture du trafic entre le client FTP et le serveur FTP avec Wireshark

On peut s'apercevoir que dans les échanges, les ligne avec l'ID entre 28 et 33 contient l'identifiant et le mot de passe de l'utilisateur FTP situé dans VM-1.

4.2.2 Test 2-B : En utilisant Ettercap

Paramétrage

- le sniffeur est installer dans VM-1
- sa configuratuion par la commande : “vim /etc/etter.conf”
- le serveur Web sécurisé (ou FTP sécurisée) est installer dans le VM-2.
- le client Web sécurisé (ou FTP sécurisée) est le VM-3.

Exploitation

1. On lance le sniffeur par la commande :

```
ettercap -TqM ARP :REMOTE /192.168.56.101/ / 192.168.56.104/
```

2. On fait le même test de connection dans le Test 2-A (l'adresse est ftp ://192.168.56.104/).
3. De plus on fait aussi un test de connection sur le serveur web sécurisé(l'adresse est https ://192.168.56.104/toto/secr.php).

Comme résultat, Ettercap nous donne les logins(USER) et leurs mot de passe(PASS). Il distingue automatiquement les mots de passe et les logins. Les résultats des échanges entre le client et le serveur dans la figure IV.3.

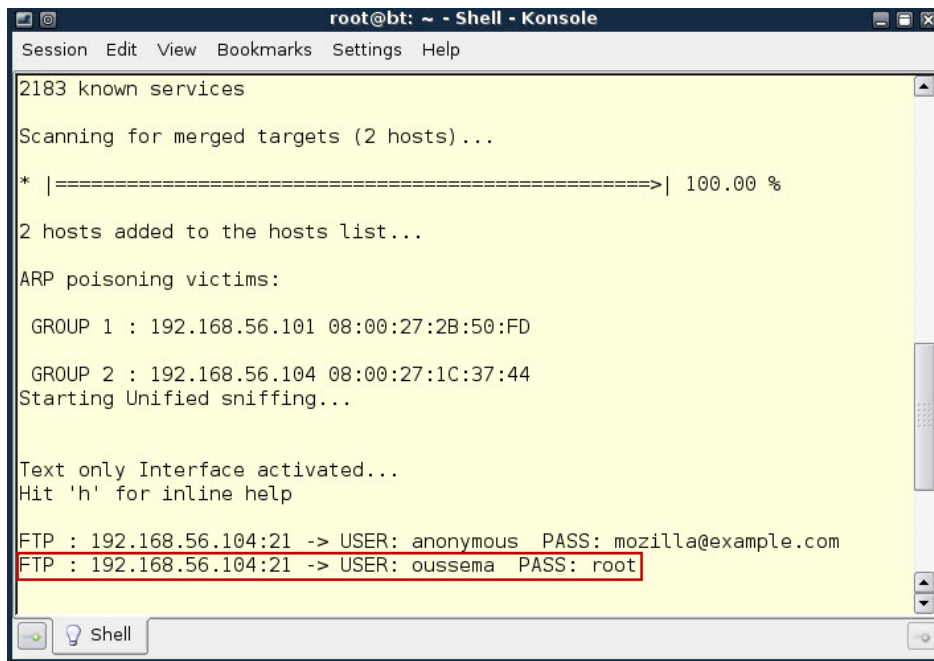


Figure IV.3 – Capture du trafic entre le client FTP et le serveur FTP avec Ettercap

4.3 Test 3 : Crackage d'un clé WEP dans un réseau sans fil (Wi-Fi)

La technologie sans fil Wi-Fi (IEEE 802.11) s'appuie sur les ondes hertziennes pour établir les communications entre les équipements. Il suffit de se trouver dans la zone de couverture des émetteurs pour écouter les données. Ainsi, Le protocole **WEP** chiffre chaque trame 802.11 échangée entre l'émetteur et le récepteur (point d'accès ou client).

Il existe plusieurs méthodes pour casser une clé WEP. On cite le processus de crackage par Aircrack-ng. La suite aircrack-ng comprend plusieurs programmes dont les 3 principaux sont :

Airodump-ng le logiciel de capture de paquets, c'est lui qui scan les réseaux et conserve les paquets qui serviront à décrypter la clé.

Aireplay-ng un logiciel dont la principale fonction est l'envoi de paquets dans le but de stimuler le reseau et capturer plus de paquets.

Aircrack-ng le logiciel de crack de clé, c'est un logiciel qui à partir des informations capturées à l'aide d'airodump va nous donner la clé (si on en a eu un nombre suffisant de paquets).

le processus d'attaque est situé dans la figure

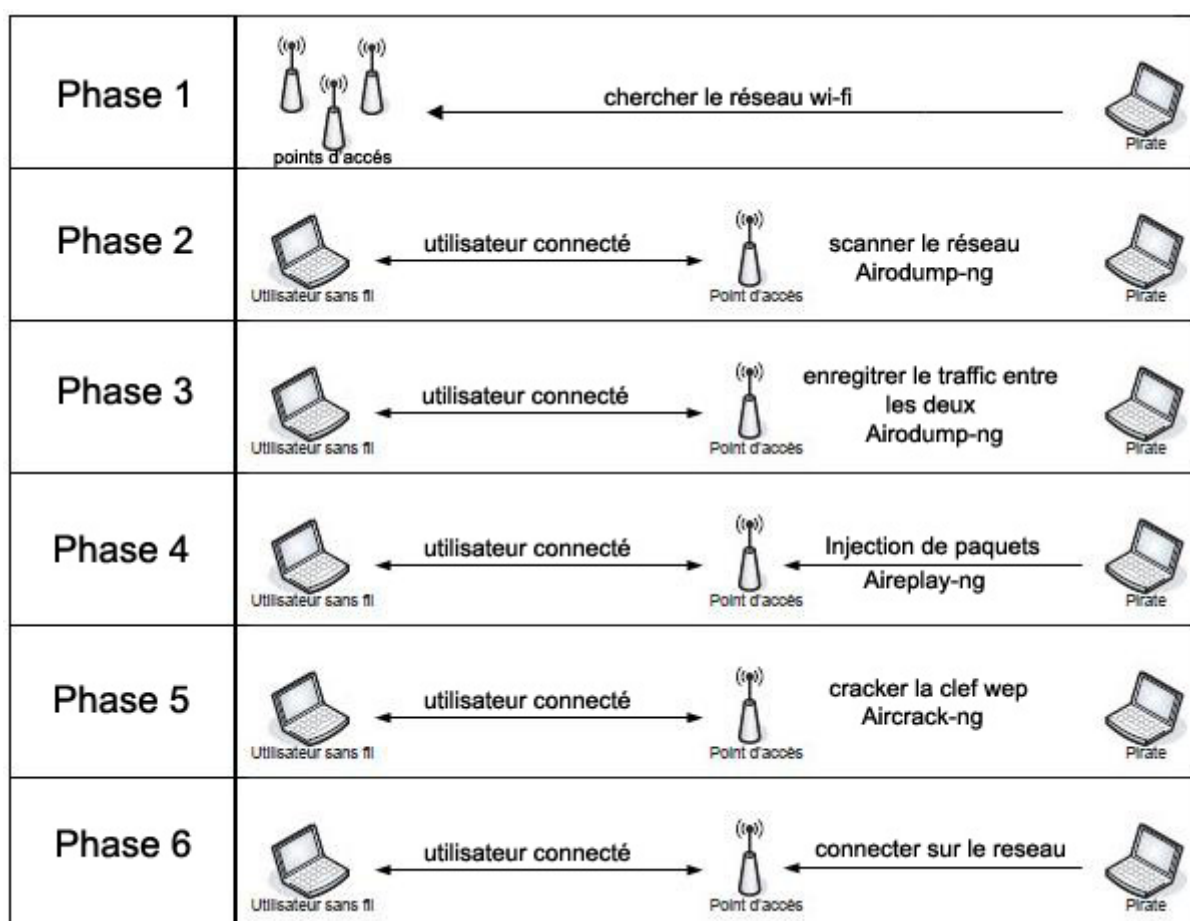


Figure IV.4 – le processus de crackage du clé Wi-Fi

4.3.1 Paramétrage

- M-1 est la machine du test de crackage qui a les même paramètre que VM-1 avec une carte wifi *chipset atheros* .

- Le point d'accès est R, un routeur wifi connecté à l'Internet .
- Une machine M-2 déjà connecté avec le point d'accès R par la clé secrète, alors il y'a des échanges de paquets entre ces derniers.(voir fig IV.5)

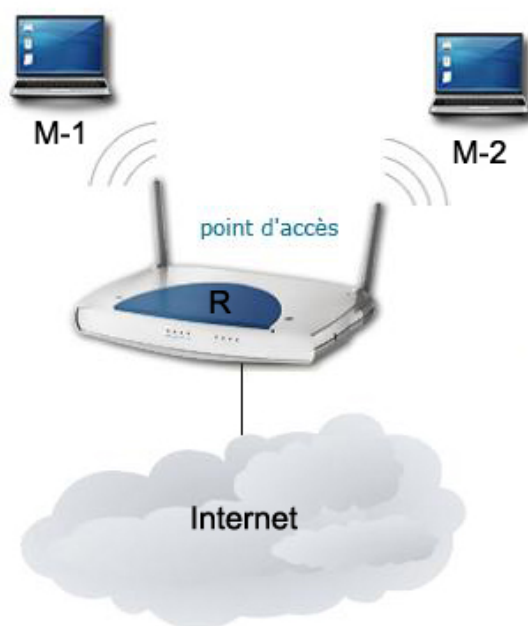


Figure IV.5 – Le réseau du test 3

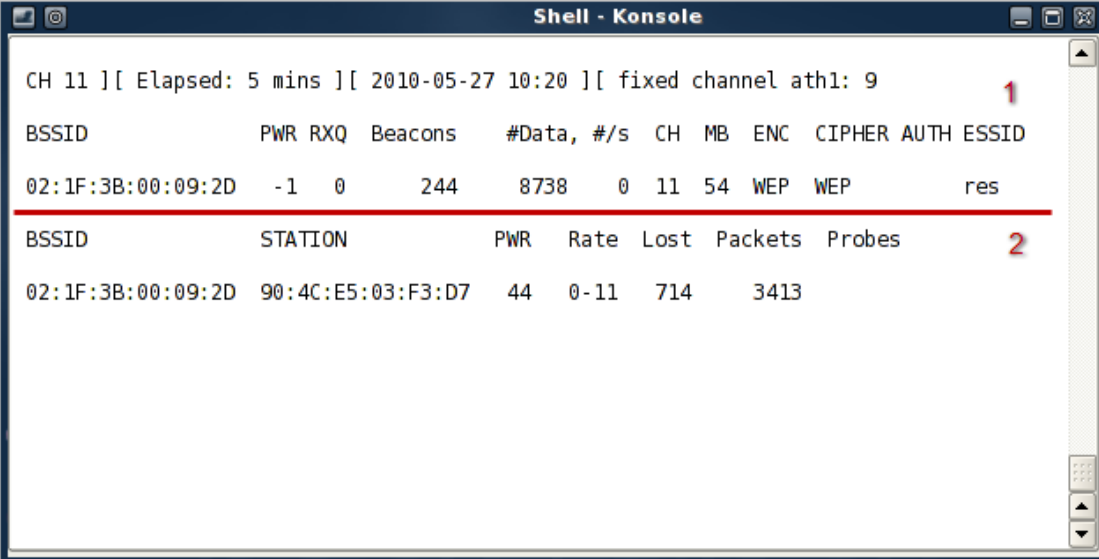
4.3.2 Exploitation

Phase 1 On commence à chercher le réseaux wifi voulu avec airodump par la commande :

```
airodump-ng interface
```

Une fois lancé airodump, on se trouve dans l'interface indiqué dans la figure

- La colonne **BSSID** correspond à l'adresse mac des points d'accès (R)
- La colonne **ESSID** correspond au nom du réseau
- La colonne **power** donne une indication de la puissance de reception, cette information peut foirer (-1) cela n'influ en rien la puissance de reception réelle , une bonne indication est la vitesse de défilement des beacons
- La partie (1) correspond aux points d'accès et la seconde partie aux stations (l' ordinateur qui se connecte à R c'est M-2).
- Airodump nous indique dans la colonne **ENC** le cryptage utilisé (WEP, WPA, OPN).On choisie notre réseau crypter par WEP.



```
CH 11 ][ Elapsed: 5 mins ][ 2010-05-27 10:20 ][ fixed channel ath1: 9
```

BSSID	PWR	RXQ	Beacons	#Data, #/s	CH	MB	ENC	CIPHER	AUTH	ESSID
02:1F:3B:00:09:2D	-1	0	244	8738 0	11	54	WEP	WEP		res

BSSID	STATION	PWR	Rate	Lost	Packets	Probes
02:1F:3B:00:09:2D	90:4C:E5:03:F3:D7	44	0-11	714	3413	

Figure IV.6 – le processus de crackage du clé Wi-Fi

- La colonne qui nous intéresse est la colonne des **IVs**, la colonne **#data**, c'est ces "bouts de fichiers" qui vont nous permettre de cracker notre clé wep aucun rapport avec la colonne beacons complètement inutile pour le crack.

Phase 2 et 3 : Lancer le scann après avoir choisie le réseau par la commande :

```
airodump-ng --write NomFichierSortie --channel NumeroChannel -b @mac_R Interface
```

Dans notre cas :

```
airodump-ng --write tuto --channel 11 -b 02:1F:3B:00:09:2D ath0
```

dontles paramètres sont :

- "*-write tuto*" *-write* indique que l'on souhaite enregistrer la capture, il est suivis du nom du fichier dans lequel on enregistre justement.
- "*-channell XX*" Indique sur quel channel on scan.

Il est nécessaire de connaître l'adresse mac d'un ordinateur (station) déjà accepté par le point d'accès R (dans notre cas l'ordinateur est M-2) car Pour aireplay, le programme qui va envoyer des paquets, a justement besoin de l'adresse mac de M-2. En fait, on se fait passer pour l'ordinateur M-2 qui a le droit d'accès au point d'accès R en spoofant son adresse mac pour pouvoir par exemple injecter des paquets ensuite.

Phase 4 Maintenant que l'on sait que le cryptage est WEP, qu'une station est présente et qu'il y a du trafic (quelques data en peu de temps), on va lancer aireplay, un injecteur de paquets pour accélérer le trafic et surtout stimuler les IVs.

- On va en premier lieu tester l'association avec le point d'accès avec une attaque "-1" dite de fake authentication. Cette étape n'est pas indispensable, elle peut servir à tester si le point d'accès possède un filtrage d'adresse mac. La syntaxe est la suivante :

```
aireplay-ng -1 0 -e ESSID -a @mac_R -h @mac_M-2 interface
```

- "-1 0" -1 indique une fake authentication et 0 indique le temps à laisser entre 2 tentatives (ici nul).
- "-e ESSID" ici il faut remplacer ESSID par le nom du réseau colonne ESSID.
- "-a adresse-mac-de-R" colonne BSSID.
- "-h adresse-mac-de-M2" colonne STATION.
- "*interface*" à remplacer par le nom de votre interface (rausb0, ath1 ...)
- En deuxième lieu, on va faire des injections de paquets. Elle est la clé pour réussir cette crack wep rapidement. En effet il est nécessaire de capturer beaucoup de IVs pour trouver la clé wep. L'attaque la plus prolifique pour générer des IVs est l'attaque "-3" dite de réinjection d'ARP :

```
aireplay-ng -3 -e ESSID -b @mac_R -h @mac_M-2 interface
```

Aireplay nous sauvegarde donc les arp capturés dans un fichier qu'il crée à chaque fois qu'il est lancé. Ce fichier se trouve dans le répertoire à partir duquel nous avons lancé airplay.

Phase 5 Sachant qu'il faut environ 300 000 IVs pour cracker une clé wep 64bits et environ de 1 000 000 pour une clé wep 128, cette phase est la plus importante de tout le processus du crackage.

On lance aircrack-ng par

```
aircrack-ng -x *.cap *.ivs
```

Il nous affiche tous les réseaux qu'il a rencontrés, leur cryptage et le nombre de IVs correspondant (voir chap4-6). Il nous suffit que d'entrer le numéro de notre réseau et on lance aircrack il commence à cracker la clé wep. Et après une attente, on a obtenu le mot de passe décrypté << Decrypted correctly : 100% >>

```

Aircrack-ng 1.0 rci r1085

[00:00:00] Tested 31 keys (got 40599 IVs)

KB    depth  byte(vote)
0     0/ 1    61(56172) 6C(50224) 64(48840) 15(48740) 81(48156)
1     0/ 1    7A(52948) 87(49168) 1F(48948) 6D(48104) A6(48036)
2     1/ 4    65(50156) 89(49168) 6F(48156) A1(47416) 17(46348)
3     0/ 1    72(54012) C0(48176) 7A(47644) CB(47584) A5(47528)
4     6/ 8    F5(46600) 26(46276) 2A(45884) 03(45880) 0A(45872)

KEY FOUND! [ 61:7A:65:72:74 ] (ASCII: azert)
Decrypted correctly: 100%

bt ~ #

```

Figure IV.7 – Résultat obtenu par Aircrack-ng

Phase 6 Cette phase n'est qu'une test du clé crypté s'il est la bonne ou non. Alors ce n'est qu'une demande à rejoindre le réseau et d'utilisé les ressource disponible : Accès Internet, Partage de fichier...

5 Conclusion

Conclusion générale et perspectives

Dans le cadre de ce projet qui nous a été très bénéfique et après une première expérience dans un domaine vaste et complexe de la trahison et la sécurité des réseaux, nous sommes arrivés à effectuer un scan de vulnérabilité, une analyse d'un réseau et une intrusion.

Ce sujet a été motivant et passionnant. Il nous a permis d'approfondir nos connaissances pour tout ce qui concerne la sécurité informatique. En effet les nouveaux logiciels ne sont pas toujours au point et nous avons dû beaucoup accéder aux forums de discussion pour résoudre des bugs. Nous avons appris à ne pas nous décourager et à contourner les difficultés et aller jusqu'au bout de soi.

Comme perspective d'évolution pour ce projet, nous recommandons d'effectuer ces tests dans un environnement sécurisé.

Conclusion générale et perspectives

Annexe A : Glossaire

ACK	Signale que le paquet est un accusé de réception (ACKnowledgement)
ARP	Address Resolution Protocol, protocole de résolution d'adresse
ASCII	Code Americain Standard pour l'Echange d'Informations
ASP	Active Server Pages, est un ensemble de logiciels développés par Microsoft et utilisés dans la programmation Web
Auditor	
Cookie	est défini par le protocole de communication HTTP comme étant une suite d'informations envoyée par un serveur HTTP à un client HTTP
démon	un processus chargé d'une mission (par exemple gérer un périphérique ou le réseau)
DHCP	Dynamic Host Configuration Protocol
DHCP-Server	Dynamic Host Configuration Protocol Serveur
DNS	Domain Name System (ou DNS, système de noms de domaine) est un service permettant d'établir une correspondance entre une adresse IP et un nom de domaine
Ethernet	est un protocole de réseau local à commutation de paquets
Firewall	pare-feu , protégeant des intrusions dans un réseau
Fuzzers	est une technique pour tester des logiciels par boîte noire
GUI	Interface graphique
hackers	utilisé pour désigner en informatique les programmeurs astucieux et débrouillards
Host-Only	mode de connexion permet la communication entre les machines virtuelles et l'hôte de virtualisation uniquement, rien ne sort sur le réseau physique.
HTTP	HyperText Transfer Protocol
HTTPS	HyperText Transfer Protocol secured

Annexe A : Glossaire

ICMP	Internet Control Message Protocol
IDS	Intrusion Detection System
IMAP	Internet Message Access Protocol
IP	Internet Protocol
ISS	Internet Security System
IV	Initialization Vector
LAN	Local Area Network. Réseaux locaux. Réseau situé dans une zone réduite ou dans un environnement commun, tels qu'un immeuble ou un bloc d'immeubles.
Mode-promiscuous	se réfère à une configuration de la carte réseau, qui permet à celle-ci d'accepter tous les paquets qu'elle reçoit, même si ceux-ci ne lui sont pas adressés.
NIDS	Network Intrusion Detection System
open-source	code source libre, s'applique aux logiciels dont la licence respecte des critères précisément établis par l'Open Source Initiative
Perl	un langage de programmation, reprenant des fonctionnalités du langage C et des langages de scripts sed, awk et shell (sh).
PHP	Hypertext Preprocessor, est un langage de scripts libre principalement utilisé pour produire des pages Web dynamiques via un serveur HTTP
POP	Post Office Protocol, est un protocole qui permet de récupérer les courriers électroniques situés sur un serveur de messagerie électronique
Rétro-ingénierie	
RFC	Requests For Comments, sont une série numérotée de documents officiels décrivant les aspects techniques d'Internet.
RIP	
rlogin	
SNMP	Simple Network Management Protocol, est un protocole de communication qui permet aux administrateurs réseau de gérer les équipements du réseau
SQL	Structured Query Language, un langage informatique normalisé qui sert à demander des opérations sur des bases de données.
SSH	Secure SHell. Shell permettant de se connecter de façon sécurisée sur une machine distante et d'y exécuter des programmes, toujours de façon sécurisée
SYN	Demande de SYNchronisation ou établissement de connexion

TCP	Transmission Control Protocol, est un protocole de transport fiable, en mode connecté
telnet	TErминаl NETwork ou TELEcommunication NETwork, ou encore TELetype NETwork, est un protocole réseau utilisé sur tout réseau supportant le protocole TCP/IP
UDP	User Datagram Protocol, est un des principaux protocoles de télécommunication utilisés par Internet
VM	Machine Virtuelle
vSwitch	Routeur Virtuelle
WEP	Wired Equivalent Privacy, est un protocole pour sécuriser les réseaux sans fil de type Wi-Fi.
Whois	contraction de l'anglais "who is ?", est un service de recherche fourni par les registres Internet
Wifi	Wireless Fidelity, est une technologie qui permet de relier sans fil plusieurs appareils informatiques au sein d'un réseau informatique. Cette technologie est régie par le groupe de normes IEEE 802.11
WPA	WiFi Protected Access est une solution de sécurisation de réseau WiFi

Annexe A : Glossaire

Bibliographie

- [1] Comment Ça Marche. *Introduction à la sécurité informatique*. Disponible sur <http://www.commentcamarche.net/contents/secu/securite-mise-en-oeuvre.php3> (consulté le 02 Mai 2010) 3
- [2] L'encyclopédie Wikipédia. *Vulnérabilité (informatique)*. Disponible sur <http://wapedia.mobi/fr/Vulnérabilité> (consulté le 02 Mai 2010) 4
- [3] CEDERIC L., LAURENT L. *Tableaux de bord de la sécurité réseau*. Paris, France : Eyrolles, 2004, 1-12p.(Bibliothèque de l'Académie Militaire) ISBN : 2-212-11973-9 5
- [4] Comment Ça Marche. *Les scanners de vulnérabilités - Balayage de ports*. Disponible sur <http://www.commentcamarche.net/contents/attaques/sniffers>(consulté le 03 Mai 2010) 6, 11
- [5] Sectools. *Top 11 Packet Sniffers*. Disponible sur <http://www.sectools.org/>(consulté le 06 Mai 2010) 12
- [6] Comité consultatif sur les technologies de l'information de l'ISICCA. *Test d'intrusion - Outil d'appréciation des risques pour la sécurité de l'information*. Canada : L'Institut Canadien des Comptables Agréés, 2003. Disponible sur (consulté le 17 Mai 2010) 17
- [7] Comment Ça Marche. *Audits de vulnérabilité*. Disponible sur <http://www.commentcamarche.net/contents/secu/audit-vulnérabilité>(consulté le 18 Mai 2010) 20

Bibliographie

- [8] L'encyclopédie Wikipédia. *BackTrack*. Disponible sur <http://fr.wikipedia.org/wiki/BackTrack> (consulté le 19 Mai 2010) 26