



REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE
LA RECHERCHE SCIENTIFIQUE



UNIVERSITE MOULOUD MAMMERI DE TIZI-OUZOU

FACULTE DE GENIE ELECTRIQUE ET INFORMATIQUE

DEPARTEMENT D'ELECTRONIQUE

Mémoire de Fin d'Etude de Master Académique

Spécialité : Réseaux & Télécommunications

Thème :

*Mise en place d'une solution de
détection d'intrusion*

Les membres de jury :

M^R LAZRI.M

Maitre de conférences A UMMTO, encadreur

M^R OUALLOUCHE.F

Maitre de conférences B UMMTO, président

M^R ALLOUACHE

Grade, UMMTO, examinateur

Présenté par :

M^{elle} BEN BRAHIM EMBARKA.

M^{elle} AMICHE SELYNA.



Remerciements

Au terme de ce projet de fin d'étude, nous adressons nos sincères remerciements à Monsieur LAZRI, notre encadreur pour ses directives précieuses, et pour la qualité de son suivi durant le travail effectué pour ce projet.

Nos remerciements s'adressent également à l'administration et aux ingénieurs de centre de calcul de Hasnaoua pour les moyens qu'ils ont mis à notre disposition afin d'élaborer ce travail.

Nous souhaitons exprimer nos gratitude et nos vifs remerciements à nos familles pour leurs soutiens

Nous voulons exprimer nos reconnaissances envers les amis et collègues qui nous ont apporté leurs supports moraux tout au long de notre travail.

Pour finir, nous remercions les membres de jury qui ont accepté d'évaluer notre projet, nous leurs présentons toute nos gratitude et nos profonds respects.

Dédicaces

Je dédie ce modeste travail

A mes très chers parents, pour leurs patiences, leurs sacrifices, leurs tendresses et soutiens durant mes études, aucun hommage ne pourra être à la hauteur de l'amour dont ils ne cessent de me combler.

Que dieu leurs procure bonne santé et longue vie.

A mes très chers frères et à mes chères sœurs, pour leurs amours et compréhension.

A mes chers grands parents que dieu les accueillent dans son vaste paradis, ainsi que toute ma famille paternelle et maternelle

A tous mes très chers amis

Que dieu les protège

Selyna

Dédicaces

Je dédie ce modeste travail

A mes très chers parents, pour leurs patiences, leurs sacrifices, leurs tendresses et soutiens durant mes études, aucun hommage ne pourra être à la hauteur de l'amour dont ils ne cessent de me combler.

Que dieu leurs procure bonne santé et longue vie.

A mes très chères sœurs qui ont été toujours un très bon exemple pour moi et à mes beaux-frères pour leurs amours et compréhensions.

A mon ange, mon petit frère adoré YACINE.

A ma très chère grand-mère qui prie toujours pour mon succès et à mes chers grands-parents que dieu les accueillent dans son vaste paradis.

A tous mes très chers amis et toutes personnes qui nous ont aidés.

Que dieu les protège tous.

Embarka.

Sommaire

Introduction	1
--------------------	---

CHAPITRE I : NOTIONS GÉNÉRALES SUR LES RESEAUX INFORMATIQUES

Préambule	3
1. Réseaux informatiques	4
2. La classification des réseaux	4
2.1 Classification selon l'étendue géographique	5
2.2 Classification selon l'architecture	5
2.2.1 Réseau poste à poste (PEER TO PEER)	5
2.2.2 Réseau client / serveur	6
2.3 Classification selon la topologie	7
2.3.1 Réseaux en étoile	7
2.3.2 Réseaux en anneau	8
2.3.3 Réseaux en bus	8
3. Modèle OSI	9
3.1 Caractérisation des différentes couches	10
4. TCP /IP	11
5. Encapsulation des données	13
6. Adressage	14
7. Le routage IP	16
7.1 Les deux modes de routages	16
7.1.1 Routage statique	17
7.1.2 Routage dynamique	17
8. Les protocoles de communication:	17
8.1 Le protocole ARP	17
8.2 Le protocole DHCP	18
8.3 Le protocole FTP	18
8.4 Le protocole HTTP	18
8.5 Le protocole RIP	18
8.6 Le protocole TELNET	19
8.7 Le protocole DNS	19
8.8 Le protocole UDP	19

Sommaire

9. Les équipements réseaux	19
9.1 Le serveur	20
9.2 La carte réseau.....	20
9.3 Le hub.....	20
9.4 Le switch	20
9.5 Le routeur	20
9.6 La passerelle	20
9.7 Le répéteur.....	20
9.8 Le pont.....	20
Discussion:.....	21

CHAPITRE II : SECURITEDES RESEAUX INFORMATIQUES

Préambule	23
1. Définition de la sécurité informatique.....	23
2 Les critères de la sécurité informatique.....	23
3. La politique de sécurité	24
3.1. Les types de politique de sécurité.....	24
3.1.1.La politique qui autorise tout par défaut.....	24
3.1.2.La politique qui interdit tout par défaut.....	24
4. Les menaces.....	25
4.1. Les types de menaces	25
4.1..1.Menaces accidentelles	25
4.1..2.Menaces intentionnelles	25
5. Les types d'attaques	26
5.1. Les attaques directes	26
5.2. Les attaques indirectes	26
5.2.1. Les attaques indirectes par rebond	26
5.2.2. Les attaques indirectes par réponse	27
6. Les techniques d'attaques	27
6.1.Attaques applicatives	27
6.2. Attaques réseaux	29
7. Les contres-mesures de différentes attaques	30

Sommaire

7.1. Antivirus	30
7.2. Les mécanismes de chiffrement	31
7.3. Pare feu	32
7.4. DMZ.....	33
7.5. Le proxy	34
7.6. Un certificat électronique	35
7.7. La signature électronique	36
8. Les protocoles de sécurité	36
9. VPN.....	38
Discussion	39

CHAPITRE III : SYSTEME DE DETECTION D'INTRUSION

Préambule	41
1. Définition d'un système d'intrusion	41
2. Les types des systèmes de détection d'intrusion	41
2.1. Les NIDS	42
2.2. Les HIDS	43
2.3. IDS hybride	44
3. Caractéristiques d'un système de détection d'intrusion	44
4. L'architecture d'un IDS	45
4.1. Capteur	45
4.2. Analyseur.....	45
4.3. Manager.....	45
5. Mise en place d'un IDS	46
6. Mode de fonctionnement	47
6.1. Mode de détection	47
6.1.1. La détection d'anomalies	47
6.1.2. La reconnaissance de signatures	48
6.2. Réponses actives et passives	49
6.2.1. Réponse passive	49
6.2.2. Réponse active	49

Sommaire

7. Les méthodes de détections	50
7.1. L'approche par scénario (mésuse détection)	50
7.2. L'approche comportementale (Anomalie Détection)	52
8. Détecter un IDS	53
9. Critères de tests d'un IDS	54
10. Techniques anti-IDS	55
11. Les limites de l'IDS	55
12. Les systèmes de prévention d'intrusion IPS	56
12.1. Système de prévention des intrusions réseaux (NIPS)	56
12.2. Système de prévention des intrusions sur hôte (HIPS)	57
Discussion	57

CHAPITRE IV : Application de système de détection

Préambule	59
1. Partie1 : Aperçu sur l'IDS réseau snort.....	59
1.1. Définition du snort.....	59
1.2. Architecture de snort	60
1.2.1 Décodeur de paquets.....	60
1.2.2 Préprocesseurs	61
1.2.3 moteur de détection	61
1.2. 4 système d'alerte et d'enregistrement des logs	62
1.3. La raison du choix du snort	62
2. Partie2 : Test d'intrusion	63
2.1. Objectif du test d'intrusion.....	63
2.2. Les outils utilisés pour effectuer cet IDS réseau	63
2.3. Installation et configuration du snort.....	63
2.3.1. Installation des pré-requis.....	63
2.3.2. Installation des dépendances	64
2.3.3. Installation du snort	65
2.3.4. Installation du snort MYSQL	69
2.3.5. Configuration du snort.....	74
2.3.6. Création de règle.....	75
2.4. Installation et configuration de BASE.....	76

Sommaire

2.5. Test de fonctionnement.....	78
2.6. Résultat analysés par BASE.....	82
Discussion	83
Conclusion.....	85
Bibliographie.....	87
Annexe.....	89

Liste des figures

Chapitre I: Notions générales sur les réseaux informatiques

Figure I.1: Les différents réseaux	5
Figure I.2: Réseau poste à poste	6
Figure I.3: Réseau client/serveur	6
Figure I.4: Architecture en étoile	7
Figure I.5: Architecture en anneau	8
Figure I.6: Architecture en bus	9
Figure I.7: Modèle OSI	9
Figure I.8: Standard du modèle TCP	12
Figure I.9: Encapsulation des données	14
Figure I.10: Les cinq classes d'adresses IP	15

Chapitre II : sécurité des réseaux informatique

Figure II.1: L'attaque directe	26
Figure II.2: Attaque indirecte par rebond	27
Figure II.3: Attaque indirecte par réponse.....	27
Figure II.4: Chiffrement symétrique.....	31
Figure II.5: Chiffrement asymétrique.....	32
Figure II.6: Fonctionnement d'un Pare feu	33
Figure II.7: VPN	38

Chapitre III : système de détection d'intrusion

Figure III.1: Système de détection d'intrusion réseau.....	42
Figure III.2: Système de détection d'intrusion hôte.....	43
Figure III.3: Architecture d'un IDS	45
Figure III.4: La position des IDS	46

Chapitre IV : application de système de détection d'intrusion

Figure IV.1: Architecture du snort.....	60
Figure IV.2: Le décodeur de paquets.....	61
Figure IV.3: Une partie du fichier snort	62
Figure IV.4: Choisir une interface réseau par défaut	66
Figure IV.5: Choisir le mode de démarrage du snort.....	66
Figure IV.6: Récupération d'adresse IP par défaut	67
Figure IV.7: Le paramétrage du mode.....	67
Figure IV.8: Suggestion d'ajout d'autres options pour snort.....	68
Figure IV.9: Le choix de recevoir des alertes par e-mail	68
Figure IV.10: Choisir une interface par défaut	69
Figure IV.11: Entrer l'adresse IP du serveur	70
Figure IV.12: Paramétrage du mode.....	70
Figure IV.13: Le choix de recevoir des alertes par e-mail	71
Figure IV.14: Vérification de la base de données	71
Figure IV.15: Donner un nom à la machine serveur	72
Figure IV.16: Donner un nom à la base de données	73
Figure IV.17: Donner un nom à l'utilisateur	73
Figure IV.18: Accorder un mot de passe lors de connexion	73
Figure IV.19: Fin de configuration du MYSQL.....	74
Figure IV.20: Remplissage du fichier local.rules.....	75
Figure IV.21: Création de la base de données	75
Figure IV.22: Activation de la base de données	76
Figure IV.23: Modification du fichier acide base	76
Figure IV.24: Accéder à l'interface graphique d'acide base	77
Figure IV.25: Affichage des paramètres configurés	77
Figure IV.26: Connexion à base autant qu'utilisateur.....	78
Figure IV.27: Démarrage du service de la machine	78
Figure IV.28: Lancement du snort.....	79
Figure IV.29: Récupération d'adresse IP du serveur	80
Figure IV.30: Lancement d'une requête ARP	80
Figure IV.31: Etablir la connexion entre les deux machines	81

Figure IV.32: Lancement d'une requête ICMP	81
Figure IV.33: Recevoir une alerte ICMP.....	82
Figure IV.34: Analyse des requêtes TCP et ICMP	82
Figure IV.35: Gestion des alertes	83

Liste des abréviations

SNORT: The Open Source Network Intrusion Detection System.

DMZ: Demilitarized Zone.

IDS: Intrusion Detection System.

IPS: Intrusion Prevention System.

VPN: Virtual Private Network.

DNS: Domain Name Service.

DOS: Denial of Service.

NIDS: Network Intrusion Detection System.

NIPS: Network Intrusion Prevention System.

HIDS: Host based Intrusion Detection System.

HIPS: Host based Intrusion Prevention System.

HTML: HyperText Markup Language.

HTTP: HyperText Transfer Protocol.

IDS: Intrusion Detection System.

INTERNET: Interconnected Network.

IP: Internet Protocol.

IPS: Intrusion Prevention System.

ISO: International Standards Organization.

LAN: Local Area Network.

MySQL: My Structured Query Language.

PDU: Protocol Data Unit.

TCP: Transmission Control Protocol.

TELNET: Terminal Network.

FTP: File Transfer Protocol.

UDP: User Datagram Protocol.

WWW: World Wide Web.

Introduction

Les réseaux informatiques sont devenus des ressources vitales et déterministes pour le bon fonctionnement des entreprises. De plus, ces réseaux sont ouverts de fait qu'ils sont pour la plus part raccordés à l'Internet.

Cette ouverture qui permet de faciliter la communication, engendre malheureusement des risques importants dans le domaine de la sécurité informatique. Les utilisateurs de l'Internet ne sont pas forcements pleins de bonnes intentions, ils peuvent exploiter les vulnérabilités des réseaux et systèmes pour réaliser leurs attaques.

La sécurité informatique consiste d'une manière générale à assurer que les ressources matérielles ou logicielles d'une organisation soient uniquement utilisées dans le cadre du réseau informatique, plusieurs formes d'attaques ont été récemment recensées avec la prolifération de la technologie. De nombreuses entreprises sont actuellement victimes des attaques d'intrusion au niveau de leur système et de base de données. C'est un problème majeur que des intrus aient accès aux données confidentielle, ceci peut conduire jusqu'à la faillite de l'entreprise et doit être résolu. Les attaques proviennent notamment depuis l'intérieur ainsi que de l'extérieur (Internet). Pour cela, les administrateurs déploient des solutions de sécurité efficace capable de protéger le réseau de l'entreprise. Dans le contexte, les IDS constituent une bonne alternative pour mieux protéger le réseau informatique.

Cette technologie consiste à rechercher une suite de mots ou de paramètres caractérisant une attaque dans un flux de paquets. Un IDS doit être conçu dans une politique globale de sécurité. Son objectif est de détecter toute violation liée à la politique de sécurité, il permet ainsi de signaler les attaques. Une solution efficace doit être mise en place, d'où la mise en place d'un système de détection d'intrusion dont le nom est Snort, c'est l'objet de ce mémoire de fin d'étude.

Dans le cas du système SNORT, la reconnaissance des attaques est basée sur le concept d'analyse de chaînes de caractères présentes dans le paquet. Pour que le système puisse être capable de détecter une attaque, cette dernière doit être décrite par une signature. C'est avec cette signature que nous pourrions écrire la règle que l'IDS va utiliser pour la détection

Nous avons structuré notre mémoire en quatre chapitres :

- Dans le premier chapitre, nous présentons une étude générale des réseaux informatiques.
- Dans le deuxième chapitre, nous abordons la sécurité informatique qui comprend les différentes attaques ainsi les mécanismes de sécurité mis-en-place.
- Le troisième chapitre est consacré à l'étude des systèmes de détection d'intrusion, ses différentes variantes et ses méthodes de détection.
- Le dernier chapitre consiste en la mise en place de Snort qui est un système open source pour réaliser la détection d'intrusion. Nous allons commencer par donner une présentation générale de SNORT, ensuite nous allons présenter sa manipulation : installation, configuration et pour finir les fonctionnalités.

Préambule

Le réseau est un ensemble d'objets connectés ou maintenus en liaison. Auparavant les communications entre différentes machines étaient juste destinées aux transports de données informatiques alors qu'aujourd'hui les réseaux permettent le partage des ressources qui intègrent d'autre type de données tels que la parole et la vidéo.

Dans ce premier chapitre, nous décrivons les notions théoriques de base sur les réseaux informatiques en général. Pour ce faire, dans un premier temps, nous présentons les types de réseau informatique, ensuite nous donnons un aperçu sur les différentes couches du modèle OSI. En fin, nous citons les protocoles de communication amenés à faire le routage des données entre les réseaux, ainsi les équipements utilisés.

1. Réseaux informatiques :

Un réseau informatique est un ensemble d'ordinateur (ou d'équipement informatique) qui sont interconnectés entre eux à travers des protocoles de communication normalisés. Il sert à l'échange de données numériques et le partage de ressources (Imprimantes, disques, ...) entre systèmes et applications informatiques tels que les traitements de textes, ou les navigateurs Web [2].

2. La classification des réseaux:

Les réseaux informatiques peuvent être divisés en plusieurs types : selon leurs étendues, leurs architectures et leurs topologies [2]. (Voir figure I.1).

2.1. Classification selon l'étendue géographique :

- **Un réseau personnel (PAN: Personnal Area Network)** interconnecte (souvent par des liaisons sans fil) des équipements personnels comme un ordinateur portable, un agenda électronique... ; il est souvent mis en œuvre dans un espace d'une dizaine de mètres.

- **Un réseau local (LAN: Local Area Network)** peut s'étendre de quelques mètres à quelques kilomètres et correspond au réseau d'une entreprise. Il peut se développer sur plusieurs bâtiments.
- **Un réseau métropolitain (MAN: Metropolitan Area Network)** interconnecte plusieurs lieux situés dans une même ville, par exemple les différents sites d'une université ou d'une administration, chacun possédant son propre réseau local.
- **Un réseau étendu (WAN: Wide Area Network)** permet de communiquer à l'échelle d'un pays ou de la planète entière, les infrastructures physiques pouvant être terrestres ou spatiales à l'aide de satellites de télécommunications.

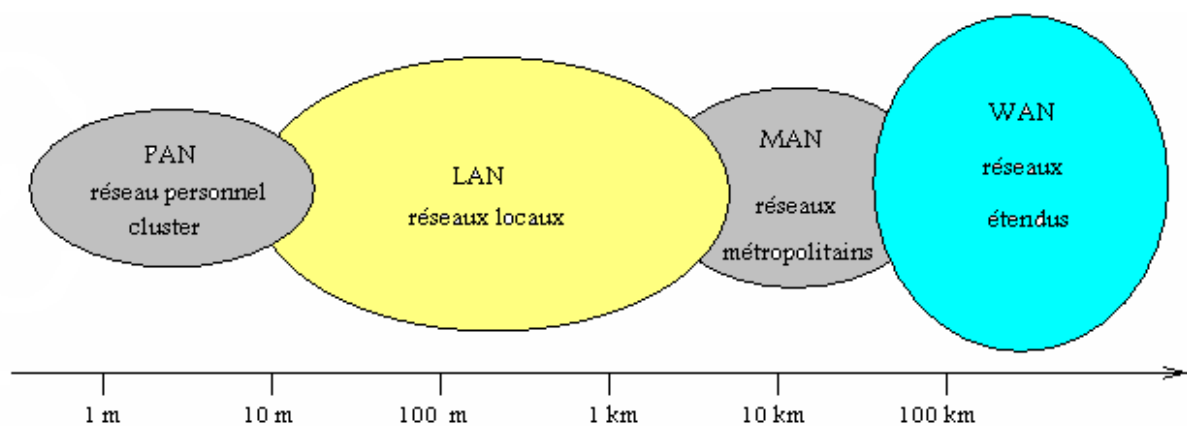


Figure I.1: Les différents réseaux.

2.2. Classification selon l'architecture:

On distingue généralement deux types de réseaux locaux :

2.2.1. Réseau poste à poste (peer to peer) :

Les ordinateurs sont reliés par un support physique, chaque poste connecté peut mettre ses données et ses ressources à disposition du réseau, il peut être à la fois client et serveur; il est plus adapté aux petites structures.

Sur un réseau poste à poste, les ressources telles que les imprimantes, les fax et les modems sont généralement connectés à un même ordinateur qui les partage avec les autres ordinateurs du réseau. (Voir figure I.2).

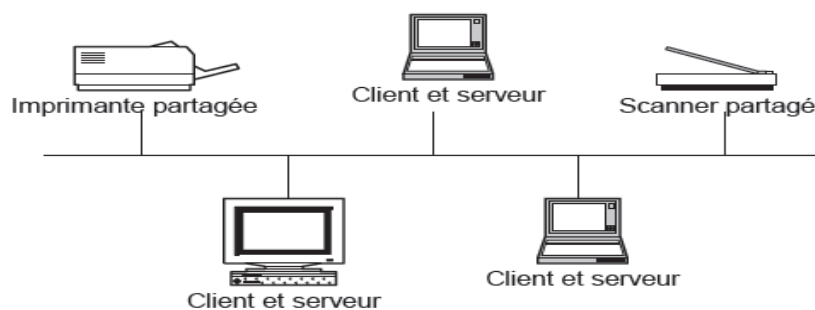


Figure I.2 : Réseau poste à poste.

Avec ce type de réseau, chaque utilisateur doit apprendre à administrer son propre ordinateur pour permettre aux autres d'y accéder ; en plus y'a un problème de sécurité des données et la personne n'aura pas accès au réseau distant ni à la messagerie internet.

2.2.2. Réseau client / serveur :

Dans une architecture client/serveur, parmi les machines du réseau, il y a une qui est considérée comme un serveur, elle est généralement très puissante et c'est elle qui délivre les informations (tels que la connexion) aux autres ordinateurs qui sont considérés comme des postes clients. (Voir figure I.3).

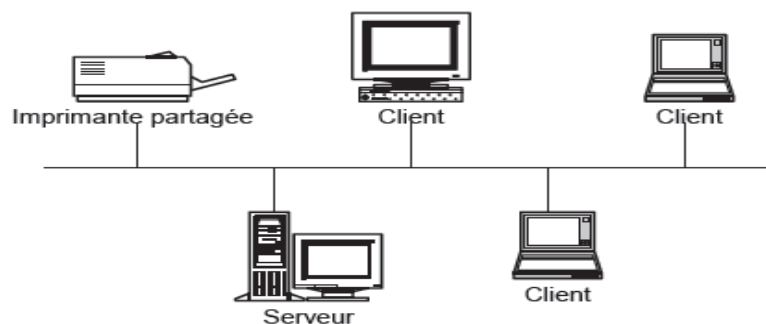


Figure I.3 : Réseau client/serveur.

L'avantage des réseaux clients/serveur :

- Les architectures client-serveur sont généralement supervisées par des administrateurs système.
- Leurs maintenance est relativement facile à assurer quand ils se trouvent dans un même local.
- C'est une architecture extensible et permet de déceler facilement les pannes, lorsque plusieurs ordinateurs présentent le même problème, c'est souvent le serveur qui est mis en cause.
- La sauvegarde des données.

2.3. Classification selon la topologie :

On distingue deux types de topologie qui sont différentes à l'utilisation [2]:

- **La topologie logique** décrit le mode de fonctionnement du réseau, la répartition des nœuds et définit la façon dont les données transitent dans les lignes de communication.
- **La topologie physique** indique comment les différentes stations sont raccordées physiquement (câblage).

La topologie des réseaux informatiques en particulier peut se partager en trois groupes :

2.3.1. Réseaux en étoile:

Dans cette topologie, chaque nœud est relié directement sur un nœud central (hubs) l'information passe d'un nœud périphérique au nœud central (un concentrateur), celui-ci doit gérer chaque liaison. (Voir figure I.4).

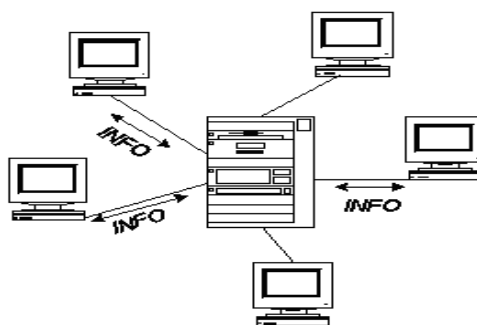


Figure I.4: Architecture en étoile.

Ce hub augmente un peu le coût du réseau mais le rend plus fiable, dans le cas d'une interruption de la connexion vers une machine, celle-ci sera la seule à être déconnectée. Le reste du réseau continue de fonctionner normalement.

2.3.2. Réseaux en anneau:

Pour cette topologie, chaque nœud est relié au nœud suivant et au nœud précédent et forme ainsi une boucle fermée (un anneau). L'information (le jeton) transite par chacun d'eux et retourne à l'expéditeur en désignant l'ordinateur qui a le droit d'émettre. Les ordinateurs s'emparent du jeton où le passe au suivant selon qu'ils ont des données à transmettre ou qu'ils peuvent passer leur tour. Cette organisation permet d'éviter les collisions. (Voir figure I.5).

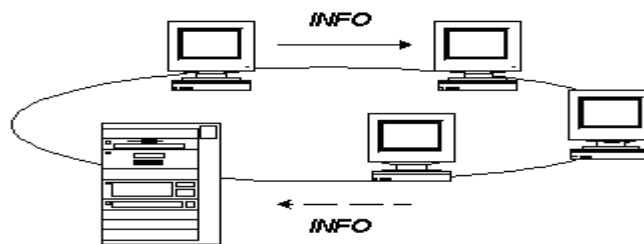


Figure I.5: Architecture en anneau.

Dans le cas d'une panne de l'une des entités actives, le trafic du réseau se paralyse et il est difficile d'insérer une nouvelle station.

2.3.3. Réseaux en bus:

Dans cette topologie, chaque nœud est connecté sur un bus (une même ligne physique), l'information passe sur chaque nœud et se termine à l'extrémité du bus. Toutes les transmissions se font donc par un seul lien sur lequel un seul ordinateur a le droit d'émettre des données à la fois. Des terminateurs ou " bouchons " sont placés aux extrémités du câble pour y amortir le signal et éviter que son écho ne vienne perturber les signaux suivants. (Voir figure I.6).

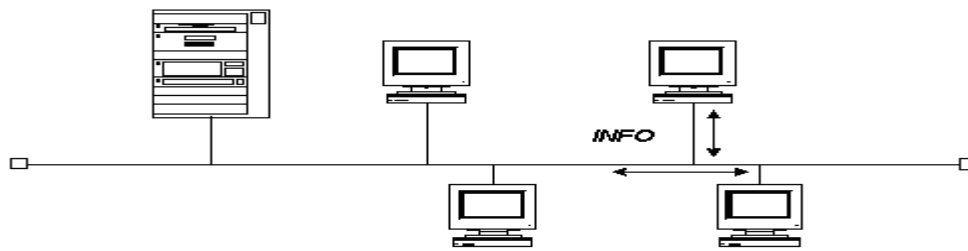


Figure I.6: Architecture en bus.

L'avantage de ce type de réseau est sa simplicité. À part le raccordement des cartes réseau des différentes machines, il n'est pas nécessaire d'utiliser d'autres équipements. L'inconvénient est que ce type de liaison est assez fragile car tout le réseau est affecté dès qu'une connexion quelconque est défectueuse.

3. Le modèle OSI :

L'OSI est un modèle de base normalisé par l'organisation internationale standardisation, il était normalisé pour travailler avec un même et seul modèle de communication une architecture réseau en utilisant ces différentes fonctions qui sont organisées en 7 couches numérotées [1].

Ces couches sont parfois réparties en deux groupes comme le tableau d'architecture nous montre ci-dessous :

	PDU	Couche	Fonction
Couches hautes	Donnée	7 Application	Point d'accès aux services réseau
		6 Présentation	Gère le chiffrement et le déchiffrement des données, convertit les données machine en données exploitables par n'importe quelle autre machine
		5 Session	Communication Interhost, gère les sessions entre les différentes applications
	Segment (en) / Datagramme	4 Transport	Connexion de bout en bout, connectabilité et contrôle de flux ; notion de port (TCP et UDP)
Couches matérielles	Paquet	3 Réseau	Détermine le parcours des données et l'adressage logique (adresse IP)
	Trame	2 Liaison	Adressage physique (adresse MAC)
	Bit	1 Physique	Transmission des signaux sous forme numérique ou analogique

Figure I.7: Modèle OSI.

Les quatre couches inférieures (1, 2, 3 et 4) sont nécessaires à l'acheminement des informations entre les extrémités concernées et dépendent du support physique.

Les trois couches supérieures (5, 6 et 7) sont responsables du traitement de l'information relative à la gestion des échanges entre systèmes informatiques. Par ailleurs, les couches 1 à 3 interviennent entre machines voisines, et non entre les machines d'extrémité qui peuvent être séparées par plusieurs routeurs. Les couches 4 à 7 sont au contraire des couches qui n'interviennent qu'entre hôtes distants.

3.1. Caractérisation des différentes couches :

Le model OSI comporte sept couches, chaque couche a des fonctions de manipulation de commandes ou de données significatives qui sont décrites et détaillées plus bas[1].

- La **couche physique** est chargée de la transmission effective des signaux entre les interlocuteurs. Son service est limité à l'émission et la réception d'un bit ou d'un train de bit continu.
- La **couche liaison de données** gère les communications entre 2 machines directement connectées entre elles, ou connectées à un équipement qui émule une connexion directe (commutateur). Un rôle important de cette couche est la détection et la correction d'erreurs intervenues sur la couche physique.
- La **couche réseau** gère les communications entre machines (sous-réseaux, routage et adressage des paquets).
- La **couche transport** c'est l'une des couches les plus importantes, car elle gère les communications de bout en bout entre processus (programmes en cours d'exécution). Cette couche est responsable du bon acheminement des messages complets au destinataire. Le rôle principal de la couche transport est de prendre les messages de la couche session, de les découper s'il le faut en unités plus petites et de les passer à la couche réseau, tout en s'assurant que les morceaux arrivent correctement de l'autre côté. Cette couche effectue donc aussi le réassemblage du message à la réception des morceaux.
- La **couche session** gère la synchronisation des échanges et les transactions, permet l'ouverture et la fermeture de session.

Elle réalise le lien entre les adresses logiques et les adresses physiques des tâches réparties. Elle établit également une liaison entre deux programmes d'application devant coopérer et commande leur dialogue

- La **couche présentation** est chargée du codage des données applicatives, précisément de la conversion entre données manipulées au niveau applicatif et chaînes d'octets effectivement transmises. Elle peut aussi reformater, crypter et compresser les données.
- La **couche application** est le point d'accès aux services réseaux, elle n'a pas de service propre spécifique et entrant dans la portée de la norme.

4. TCP /IP :

TCP/IP est l'ensemble des protocoles utilisés pour le transfert des données sur internet [1].

Elle est souvent appelée TCP/IP, d'après le nom de ses deux premiers protocoles :

TCP (Transmission Control Protocol) et IP (Internet Protocol).

La couche IP (Internet Protocol), appelée aussi la couche basse, elle s'occupe principalement de gérer l'adressage logique et d'assurer l'acheminement des paquets d'un nœud à l'autre.

La couche TCP (Transmission Control Protocol), appelée la couche haute, s'occupe de gérer les erreurs et de contrôler le flux en mettant en place des mécanismes de répétition de paquets et d'ajustage de fenêtre de réception (c'est-à-dire le nombre de paquets qu'il est possible de recevoir avant de les valider/invalider).

Le modèle TCP/IP comporte quatre couches : la couche application, la couche transport, la couche Internet et la couche d'accès au réseau (voir figure I.8). Certaines couches du modèle TCP/IP portent le même nom que des couches du modèle OSI, mais elles comportent des fonctions différentes qu'on cite au-dessous.

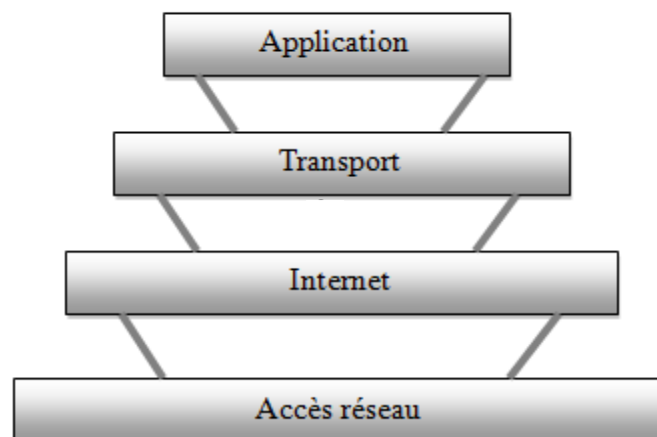


Figure I.8: Standard du modèle TCP.

➤ **Couche application :**

Elle est située au sommet des couches du protocole TCP/IP. Elle contient des applications réseau permettant de communiquer grâce aux couches inférieures.

La couche application gère les protocoles de haut niveau : représentation codage et contrôle du dialogue.

➤ **Couche transport :**

Elle assure tout d'abord une communication de bout en bout en faisant abstraction des machines intermédiaires entre l'émetteur et le destinataire.

La couche transport contient deux protocoles permettant à deux applications d'échanger des données ; dans le cas de TCP (Transmission Control Protocol), elle s'occupe de réguler le flux de données et assure un transport fiable (données transmises sans erreur et reçues dans l'ordre de leur émission) et non fiable dans le cas d'UDP (User Datagram Protocol). Il n'est pas garanti qu'un paquet (appelé dans ce cas datagramme) arrive à bon port, c'est à la couche application de s'en assurer.

➤ **Couche internet :**

Le rôle de la couche Internet consiste à envoyer des paquets source à partir d'un réseau quelconque de l'inter réseau et à les faire parvenir à destination, indépendamment du trajet et des réseaux traversés pour y arriver. Le protocole qui régit cette couche est appelé protocole IP (Internet Protocol). L'identification du meilleur chemin et la commutation de paquets ont lieu au niveau de cette couche.

➤ **Couche accès réseau :**

C'est la première couche de la pile TCP/IP, elle se charge de tout ce dont un paquet IP a besoin pour établir une liaison physique. Elle prend en charge les notions suivantes :

- Acheminement des données sur la liaison.
- Coordination de la transmission de données (synchronisation).
- Conversion des signaux (analogique-numérique).
- Contrôle des erreurs à l'arrivée.

5. Encapsulation des données :

Lors d'une transmission, les données traversent chacune des couches au niveau de la machine émettrice. A chaque couche, une information est ajoutée au paquet de données, il s'agit d'une entête, ensemble d'informations qui garantissent la transmission. Au niveau de la machine réceptrice, lors du passage dans chaque couche, l'entête est lu, puis supprimé. Ainsi, à la réception, le message est dans son état original [2]. (Voir figure I.9).

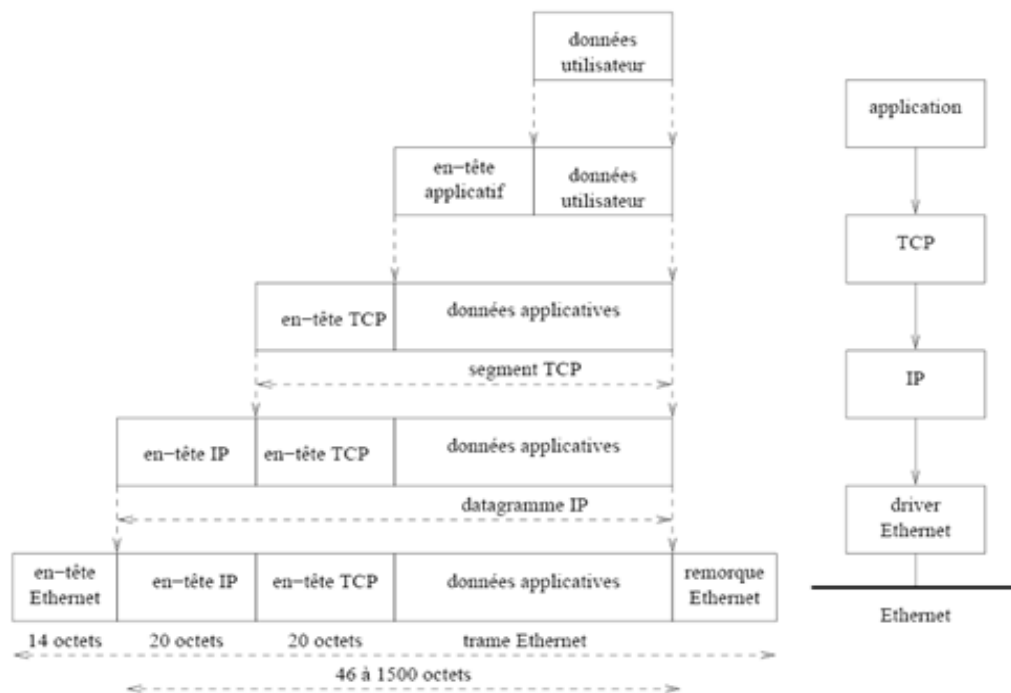


Figure I.9: Encapsulation des données.

6. Adressage :

Chaque ordinateur dans un réseau possède une adresse unique sur 32 bits. Plus précisément, chaque interface dispose d'une adresse IP particulière [2].

En effet, une adresse IP est toujours représentée dans une notation décimale constituée de 4 octets (0–155) et séparés par un point.

Une adresse IP est désignée par un identifiant de réseau et un identifiant de machine et appartient à l'une des classes (A, B, C, D ou E) selon la valeur de son premier octet, comme le représente la figure suivante.

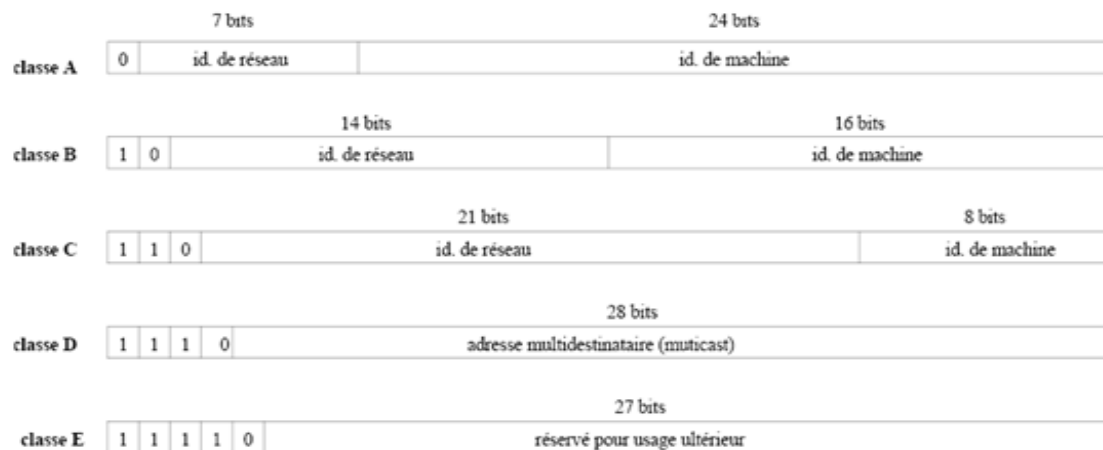


Figure I.10 : Les cinq classes d'adresses IP.

Le tableau ci-après donne l'espace d'adresses possibles pour chaque classe :

classe	adresses
A	0.0.0.0 à 127.255.255.255
B	128.0.0.0 à 191.255.255.255
C	192.0.0.0 à 223.255.255.255
D	224.0.0.0 à 239.255.255.255
E	240.0.0.0 à 247.255.255.255

Tableau : L'espace d'adresse.

On trouve des adresses particulières qui sont des adresses à ne pas utiliser comme :

- **0.0.0.0** est utilisée par une machine pour connaître sa propre adresse IP
- **<partie réseau>.<partie machine nulle>** n'est jamais affectée à une machine car elle permet de désigner le réseau lui-même
- **<partie réseau>.<partie machine avec tous ses bits à 1>** est une adresse de diffusion ou de broadcasting (toutes les machines du réseau concerné).
- **255.255.255.255** est une adresse de diffusion locale car elle désigne toutes les machines du réseau.
- **127.X.Y.Z** est une adresse de rebouclage qui est utilisée pour permettre les communications inter-processus sur un même ordinateur ou réaliser des tests de logiciels.

7. Le routage IP :

C'est une fonction qui permet de déterminer le meilleur chemin dans un réseau maillé vers une destination identifiée par une adresse IP [1].

Le routage est réalisé par trois fonctions :

- **Le relaying (forwarding) :** il calcule le port de sortie en analysant l'adresse destination du paquet et après avoir consulté une table de routage effectuée a celui-ci.
- **La commutation (switching) :** transfert du ou des fragments de paquet d'un port d'entrée vers un port de sortie à travers un bus.
- **L'ordonnancement (Scheduling) :** détermination de l'ordre d'émission des paquets sur la liaison de sortie.

On divise le routage en deux grandes familles :

➤ **Le routage direct :**

- Il s'agit de délivrer un datagramme à une machine raccordée au même LAN.
- L'émetteur trouve l'adresse physique du correspondant (ARP), encapsule le datagramme dans une trame et l'envoie.

➤ **Le routage indirect :**

- Le destinataire n'est pas sur le même LAN comme précédemment. Il est absolument nécessaire de franchir une passerelle connue d'avance ou d'employer un chemin par défaut.
- En effet, toutes les machines à atteindre ne sont pas forcément sur le même réseau physique. C'est le cas le plus courant, par exemple sur l'Internet qui regroupe des centaines de milliers de réseaux différents.
- Cette opération est beaucoup plus délicate que la précédente, car il faut sélectionner une passerelle.

7.1. Les deux modes de routages :

Il existe deux modes de routages bien distincts lorsque nous souhaitons aborder la mise en place d'un protocole de routage, il s'agit du routage statique et du routage dynamique.

7.1.1. Routage statique :

Dans le routage statique, les administrateurs vont configurer les routeurs un à un au sein du réseau afin d'y saisir les routes (par l'intermédiaire de port de sortie ou d'IP de destination) à emprunter pour aller sur tel ou tel réseau. Concrètement, un routeur sera un pont entre deux réseaux et le routeur d'après sera un autre pont entre deux autres réseaux.

7.1.2. Routage dynamique :

Le routage dynamique permet quant à lui de se mettre à jour de façon automatique. La définition d'un protocole de routage va permettre au routeur de se comprendre et d'échanger des informations de façon périodique ou événementielle afin que chaque routeur soit au courant des évolutions du réseau sans intervention manuelle de l'administrateur du réseau. Concrètement, le protocole de routage fixe la façon dont les routeurs vont communiquer mais également la façon dont ils vont calculer les meilleures routes à emprunter.

8. Les protocoles de communication:

Un protocole réseau est un ensemble de règles et de procédures de communication utilisées d'une part et d'autre par toutes les stations qui échangent des données sur le réseau [1].

Il existe de nombreux protocoles réseaux, mais ils n'ont pas tous, ni le même rôle, ni la même façon de procéder. Certains protocoles réseaux fonctionnent au niveau de plusieurs couches du modèle OSI, d'autres peuvent être spécialisés dans la réalisation d'une tâche correspondant à une seule couche du modèle OSI. Un paquet transmis sur le réseau est constitué de plusieurs couches d'informations correspondant aux différents traitements de chacun des protocoles de la pile.

8.1. Le protocole ARP :

Address Resolution Protocol est un protocole effectuant la traduction d'une adresse de protocole de couche réseau (typiquement une adresse IPv4) en une adresse MAC (typiquement une adresse Ethernet), ou même de tout matériel de couche de liaison.

Il se situe à l'interface entre la couche réseau (couche 3 du modèle OSI) et la couche de liaison (couche 2 du modèle OSI).

8.2. Le protocole DHCP :

Dynamic Host Configuration Protocol est un protocole réseau dont le rôle est d'assurer la configuration automatique des paramètres IP d'une station, notamment en lui affectant automatiquement une adresse IP et un masque de sous-réseau.

8.3. Le protocole FTP :

File Transfer Protocol (protocole de transfert de fichier), est un protocole de communication destiné au partage de fichiers sur un réseau TCP/IP. Il permet, depuis un ordinateur, de copier des fichiers vers un autre ordinateur du réseau, ou encore de supprimer ou de modifier des fichiers sur cet ordinateur.

FTP obéit à un modèle client-serveur, c'est-à-dire qu'une des deux parties, le client, envoie des requêtes auxquelles réagit l'autre, appelé serveur. En pratique, le serveur est un ordinateur sur lequel fonctionne un logiciel lui-même appelé serveur FTP.

FTP appartient à la couche application du modèle OSI ; il utilise une connexion TCP.

8.4. Le protocole HTTP :

L'HyperText Transfer Protocol, est un protocole de la couche application. Il peut fonctionner sur n'importe quelle connexion fiable, il permet l'échange de données de différents types.

Les clients HTTP les plus connus sont les navigateurs Web permettant à un utilisateur d'accéder à un serveur contenant les données.

8.5. Le protocole RIP :

Routing Information Protocol est un protocole de routage IP de type vecteur de distances. Il permet à chaque routeur de communiquer aux routeurs voisins la métrique, c'est-à-dire la distance qui les sépare d'un réseau IP déterminé.

Pour chaque réseau IP connu, chaque routeur conserve l'adresse du routeur voisin dont la métrique est la plus petite. Ces meilleures routes sont diffusées toutes les 30 secondes.

8.6. Le protocole TELNET:

Terminal network ou telecommunication network est un protocole utilisé sur tout réseau TCP/IP, permettant de communiquer avec un serveur distant en échangeant des lignes de textes et en recevant des réponses également sous forme de texte. Il appartient à la couche application du modèle OSI.

8.7. Le protocole DNS:

Domain name system (ou système de noms de domaine) est un service permettant d'établir une correspondance entre une adresse IP et un nom de domaine. Le DNS est un protocole indispensable au fonctionnement d'internet, Un nom convivial est plus facile à apprendre et à mémoriser que les adresses numériques que les ordinateurs utilisent pour communiquer sur un réseau.

Autrement dit, il permet aux utilisateurs d'ordinateurs clients d'utiliser des noms plutôt que des adresses IP numériques pour identifier les hôtes distants.

8.8. Le protocole UDP:

User Datagram Protocol (en français protocole de datagramme utilisateur) est un des principaux protocoles de télécommunication utilisés par Internet. Il fait partie de la couche transport du modèle OSI, il appartient à la couche 4, comme TCP.

Le rôle de ce protocole est de permettre la transmission de données de manière très simple entre deux entités, chacune étant définie par une adresse IP et un numéro de port. Contrairement au protocole TCP, il fonctionne sans négociation : il n'existe pas de procédure de connexion préalable à l'envoi des données. Donc UDP ne garantit pas la bonne livraison des datagrammes à destination, ni leur ordre d'arrivée. Il est également possible que des datagrammes soient reçus en plusieurs exemplaires.

9. Les équipements réseaux :

L'infrastructure d'un réseau définit d'une part les équipements qui le composent et d'autre part, les connexions entre ces éléments [1]. Elle établit donc les liaisons possibles entre les équipements et elle assure l'interconnexion des moyens physiques grâce à des protocoles de communication. Les principaux équipements d'une infrastructure sont :

9.1. Un serveur : est un ordinateur et un ensemble de logiciels dont le rôle est de répondre automatiquement à des requêtes pour fournir des services. Les serveurs sont généralement utilisés simultanément par de nombreux clients, ils peuvent être utilisés pour stocker, partager et échanger des informations.

9.2. Une carte réseau : est le dispositif qui permet de connecter un équipement (ordinateur, imprimante, etc.) à un réseau. Chaque carte dispose d'une adresse MAC adresse physique fixée par le constructeur, qui identifie de façon unique un équipement dans un réseau.

9.3. Un concentrateur ou hub : il diffuse la trame reçue vers tous les équipements connectés. Cet équipement est un simple répéteur de données, il est utilisé pour créer un réseau local de type Ethernet.

9.4. Un commutateur réseau ou switch : il diffuse la trame reçue vers l'équipement concerné car il est capable de déterminer l'adresse à laquelle cette trame est destinée. Le commutateur s'appuie sur une table des adresses MAC pour diffuser la trame vers l'équipement concerné. Le concentrateur et le commutateur réseau agissent au niveau de la couche de liaison des données du modèle OSI.

9.5. Un routeur : il assure le routage des paquets entre deux réseaux utilisant ou non le même protocole. Il doit être connecté à deux réseaux pour avoir des trames à router ; il agit au niveau de la couche réseau du protocole OSI.

9.6. Une passerelle ou gateway : est un équipement qui connecte deux réseaux informatiques hétérogènes, la passerelle agit entre les niveaux 4 et 7 du modèle OSI. La plupart du temps, la passerelle englobe des éléments tels qu'un pare-feu ou un serveur mandataire.

9.7. Un répéteur : est un équipement électronique simple permettant d'amplifier un signal et d'augmenter la taille d'un réseau. Ce n'est pas un organe intelligent capable d'apporter des fonctionnalités supplémentaires, il ne fait qu'augmenter la longueur du support physique. Il travaille uniquement au niveau physique du modèle OSI.

9.8. Un pont ou bridge : est un équipement permettant de relier des réseaux travaillant avec le même protocole. Ils travaillent au niveau logique. Sa fonction est d'interconnecter deux segments de réseaux distincts, soit de technologies différentes, soit de même technologie, mais physiquement séparés à la conception pour diverses raisons (géographique, extension de site etc.).

Discussion

Dans ce chapitre, nous avons présenté les principes de bases des réseaux informatiques (architectures, types, topologies...), aussi nous avons cité les modèles de standardisation (OSI, TCP/IP) en détails; puis les différents protocoles qui comptent dans un réseau et citer au moins un protocole qui appartient à chaque couche des deux modèles ; ainsi que le routage IP.

Pour compléter les informations, nous avons présenté les éléments du réseau qui servent au transfert d'informations (switch, hub, serveur,...).

Le développement de ce dernier a posé des conflits majeurs aux utilisateurs qui restent confrontés à une augmentation et à une complexité croissante d'intrusions et attaques informatiques dans leurs réseaux. Cependant, dans le chapitre 2, nous allons étudier les vulnérabilités et proposer des précautions qu'il faut entreprendre.

Préambule

L'échange de données dans un réseau engendre des risques important dans le domaine de la sécurité informatique, les ordinateurs connectés à un réseau externe (internet) ont des failles qui peuvent être exploité par des pirates pour réaliser leurs attaques.

Les conséquences de ces attaques peuvent être lourdes (vol d'information, accès à des informations confidentielles....). Par conséquent, il est important de prendre des mesures de sécurité afin de se protéger de ces attaques

Dans ce chapitre nous citons les différentes attaques et les moyens mit à la disposition pour sécuriser les données informatiques.

1. Définition de la sécurité informatique :

La sécurité informatique est l'ensemble des moyens mis en œuvre afin de réduire la vulnérabilité d'un système contre les menaces accidentelles ou intentionnelles.

Il convient d'identifier les exigences fondamentales en sécurité informatique. Elles caractérisent ce à quoi s'attendent les utilisateurs de systèmes informatiques en regard de la sécurité [3].

2. Les critères de la sécurité informatique :

Les objectifs d'une politique de sécurité sont de garantir la sécurité des informations et des réseaux d'entreprise. Ces impératifs peuvent être définis à plusieurs niveaux

- **L'intégrité** : consiste à déterminer si les données n'ont pas été altérées durant la communication. C'est-à-dire garantir que les données sont bien celles que l'on croit être.
- **La confidentialité** : consistant à assurer que seules les personnes autorisées aient accès aux ressources échangées.
- **La disponibilité** : L'objectif de la disponibilité est de garantir l'accès à un service ou à des ressources, permettant de maintenir le bon fonctionnement du système d'information quand les informations sont accessibles au moment voulu.

- **Non répudiation** : permettant de garantir qu'une transaction ne peut être niée. La non-répudiation de l'origine prouve que les données ont été envoyées, et la non-répudiation de l'arrivée prouve qu'elles ont été reçues.
- **L'authentification** : consistant à assurer que seules les personnes autorisées aient accès aux ressources. Elle 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 l'accès à des ressources uniquement aux personnes autorisées [3].

3. La politique de sécurité :

Une politique de sécurité ou stratégie de sécurité est une déclaration formelle des règles qui doivent être respectées par les personnes qui ont accès aux ressources et données de l'entreprise en vue de protéger son réseau contre les attaques menées soit de l'intérieur, soit de l'extérieur.

Autrement dit une politique de sécurité définit un certain nombre de règles, de procédures et une bonne pratique permettant d'assurer un niveau de sécurité conforme au besoin de l'organisation. Elle a pour objectif :

- Identifier les risques informatiques et leurs éventuelles conséquences.
- Elaborer des règles et des procédures à mettre en œuvre dans les services de l'organisation pour les risques identifiées.
- Surveiller et détecter les failles du système informatique.

3.1. Les types de politique de sécurité :

3.1.1. La politique qui autorise tout par défaut : Dans cette approche, tout ce qui n'est pas explicitement interdit est autorisé. Elle consiste à analyser les différents risques d'application qui doivent s'exécuter, en déduire les interdictions à appliquer et autoriser tout le reste.

3.1.2. La politique qui interdit tout par défaut : Dans cette approche, tout ce qui n'est pas explicitement permis est interdit. Elle consiste à définir les services à autoriser (http pour l'hôte devant accéder au web) et définir les droits de chaque utilisateur [3].

4. Les menaces:

Une menace informatique représente le type d'actions susceptibles de nuire dans l'absolu à un système informatique. En termes de sécurité informatique les menaces peuvent être le résultat de diverses actions. Ce sont des adversaires déterminés capables de monter une attaque exploitant une vulnérabilité [4].

4.1. Les types de menaces :

4.1.1 Menaces accidentelles :

Ils sont liés à des erreurs involontaires des utilisateurs du poste, perte accidentelle de données, dégradation ou destruction involontaire de matériel, copies illicites de logiciels... .

4.1.2 Menaces intentionnelles :

Une menace intentionnelle est une action exécutée par une entité pour violer la sécurité de l'information et l'utilisation non autorisée des ressources. Les menaces intentionnelles peuvent être passives ou active.

- **Menaces passives :**

Une menace passive constitue à écouter le trafic du réseau (ou de la machine) cible, donc l'interface de la machine attaquante (du pirate) est en mode écoute. L'objectif est de découvrir et capturer des trames du réseau cible pour y rechercher des informations particulières: clés de cryptages, mots de passe ou données. Elle se réalise grâce à des outils tels que: les sniffer, les scanners.

- **Menaces actives :**

Contrairement à une menace passive, ici l'attaquant n'est plus en mode écoute. Elles consistent à modifier des données ou des messages, à s'introduire dans des équipements réseau ou à perturber le bon fonctionnement de ce réseau, et aussi à interroger le réseau (ou la machine) cible, contourner le dispositif de sécurité existant par diverses méthodes tels que: Déni de Service et Virus.

5. Les types d'attaques :

Les hackers utilisent plusieurs techniques d'attaques. Ces attaques peuvent être regroupées en deux familles différentes [5]:

5.1. Les attaques directes :

C'est la plus simple des attaques, le hacker attaque directement sa victime à partir de son ordinateur. En effet, les programmes de hack qu'ils utilisent ne sont que faiblement paramétrables, et un grand nombre de ces logiciels envoient directement les paquets à la victime. (Voir figure II.1).

Dans ce cas, il est possible en général de remonter à l'origine de l'attaque, identifiant par la même occasion l'identité de l'attaquant.

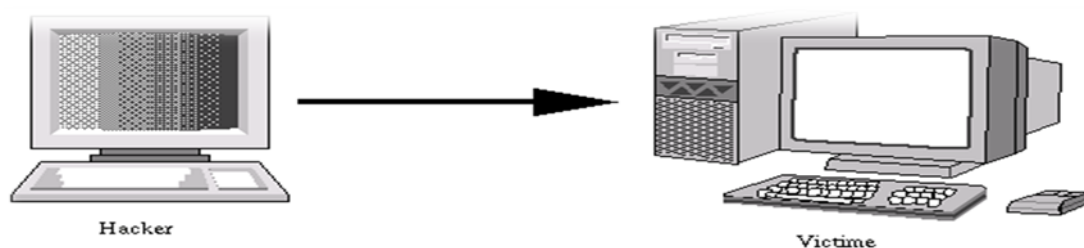


Figure II.1: Attaque directe.

5.2. Les attaques indirectes :

5.2.1. Les attaques indirectes par rebond :

Cette attaque est très prisée des hackers. En effet, le rebond a deux avantages :

- Masquer l'identité (l'adresse IP) du hacker.
- Éventuellement, utiliser les ressources de l'ordinateur intermédiaire car il est plus puissant (CPU, bande passante...) pour attaquer.

Pour que l'attaquant ne se fasse pas repérer, les paquets d'attaques sont envoyés à l'ordinateur intermédiaire, qui répercute l'attaque vers la victime 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. (Voir figure II.2).

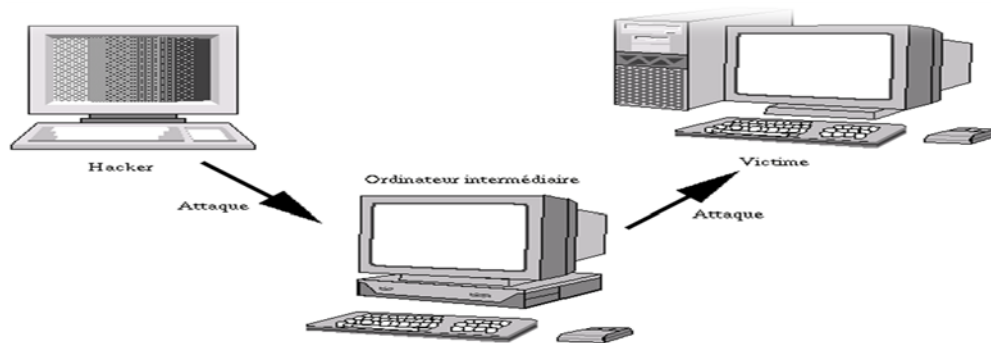


Figure II.2 : Attaque indirecte par rebond.

5.2.2. Les attaques indirectes par réponse :

Cette attaque est un dérivé de l'attaque par rebond. Elle offre les mêmes avantages, du point de vue du hacker, mais au lieu d'envoyer une attaque à l'ordinateur intermédiaire pour qu'il la répercute, l'attaquant va lui envoyer une requête. Et c'est cette réponse à la requête qui va être envoyée à l'ordinateur victime. (Voir figure II.3).



Figure II.3: Attaque indirecte par réponse.

6. Les techniques d'attaques :

6.1. Attaques applicatives :

6.1.1. Virus : code inclus dans un programme qui se recopie lui-même dans d'autres programmes, il est capable de se propager à d'autres ordinateurs en s'insérant dans d'autres programmes ou des documents légitimes appelés « hôtes ». Son but est de détruire une partie

ou toutes les données de l'ordinateur, ou encore de rendre inutilisables certaines fonctions du PC. Il peut en outre ralentir certaines procédures.

6.1.2. Vers : Un Ver est un petit programme qui se copie d'ordinateur en ordinateur. La différence entre un ver et un virus est que le ver ne peut pas se greffer à un autre programme et ne peut donc l'infecter, il va simplement se copier d'ordinateur en ordinateur par l'intermédiaire d'un réseau comme Internet ou même par les lecteurs de disquettes, graveur.

Le ver peut donc non seulement affecter un ordinateur, mais aussi dégrader les performances des réseaux. Comme un virus, le ver peut contenir une action nuisible qui peut être très grave comme le formatage de votre disque dur ou l'envoi de données confidentielles.

6.1.3. Cheval de Troie : Un cheval de Troie sur un ordinateur est un programme exécutable qui est présenté comme ayant une action précise, généralement bénéfique pour l'ordinateur (comme un logiciel gratuit ou sous licence), mais lorsque ce programme est lancé, il va causer des actions plus ou moins graves sur votre ordinateur, comme supprimer des mots de passe, voler des mots de passe, envoyer des informations confidentielles au créateur du programme, formater votre disque dur...

6.1.4. Bombe logique : code incorporé dans des programmes légitimes et qui s'active à un moment déterminé en exploitant la date du système, le lancement d'une commande, ou n'importe quel appel au système.

6.1.5. Trappes (portes dérobées) : Point d'entrée secret écrit dans un code (permet de mettre au point et évaluer des programmes quand il y a un bug par exemple) qui peut autoriser des utilisateurs illégitimes.

6.1.6. Un exploit : est un élément de programme permettant à un individu ou à un logiciel malveillant d'exploiter une faille de sécurité informatique dans un système d'exploitation ou un logiciel, que ce soit à distance ou sur la machine sur laquelle cet exploit est exécuté ; ceci, afin de s'emparer d'un ordinateur ou d'un réseau, d'accroître le privilège d'un logiciel ou d'un utilisateur, ou encore d'effectuer une attaque par déni de service.

6.1.7. Le logiciel espion (spyware en anglais) : collecteur d'informations personnelles sur l'ordinateur d'un utilisateur sans son autorisation, et en envoyant celles-ci à un organisme tiers.

6.1.8. L'enregistreur de frappe (keylogger en anglais) : programme généralement invisible installé sur le poste d'un utilisateur et chargé d'enregistrer à son insu ses frappes clavier.

6.1.9. Déni de service : Le déni de service (DoS, en anglais) apparaît lorsqu'un pirate désactive ou altère un réseau, des systèmes ou des services dans le but de refuser le service prévu aux utilisateurs normaux. Les attaques par déni de service mettent le système en panne ou le ralentissent au point de le rendre inutilisable. Le déni de service peut consister simplement à supprimer ou altérer des informations. Dans la plupart des cas, l'attaque se résume à exécuter un programme pirate ou un script. C'est pour cette raison que les attaques par déni de service sont les plus redoutées.

6.1.10. Attaques de mot de passe : Les attaques de mot de passe peuvent se faire à l'aide d'un analyseur de paquets pour glaner les comptes et les mots de passe utilisateur transmis en clair. Les attaques de mot de passe se rapportent en général aux tentatives de connexion répétées à une ressource partagée, comme un serveur ou un routeur, afin d'identifier un compte utilisateur, un mot de passe ou les deux. Ces tentatives répétées s'appellent attaques par dictionnaire ou attaques en force.

6.1.11. Attaque l'homme au milieu: L'attaque man in the middle ou « attaques de l'intercepteur », 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. La plupart des attaques de type « man in the middle » consistent à écouter le réseau à l'aide d'un outil appelé sniffer.

6.2. Attaques réseaux :

6.2.1. Le sniffing : technique permettant à un hacker d'examiner le contenu d'un certain nombre de paquets et de récupérer toutes les informations transitant sur un réseau qui ne lui ont pas été initialement destinés. Elle est généralement utilisée pour récupérer les mots de passe des applications qui ne chiffrent pas leurs communications, et pour identifier les machines qui communiquent sur le réseau.

6.2.2. La mystification (en anglais spoofing) : technique consistant à prendre l'identité d'une autre personne ou d'une autre machine. Elle est généralement utilisée pour récupérer des informations sensibles, que l'on ne pourrait pas avoir autrement.

6.2.3. L'usurpation d'adresse IP : L'usurpation d'adresse IP également appelé mystification ou en anglais 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.

Ainsi, certains tendent à assimiler l'utilisation d'un proxy (permettant de masquer d'une certaine façon l'adresse IP) avec du spoofing IP. Toutefois, le proxy ne fait que relayer les paquets. Ainsi même si l'adresse est apparemment masquée, un pirate peut facilement être retrouvé grâce au fichier journal (logs) du proxy.

6.2.4. ARP Poisoning : Une des attaques 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 falsifié indiquant que l'adresse MAC de l'autre machine a changé.

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 destinataire.

7. Les contre-mesures de différentes attaques :

7.1. Antivirus :

Les logiciels antivirus sont des logiciels capables de détecter des virus sur un ordinateur, détruire, mettre en quarantaine et parfois de réparer les fichiers infectés sans les endommager. Ils utilisent pour cela de nombreuses techniques, parmi lesquelles :

- Le contrôle général du système de l'ordinateur
- La surveillance des lecteurs de supports amovibles

On préconise souvent d'utiliser des antivirus avec licence et les mettre à jour, car celle-ci permet de corriger les failles détectées soit par les utilisateurs ou les concepteurs de ces systèmes informatiques. [5]

7.2. Les mécanismes de chiffrement

7.2.1. Le chiffrement symétrique :

Le chiffrement symétrique utilise une clé unique partagée entre les 2 interlocuteurs. On encode et on décode le message avec la même clé et doit bien évidemment rester secrète.

La situation peut cependant devenir complexe, si on doit envoyer un message chiffré à deux personnes, plus le nombre de personnes est grand, plus il est difficile de gérer les clés symétriques. D'autant qu'il faut au préalable trouver un moyen sûr de transmettre la clé.

Un algorithme répandu consiste à effectuer des substitutions et des transpositions en cascade sur les lettres du message. Par exemple: MICROAPPLICATION devient IRMCAPOPIALCINTO si l'on fixe comme clé le principe suivant : "la première lettre sera la troisième, la seconde sera la première, la troisième sera la quatrième et la quatrième sera la seconde". La figure suivante illustre le principe du chiffrement symétrique.

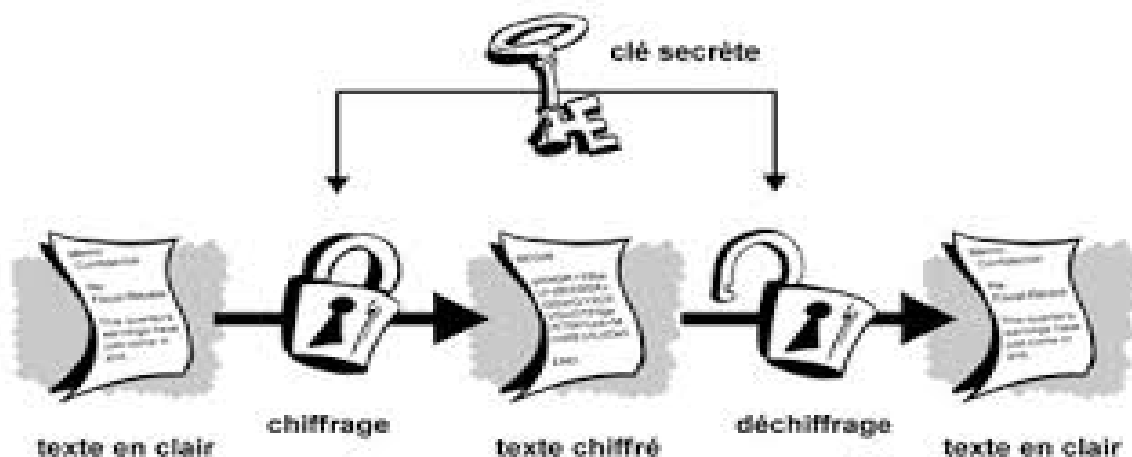


Figure II.4. Chiffrement symétrique.

Les inconvénients de ce système de chiffrement sont les suivants :

- il faut autant de paires de clés que de couples de correspondants ;
- la non-répudiation n'est pas assurée. Mon correspondant possédant la même clé que moi, il peut fabriquer un message en usurpant mon identité ;
- il faut pouvoir se transmettre la clé au départ sans avoir à utiliser le média à sécuriser.

7.2.2. Le chiffrement asymétrique :

Le principe du chiffrement asymétrique est d'avoir 2 clés (que l'on fabrique soi même):

- Quand on encode avec la première clé, on peut décoder avec la 2e clé.
- Quand on encode avec la 2e clé, on peut décoder avec la 1ère clé.

Par convention, on appelle une des 2 clés la clé privée et l'autre la clé publique. La clé privée n'est jamais transmise à personne ; par contre la clé publique est en revanche diffusée publiquement sans problème. Dans une transmission simple, ce système permet 2 choses essentielles:

- Assurer la confidentialité d'un message transmis : personne ne peut le lire sauf le destinataire.
- Signer un message : le destinataire est certain que le message vient bien du bon interlocuteur [5].

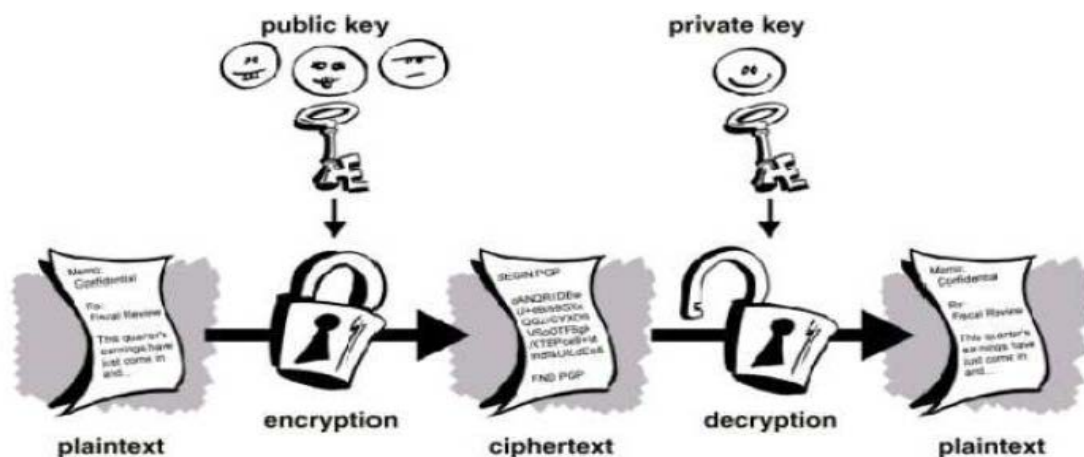


Figure II.5: Chiffrement asymétrique.

7.3. Pare feu :

Un pare-feu (appelé aussi coupe-feu, garde-barrière ou firewall en anglais), 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 réseaux suivantes : une interface pour le réseau à protéger (réseau interne) ; une interface pour le réseau externe. (Voir figure II.6).

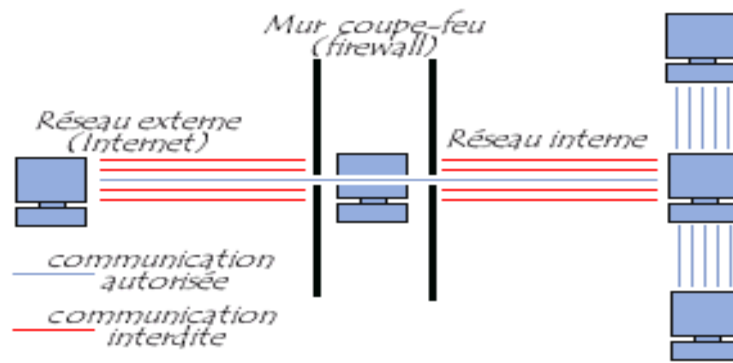


Figure II.6. Fonctionnement d'un Pare feu.

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 doute la plus sûre, mais elle impose toutefois une définition précise et contraignante des besoins en communication [5].

7.4. DMZ :

Une DMZ ou zone démilitarisée est une partie du réseau local dont l'objectif est d'être accessible depuis l'extérieur du réseau local, avec ou sans authentification préalable. En effet, pour des raisons à la fois techniques et stratégiques.

Une zone démilitarisée est définie aussi comme un sous-réseau séparé du réseau local et isolé de celui-ci et d'Internet (ou d'un autre réseau) par un pare-feu. Ce sous-réseau contient les machines étant susceptibles d'être accédées depuis Internet.

Le pare-feu contrôle toutes les communications et les autorisera à passer ou bloquera donc les accès au réseau local pour garantir sa sécurité. Et les services susceptibles d'être accédés depuis Internet seront situés en DMZ.

7.5. Le proxy :

Un serveur proxy, appelé aussi «**serveur mandataire**» est à l'origine une machine faisant fonction d'intermédiaire entre les ordinateurs d'un réseau local 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, ...).

Le principe de fonctionnement basique d'un serveur proxy est assez simple : lorsqu'un utilisateur se connecte à internet 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 [5].

Les serveurs proxy sont toujours d'actualité grâce à un certain nombre d'autres fonctionnalités telles que :

- **La fonction cache :**

La capacité à garder en mémoire «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.

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 valables.

- **Le filtrage :**

D'autre part, grâce à l'utilisation d'un proxy, il est possible d'assurer un suivi des connexions 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. 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 appelé filtrage de contenu.

- **L'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.

- **Le reverse proxy :**

Comme son nom l'indique c'est du proxy inversé. Le reverse-proxy sert ainsi de relais pour les utilisateurs internet souhaitant accéder à un site web interne en lui transmettant indirectement les requêtes. Grace au reverse-proxy, le serveur web est protégé des attaques directes de l'extérieur, ce qui renforce la sécurité du réseau interne. Le reverse-proxy peut servir à repartir la charge en rédigeant les requêtes vers différents serveurs équivalents (load balancing).

7.6. Un certificat électronique :

Il est aussi appelé certificat numérique ou certificat de clé publique peut être vu comme une carte d'identité numérique. Il est utilisé principalement pour identifier et authentifier une personne, mais aussi pour chiffrer des échanges.

Un certificat numérique est un ensemble de données contenant :

- Au moins une clé publique.
- Des informations d'identification, par exemple : nom, localisation, adresse électronique.
- Au moins une signature (clé privée) ; de fait quand il n'y en a qu'une, l'entité signataire est la seule autorité permettant de prêter confiance (ou non) à l'exactitude des informations du certificat... .

Un certificat électronique permet de s'authentifier sur des applications ou des sites, de signer électroniquement des documents mais aussi de chiffrer électroniquement ces derniers. En aucun cas, cet élément d'identité numérique ne peut être prêté ou utilisé par une personne

autre que son titulaire. Par ailleurs, le certificat électronique peut aussi être délivré à un serveur web afin de certifier le lien entre une organisation et un nom de domaine.

7.7. La signature électronique :

Une signature électronique peut être ajoutée aux fichiers pour servir de signe reconnaissable approuvant le contenu d'un document. Grâce à la signature électronique, il est possible de vérifier l'éditeur d'un fichier mais aussi de savoir que le document n'a subi aucun changement ni aucune modification. En effet, sans une signature numérique, il est encore possible de douter la source prétendue d'un document mais aussi d'imaginer que celui-ci ait été falsifié depuis sa publication. Autant donc ne pas ouvrir le fichier surtout si l'identité de l'auteur n'est pas connue avec certitude et que son contenu ne détient aucun risque de danger.

Parmi donc les principaux avantages d'une signature électronique, elle permet entre autres d'apporter une forte authentification de l'émetteur mais aussi de garantir l'intégrité d'un document car le système permet facilement de détecter toute modification apportée au document après sa signature.[5]

8. Les protocoles de sécurité :

8.1. Le protocole HTTPS :

L'HyperText Transfer Protocol Secure est la combinaison du HTTP avec une couche de chiffrement comme SSL.

HTTPS permet au visiteur de vérifier l'identité du site web auquel il accède, grâce à un certificat d'authentification émis. Il garantit théoriquement la confidentialité et l'intégrité des données envoyées par l'utilisateur (notamment des informations entrées dans les formulaires) et reçues du serveur. Pour cela, il fait usage de méthodes de cryptographie asymétrique pour l'authentification et de méthode de cryptographie symétrique pour le chiffrement des échanges.

HTTPS est généralement utilisé pour les transactions financières en ligne : commerce électronique, banque en ligne, courtage en ligne, etc. Il est aussi utilisé pour la consultation de données privées, comme les courriers électroniques, par exemple.

8.2. Le protocole SSH :

Secure Shell est un protocole qui facilite les connexions sécurisées entre deux systèmes à l'aide d'une architecture client/serveur et permet aux utilisateurs de se connecter à distance à des systèmes hôte de serveurs. Toutefois, contrairement à d'autres protocoles de communication à distance, tels que FTP ou Telnet, SSH crypte la session de connexion et empêche ainsi tout agresseur de recueillir des mots de passe non-cryptés.

SSH est conçu pour remplacer les applications de terminal plus anciennes et moins sécurisées qui sont utilisées pour se connecter à des hôtes distants, comme telnet. Étant donné que ces applications plus anciennes ne cryptent pas les mots de passe entre le client et le serveur, il est recommandé d'éviter autant que possible de les utiliser. En effet, l'utilisation de méthodes sécurisées pour se connecter à des systèmes distants, réduit les risques aussi bien pour le système client que pour l'hôte distant.

8.3. Le protocole IPsec :

IP Security est un protocole qui vise à sécuriser l'échange de données au niveau de la couche réseau. Il est basé sur deux mécanismes. Le premier, AH, pour Authentication Header vise à assurer l'intégrité et l'authenticité des datagrammes IP. Il ne fournit par contre aucune confidentialité : les données fournies et transmises par Ce "protocole" ne sont pas encodées. Le second, Esp, pour Encapsulating Security Payload peut aussi permettre l'authentification des données mais est principalement utilisé pour le cryptage des informations. Bien qu'indépendants ces deux mécanismes sont presque toujours utilisés conjointement.

8.4. Le protocole TLS/SSL :

TLS signifie Transport Layer Security (sécurité de couche de transmission) est le successeur du protocole SSL (Secure Sockets Layer, couche de sockets de sécurité). Le protocole TLS/SSL est un système de règles communes suivies aussi bien par les clients que par les serveurs lorsqu'un client visite un site web sécurisé par HTTPS, avant que le client et le serveur ne puissent initier une communication sécurisée, ils doivent passer par une procédure nommée "**négociation TLS**" ; le résultat étant une clé de session sûre, symétrique permettant des transmissions de données chiffrées entre le serveur et le client. La clé de session est symétrique vu qu'elle est utilisée par le client et par le serveur pour chiffrer et déchiffrer des transmissions.

En d'autres mots, la clé de session est un ensemble unique d'instructions informatiques masquant (ou en termes plus simple chiffrant) les contenus d'origine d'une transmission de données, de sorte que seul ceux ayant la clé de session identique puisse déchiffrer et lire ces contenus.

9. Le VPN :

Un réseau VPN repose sur un protocole appelé "protocole de tunneling". Ce protocole permet de faire circuler les informations de l'entreprise de façon cryptée d'un bout à l'autre du tunnel. Ainsi, les utilisateurs ont l'impression de se connecter directement sur le réseau de leur entreprise. (Voir figure II.7).

Le principe de tunneling consiste à construire un chemin virtuel après avoir identifié l'émetteur et le destinataire. Par la suite, la source chiffre les données et les achemine en empruntant Ce chemin virtuel. Afin d'assurer un accès aisé et peu coûteux aux intranets ou aux extranets d'entreprise, les réseaux privés virtuels d'accès simulent un réseau privé, alors qu'ils utilisent en réalité une infrastructure d'accès partagée, comme Internet [5].

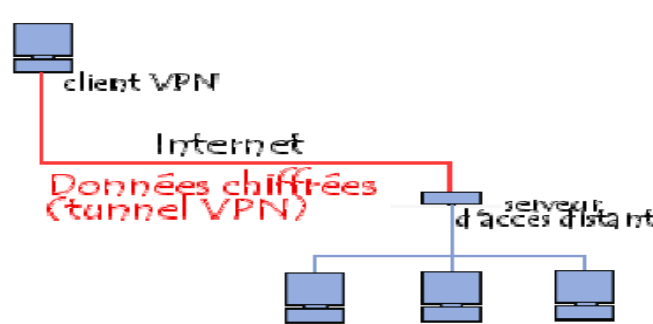


Figure II.7. VPN.

Discussion :

Dans ce chapitre, nous avons donné en premier lieu la description des différents types d'attaques et menaces pour s'introduire dans un système. En second lieu, nous avons exposé les différentes contre-mesures et les protocoles de sécurité du réseau.

De nombreux mécanismes ont été conçus pour tester les vulnérabilités des systèmes, détecter, prévenir et lutter contre les attaques informatiques. Dans le chapitre suivant, nous allons étudier les méthodes et techniques des systèmes d'intrusions qui permettent de prendre le contrôle d'une machine et assurer sa protection.

Préambule :

Nous avons présenté au chapitre II un scénario possible d'attaque sur un réseau et les différents mécanismes de protection contre ces attaques. Malgré leurs grands intérêts, ils présentent des lacunes dues à l'évolution des techniques utilisées par les hackers.

Ainsi et dans l'impossibilité de détecter toute les attaques, le système de détection d'intrusion tente de déceler les attaques qui passent inaperçues à travers les mécanismes de sécurité. Un IDS a pour fonction d'analyser au temps réel ou différer le trafic réseau et de détecter les différentes attaques lancées contre le réseau.

Dans ce chapitre nous allons étudier les systèmes de détection d'intrusion et les systèmes de prévention d'intrusion, ainsi leurs types, leur fonctionnement et nous allons présenter l'architecture des IDS.

1. Définition d'un système d'intrusion :

Un IDS (Intrusion Detection System) est un ensemble de composants logiciels et/ou matériels dont la fonction principale est de détecter et analyser des activités anormales ou suspectes sur la cible analysée (un réseau ou un hôte). Il permet ainsi d'avoir une connaissance sur les tentatives réussies comme échouées des intrusions.

Certains termes sont souvent employés quand on parle d'IDS :

- **Faux positif** : une alerte provenant d'un IDS, mais qui ne correspond pas à une attaque réelle.
- **Faux négatif** : une intrusion réelle qui n'a pas été détectée par l'IDS.

2. Les types des systèmes de détection d'intrusion :

À cause de la diversité des attaques que mettent en œuvre les pirates, la détection d'intrusion doit se faire à plusieurs niveaux. Il existe donc différents types d'IDS :

2.1. Les NIDS (Network-based Intrusion Detection System) :

Les NIDS sont des IDS dédiés aux réseaux. Ils comportent généralement une machine qui écoute sur le segment de réseau à surveiller, un capteur et un moteur qui réalise l'analyse du trafic afin de détecter les intrusions en temps réel. Un NIDS écoute donc tout le trafic réseau, puis l'analyse et génère des alertes si des paquets semblent dangereux. (Voir figure III.1).

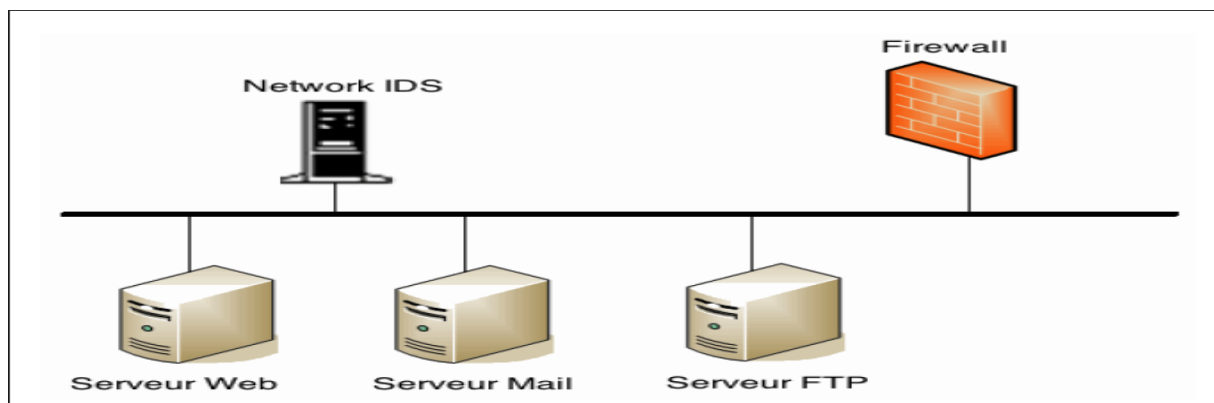


Figure III.1 : Système de détection d'intrusion réseau.

L'implantation d'un NIDS sur un réseau se fait de la façon suivante: des capteurs (souvent de simples hôtes) sont placés aux endroits stratégiques du réseau et génèrent les alertes s'ils détectent une attaque. Ces alertes sont envoyées à une console sécurisée qui les analyse et les traite éventuellement. Cette console est généralement située sur un réseau isolé qui relie uniquement les capteurs et la console [6].

On peut placer les capteurs dans deux endroits différents :

- **A l'intérieur du pare-feu :**

Si les capteurs se trouvent à l'intérieur du pare-feu, il sera plus facile de dire si le pare-feu a été mal configuré et nous pouvons ainsi savoir si une attaque est venue par ce pare-feu.

- **A l'extérieur du pare-feu :**

Les capteurs placés à l'extérieur du pare-feu servent à la détection et l'analyse d'attaques. Il offre l'avantage d'écrire dans les logs, ainsi l'administrateur voit ce qu'il doit modifier dans la configuration du pare-feu.

Quelques exemples des NIDS :

- NetRanger [<http://www.cisco.com>].
- Dragon [<http://www.securitywizards.com>].
- NFR [<http://www.nfr.net>].
- Snort [<http://www.snort.org>].
- DTK [<http://all.net/dtk/dtk.html>].

2.2. Les HIDS (Host-based Intrusion Detection System):

Les systèmes de détection d'intrusion basés sur l'hôte analysent exclusivement l'information concernant cet hôte. Comme ils n'ont pas à contrôler le trafic du réseau mais seulement les activités d'un hôte ils se montrent habituellement plus précis sur les types d'attaques. De plus, nous remarquons immédiatement l'impact sur la machine concernée comme par exemple si un utilisateur l'attaquait avec succès. Ces IDS utilisent deux types de sources pour fournir une information sur l'activité : les logs et les traces d'audit du système d'exploitation. Chacun a ses avantages : les traces d'audit sont plus précises et détaillées et fournissent une meilleure information alors que les logs qui ne fournissent que l'information essentielle sont plus petits. (Voir figure III.2).

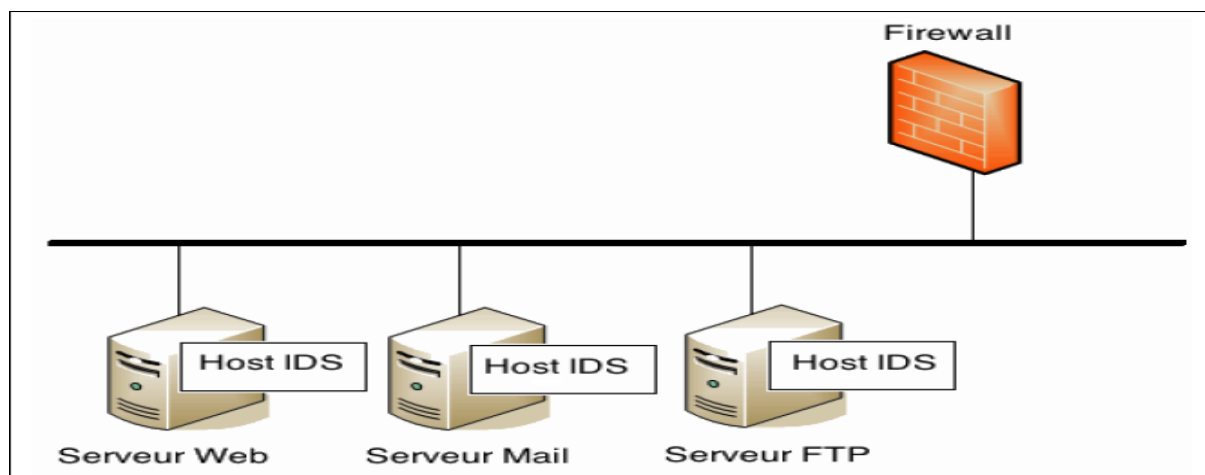


Figure III.2 : Système de détection d'intrusion hôte.

Quelques exemples des HIDS :

- Tripwire [<http://www.tripwire.com/products/index.cfm>].
- SWATCH [http://freshmeat.net/redir/swatch/10125/url_homepage/swatch].
- DragonSquire [<http://www.enterasys.com/ids/squire/>].
- Tiger [http://freshmeat.net/redir/tiger-audit/30581/url_homepage/tiger].

2.3. IDS hybride :

Les systèmes de détections d'intrusions hybrides, rassemblent les caractéristiques de plusieurs systèmes de détections différents. En pratique, on trouve la combinaison des NIDS et HIDS qui permettent de surveiller le réseau et l'hôte.

Les sondes agissent comme un NIDS ou un HIDS Il permet de réunir les informations de diverses sondes placées sur le réseau. L'exemple le plus connu dans le monde Open-Source est Prelude.

- Cet IDS permet de stocker dans une base de données des alertes provenant de différents systèmes relativement variés.
- Utilisant Snort comme NIDS, et d'autres logiciels tels que Samhain en tant que HIDS, il permet de combiner des outils puissants tous ensemble pour permettre une visualisation centralisée des attaques.

3. Caractéristiques d'un système de détection d'intrusion :

Parmi les caractéristiques souhaitables trouvées dans un système de détection d'intrusion nous pouvons citer :

- Résister aux tentatives de corruption, c'est-à-dire, il doit pouvoir détecter s'il a subi lui-même une modification indésirable.
- Utiliser un minimum de ressources de système sous surveillance.
- S'adapter au cours du temps aux changements du système surveillé et du comportement des utilisateurs.
- Etre facilement configurable pour implémenter une politique de sécurité spécifique d'un réseau.

4. L'architecture d'un IDS :

Cette section décrit les trois composants qui constituent classiquement un système de détection d'intrusions. La Figure III.3 illustre les interactions entre ces trois composants[6].

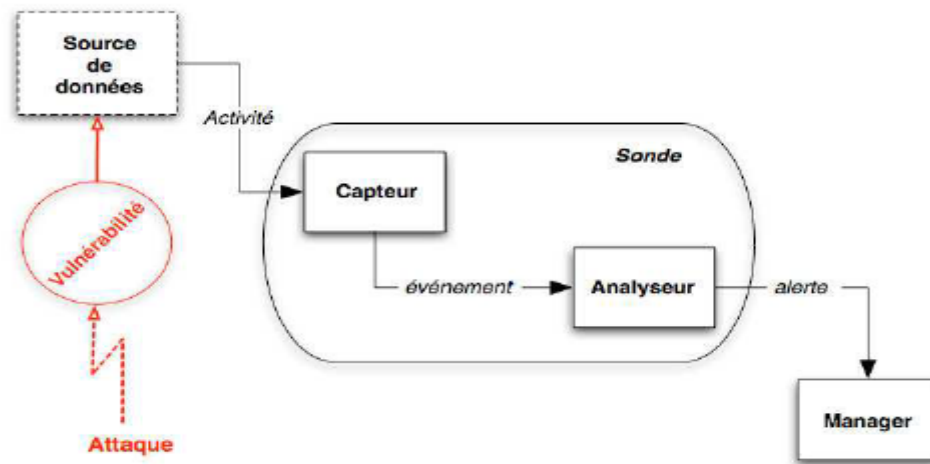


Figure III.3: L'architecture d'un IDS.

4.1. Capteur :

Le capteur observe l'activité du système par le biais d'une source de données et fournit à l'analyseur une séquence d'événements qui renseignent de l'évolution de l'état du système. Le capteur peut se contenter de transmettre directement ces données brutes, mais en général un prétraitement est effectué.

On distingue classiquement trois types de capteurs en fonction des sources de données utilisées pour observer l'activité du système : les capteurs système, les capteurs réseau et les capteurs applicatifs.

4.2. Analyseur :

L'objectif de l'analyseur est de déterminer si le flux d'événements fourni par le capteur contient des éléments caractéristiques d'une activité malveillante.

4.3. Manager :

Le manager collecte les alertes produites par le capteur, les met en forme et les présente à l'opérateur. Éventuellement, le manager est chargé de la réaction à adopter qui peut être :

- Confinement de l'attaque, qui a pour but de limiter les effets de l'attaque.
- Eradication de l'attaque, qui tente d'arrêter l'attaque.
- Recouvrement, qui est l'étape de restauration du système dans un état sain.
- Diagnostic, qui est la phase d'identification du problème.

5. Mise en place d'un IDS :

Le positionnement de l'IDS :

Il existe plusieurs endroits stratégiques où il convient de placer un IDS. Le schéma suivant illustre un réseau local ainsi que les trois positions que peut y prendre un IDS :

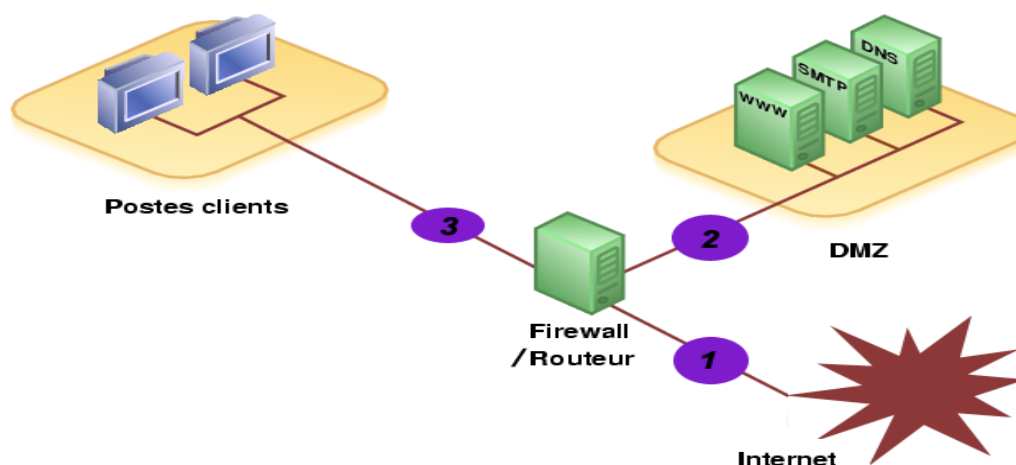


Figure III.4: La position des IDS.

- **Position (1) :** Sur cette position, l'IDS va pouvoir détecter l'ensemble des attaques frontales, provenant de l'extérieur, en amont du firewall. Ainsi, beaucoup d'alertes seront remontées ce qui rendra les logs difficilement consultables.
- **Position (2) :** Si l'IDS est placé sur la DMZ, il détectera les attaques qui n'ont pas été filtrées par le firewall et qui relèvent d'un certain niveau de compétence. Les logs seront ici plus clairs à consulter puisque les attaques ne seront pas recensées.
- **Position (3) :** L'IDS peut ici rendre compte des attaques internes, provenant du réseau local de l'entreprise. Il peut être judicieux d'en placer un à cet endroit étant donné le fait que 80% des attaques proviennent de l'intérieur. De plus, si des trojans ont contaminé le parc informatique (navigation peu méfiante sur internet) ils pourront être ici facilement identifiés pour être ensuite éradiqués.

6. Mode de fonctionnement :

Il faut distinguer deux aspects dans le fonctionnement d'un IDS : le mode de détection utilisé et la réponse apportée par l'IDS lors de la détection d'une intrusion.

Il existe deux modes de détection, la détection d'anomalies et la reconnaissance de signatures. D'eux-mêmes, deux types de réponses existent, la réponse passive et la réponse active.

Nous allons tout d'abord étudier les modes de détection d'un IDS, avant de présenter les réponses possibles à une attaque[7].

6.1. Modes de détection :

Nous notons deux modes de détection qui sont :

- La détection d'anomalies
- La reconnaissance de signature

Il faut noter que la reconnaissance de signature est le mode de fonctionnement le plus implémenté par les IDS du marché. Cependant, les nouveaux produits tendent à combiner les deux méthodes pour affiner la détection d'intrusion.

6.1.1. La détection d'anomalies :

Elle consiste à détecter des anomalies par rapport à un profil "de trafic habituel". La mise en œuvre comprend toujours une phase d'apprentissage au cours de laquelle les IDS vont découvrir le fonctionnement normal des éléments surveillés. Ils sont ainsi en mesure de signaler les divergences par rapport au fonctionnement de référence.

Les modèles comportementaux peuvent être élaborés à partir d'analyses statistiques. Ils présentent l'avantage de détecter des nouveaux types d'attaques.

Cependant, de fréquents ajustements sont nécessaires afin de faire évoluer le modèle de référence de sorte qu'il reflète l'activité normale des utilisateurs et réduire le nombre de fausses alertes générées.

Dans le cas d'HIDS, ce type de détection peut être basé sur des informations telles que le taux d'utilisation CPU, l'activité sur le disque, les horaires de connexion ou d'utilisation de certains fichiers (horaires de bureau...).

6.1.2. La reconnaissance de signatures :

Cette approche consiste à rechercher dans l'activité de l'élément surveillé les empreintes (ou signatures) d'attaques connues. Ce type d'IDS est purement réactif ; il ne peut détecter que les attaques dont il possède la signature.

De ce fait, il nécessite des mises à jour fréquentes. De plus, l'efficacité de ce système de détection dépend fortement de la précision de sa base de signature. C'est pourquoi ces systèmes sont contournés par les pirates qui utilisent des techniques dites "d'évasion" qui consistent à maquiller les attaques utilisées. Ces techniques tendent à faire varier les signatures des attaques qui ainsi ne sont plus reconnues par l'IDS.

Il est possible d'élaborer des signatures plus génériques, qui permettent de détecter les variantes d'une même attaque, mais cela demande une bonne connaissance des attaques et du réseau, de façon à stopper les variantes d'une attaque et à ne pas gêner le trafic normal du réseau.

Une signature permet de définir les caractéristiques d'une attaque, au niveau des paquets (jusqu'à TCP ou UDP) ou au niveau des protocoles (HTTP, FTP...).

- Au niveau paquet, l'IDS va analyser les différents paramètres de tous les paquets transitant et les comparer avec les signatures d'attaques connues.
- Au niveau protocole, l'IDS va vérifier au niveau du protocole si les commandes envoyées sont correctes ou ne contiennent pas d'attaque. Cette fonctionnalité a surtout été développée pour HTTP actuellement.

Il faut savoir que les signatures sont mises à jour en fonction des nouvelles attaques identifiées.

Néanmoins, plus il y a de signatures différentes à tester, plus le temps de traitement sera long. L'utilisation de signatures plus élaborées peut donc procurer un gain de temps appréciable.

Cependant, une signature mal élaborée peut ignorer des attaques réelles ou identifiées du trafic normal comme étant une attaque. Il convient donc de manier l'élaboration de signatures avec précaution, et en ayant de bonnes connaissances sur le réseau surveillé et les attaques existantes.

Une fois une attaque détectée, un IDS a le choix entre plusieurs types de réponses, que nous allons maintenant détailler.

6.2. Réponses actives et passives :

Il existe deux types de réponses, suivant les IDS utilisés. La réponse passive est disponible pour tous les IDS alors que la réponse active est plus ou moins implémentée.

6.2.1. Réponse passive :

La réponse passive d'un IDS consiste à enregistrer les intrusions détectées dans un fichier de log qui sera analysé par le responsable de sécurité.

Certains IDS permettent de logger l'ensemble d'une connexion identifiée comme malveillante. Ceci permet de remédier aux failles de sécurité pour empêcher les attaques enregistrées de se reproduire, mais elle n'empêche pas directement une attaque de se produire.

6.2.2. Réponse active :

La réponse active, au contraire a pour but de stopper une attaque au moment de sa détection. Pour cela nous disposons d'une reconfiguration du firewall.

La reconfiguration du firewall permet de bloquer le trafic malveillant au niveau du firewall, en fermant le port utilisé ou en interdisant l'adresse de l'attaquant. Cette fonctionnalité dépend du modèle de firewall utilisé, tous les modèles ne permettent pas la reconfiguration par un IDS.

Dans le cas d'une réponse active, il faut être sûr que le trafic détecté comme malveillant l'est réellement, sous peine de déconnecter des utilisateurs normaux.

En général, les IDS ne réagissent pas activement à toutes les alertes. Ils ne répondent aux alertes que quand celles-ci sont positivement certifiées comme étant des attaques. L'analyse des fichiers d'alertes générés est donc une obligation pour analyser l'ensemble des attaques détectées.

7. Les méthodes de détections :

Pour bien gérer un système de détection d'intrusions, il est important de comprendre comment celui-ci fonctionne. Une question simple se pose sur la détection d'un tel système et sur le critère de différencier un flux contenant une attaque d'un flux normal.

Ces questions nous ont amenés à étudier le fonctionnement interne d'un IDS. De là, nous en avons déduit deux techniques mises en place dans la détection d'attaques. La première consiste à détecter des signatures d'attaques connues dans les paquets circulant sur le réseau. La seconde, consiste quant à elle, à détecter une activité suspecte dans le comportement de l'utilisateur. Ces deux techniques, aussi différentes soient-elles, peuvent être combinées au sein d'un même système afin d'accroître la sécurité[6].

7.1. L'approche par scénario (mésuse détection):

Cette technique s'appuie sur la connaissance des techniques utilisées par les attaquants pour déduire des scénarios typiques. Elle ne tient pas compte des actions passées de l'utilisateur et utilise des signatures d'attaques (ensemble de caractéristiques permettant d'identifier une activité intrusive : une chaîne alphanumérique, une taille de paquet inhabituelle, une trame formatée de manière suspecte...).

Plusieurs analyses peuvent être utilisées pour la méthode de détection par scénario:

- **Recherche de motifs (pattern matching):**

La méthode la plus connue et la plus facile à comprendre. Elle se base sur la recherche de motifs (chaînes de caractères ou suite d'octets) au sein du flux de données. L'IDS comporte une base de signatures où chaque signature contient les protocoles et port utilisés par l'attaque ainsi que le motif qui permettra de reconnaître les paquets suspects.

Le principal inconvénient de cette méthode est que seules les attaques reconnues par les signatures seront détectées. Il est donc nécessaire de mettre à jour régulièrement la base de signatures.

Un autre inconvénient est que les motifs sont en général fixes. Or une attaque n'est pas toujours identique à 100 %. Le moindre octet différent par rapport à la signature provoquera la non-détection de l'attaque.

Pour les IDS utilisant cette méthode, il est nécessaire d'adapter la base de signatures en fonction du système à protéger. Cela permet non seulement de diminuer les ressources nécessaires et donc augmenter les performances ; mais également réduire considérablement le nombre de fausses alertes et donc faciliter le travail des administrateurs réseau qui analyseront les fichiers d'alertes.

- **Recherche de motifs dynamiques:**

Le principe de cette méthode est le même que précédemment, mais les signatures des attaques évoluent dynamiquement. L'IDS est de ce fait doté de fonctionnalités d'adaptation et d'apprentissage.

- **Analyse de protocoles:**

Cette méthode se base sur une vérification de la conformité des flux, ainsi que sur l'observation des champs et paramètres suspects dans les paquets.

L'analyse protocolaire est souvent implémentée par un ensemble de préprocesseurs, où chaque préprocesseur est chargé d'analyser un protocole particulier (FTP, HTTP, ICMP...). Du fait de la présence de tous ces préprocesseurs, les performances dans un tel système s'en voient fortement dégradées.

L'intérêt fort de l'analyse protocolaire est qu'elle permet de détecter des attaques inconnues, contrairement au pattern matching qui doit connaître l'attaque pour pouvoir la détecter.

- **Analyse heuristique et détection d'anomalies:**

Le but de cette méthode est une analyse intelligente, de détecter une activité suspecte ou toute autre anomalie. Par exemple : une analyse heuristique permet de générer une alarme quand le nombre de sessions à destination d'un port donné dépasse un seuil dans un intervalle de temps prédéfini.

7.2. L'approche comportementale (Anomalie Detection) :

Cette technique consiste à détecter une intrusion en fonction du comportement passé de l'utilisateur. Pour cela, il faut préalablement dresser un profil utilisateur à partir de ses habitudes et déclencher une alerte lorsque des événements hors profil se produisent.

Cette technique peut être appliquée non seulement à des utilisateurs, mais aussi à des applications et services. Plusieurs métriques sont possibles : la charge CPU, le volume de données échangées, le temps de connexion sur des ressources, la répartition statistique des protocoles et applications utilisés, les heures de connexion... Cependant elle possède quelques inconvénients :

- **peu fiable** : tout changement dans les habitudes de l'utilisateur provoque une alerte ;
- **nécessite une période de non-fonctionnement** pour mettre en œuvre les mécanismes d'auto-apprentissage : si un pirate attaque pendant ce moment, ses actions seront assimilées à un profil utilisateur, et donc passeront inaperçues lorsque le système de détection sera complètement mis en place.
- **l'établissement du profil doit être souple** afin qu'il n'y ait pas trop de fausses alertes : le pirate peut discrètement intervenir pour modifier le profil de l'utilisateur afin d'obtenir après plusieurs jours ou semaines, un profil qui lui permettra de mettre en place son attaque sans qu'elle ne soit détectée.

Plusieurs approches peuvent être utilisées pour la méthode de détection comportementale :

- **Approche probabiliste :**

Des probabilités sont établies permettant de représenter une utilisation courante d'une application ou d'un protocole. Toute activité ne respectant pas le modèle probabiliste provoquera la génération d'une alerte.

- **Approche statistique :**

Le but est de quantifier les paramètres liés à l'utilisateur : taux d'occupation de la mémoire, utilisation des processeurs, valeur de la charge réseau, nombre d'accès à l'Intranet par jour, vitesse de frappe au clavier, sites les plus visités...

Cette méthode est très difficile à mettre en place. Elle n'est actuellement présente que dans le domaine de la recherche, où les chercheurs utilisent des réseaux neuronaux pour tenter d'avoir des résultats convaincants.

8. Détecter un IDS :

Comme nous l'avons déjà signalé, il est très dangereux que la présence d'un IDS soit remarquée par un pirate. Car dans ce cas, il tentera d'obtenir un maximum d'informations sur l'IDS installé (ex. : la version utilisée) pour pouvoir l'outrepasser et attaquer sans se faire remarquer [7]. Voici quelques techniques qui permettent de détecter un IDS :

Usurpation d'adresse MAC : les NIDS mettent l'interface de capture en mode promiscuité (promiscuous mode), il est donc possible de détecter l'IDS en envoyant par exemple un ICMP « echo request » à la machine soupçonnée d'être un NIDS avec une adresse MAC inexistante. Si la machine répond alors elle est en mode promiscuous et peut donc être un NIDS.

Mesure des temps de latence : puisque l'interface est en mode promiscuous, les temps de réponse sont plus longs. Voici une méthode pour exploiter ces temps de latence :

- le pirate génère une série de pings vers l'adresse à tester, puis il mesure et note les temps de réponse.
- le pirate sature ensuite le réseau en broadcast dans le but de ralentir l'IDS, qui recevra tous les paquets. Enfin, le pirate réémet la même série de pings en mesurant les nouveaux temps de réponse. S'ils sont bien plus élevés que les premiers temps obtenus, il est fort possible que la machine soit en mode promiscuous.

Exploiter les mécanismes de réponses actives : les IPS réagissent à certaines attaques (fermer session, bloquer port...), mais en faisant cela, ils laissent souvent des empreintes (header des paquets) permettant d'identifier le type d'IPS.

Observation des requêtes DNS : Les IDS génèrent souvent des requêtes DNS lors des alertes. En observant le DNS primaire lors de fausses attaques, on peut détecter qu'il y a un IDS.

9. Critères de tests d'un IDS :

Lors de la mise en place d'un IDS, il est nécessaire de prendre en considération plusieurs critères qui permettront de choisir au mieux l'IDS.[7] Tester un IDS avec des scanners de vulnérabilité est une mesure nécessaire pour évaluer un IDS, mais est loin d'être suffisante. D'autres critères doivent être pris en compte :

- **Méthodes et capacités de détection :** estimer le taux de faux positifs et la qualité d'information fournie par l'IDS.
- **Rapidité :** tester l'IDS en condition de charge élevée. Il est important de tester cela de manière réaliste, et non pas en utilisant des générateurs de paquets.
- **Ouverture :** il faut que l'IDS permette de modifier les signatures afin d'éviter certains faux positifs, mais aussi d'ajouter de nouvelles signatures spécifiques à l'environnement.
- **Architecture logicielle :** pour les grandes entreprises, il est intéressant de pouvoir séparer les fonctions d'administration.
- **Exploitabilité des données :** il faut disposer d'outils permettant de retrouver et analyser facilement les événements suspects, car le volume généré par les IDS est important. Afin de centraliser les données, il peut être intéressant de disposer de consoles de reporting ou de tableaux de bord.
- **Ergonomie :** on retrouve différents types d'interfaces dans les IDS. Tout d'abord, les interfaces graphiques qui sont adaptées aux particuliers. Ensuite, les interfaces de type Web ou encore les interfaces en ligne de commandes réservées aux spécialistes. Dans tous les cas, l'interface doit offrir de nombreuses fonctionnalités.

D'autres critères, comme la réactivité de l'éditeur (mises à jour des signatures, correctifs...), ou le prix (solution libre ou non) rentrent en jeu. Pour évaluer un IDS, il est intéressant de pondérer chacun de ces critères selon l'importance qu'on leur attribue.

10. Techniques anti-IDS :

Comme tout système informatique, ou presque, il existe des failles dans les IDS, ou plutôt des techniques qui permettent d'outrepasser ces systèmes sans se faire repérer. Si un pirate détecte la présence d'un IDS, il peut le désactiver, ou mieux encore, générer de fausses attaques pendant qu'il commettra son forfait tranquillement[6]. Il existe trois catégories d'attaques contre les IDS :

- **attaque par déni de service** : rendre l'IDS inopérant en le saturant.
- **attaque par insertion** : le pirate, pour éviter d'être repéré, injecte des paquets de leurres qui seront ignorés par le système d'exploitation de la cible, mais pris en compte par l'IDS : l'IDS ne détecte rien d'anormal, alors que sur le système cible, l'attaque a bien lieu puisque les paquets superflus sont ignorés.
- **attaque par évasion** : il s'agit de la technique inverse à l'attaque par insertion. Ici, des données superflues sont ignorées par l'IDS, mais prises en compte par le système d'exploitation.

11. Les limites de l'IDS :

Comme tout système informatique, les IDS ont des limites. On peut en citer :

- **Pollution/surcharge** : Les IDS peuvent être pollués ou surchargés, par exemple par la génération d'un trafic important (le plus difficile et lourd possible à analyser). Une quantité importante d'attaques peut également être envoyée afin de surcharger les alertes de l'IDS. Des conséquences possibles de cette surcharge peuvent être la saturation de ressources (disque,CPU, mémoire), la perte de paquets, le déni de service partiel ou total
- **Consommation de ressources** : outre la taille des fichiers de logs (de l'ordre du Go), la détection d'intrusion est excessivement gourmande en ressources. En effet un système NIDS doit générer des journaux de comptes-rendus d'activité anormale ou douteuse sur le réseau.
- **Perte de paquets (limitation des performances)** : les vitesses de transmission sont parfois telles qu'elles dépassent largement la vitesse d'écriture des disques durs, ou même la vitesse de traitement des processeurs.

Il n'est donc pas rare que des paquets ne soient pas traités par l'IDS, et que certains d'entre eux soient néanmoins reçus par la machine destinataire.

- **Vulnérabilité aux dénis de service** : un attaquant peut essayer de provoquer un déni de service au niveau du système de détection d'intrusion, ou pire au niveau du système d'exploitation de la machine supportant l'IDS.

Une fois que l'IDS est désactivé (« hors service »), l'attaquant peut tenter tout ce qui lui convient.

12. Les systèmes de prévention d'intrusion IPS :

Les IPS ont pour fonction principale d'empêcher toute activité suspecte détectée au sein d'un système, ils sont capables de prévenir une attaque avant qu'elle atteigne sa destination.

Contrairement aux IDS, les IPS sont des outils aux fonctions « actives », qui en plus de détecter une intrusion, tentent de la bloquer[7].

Le principe de fonctionnement d'un IPS est analogue à celui d'un IDS, ajoutant à cela l'analyse des contextes de connexion, l'automatisation d'analyse des logs et la coupure des connexions suspectes. Contrairement aux IDS classiques, aucune signature n'est utilisée pour détecter les attaques. Avant toute action, une décision en temps réel est exécutée (i.e., l'activité est comparée aux règles existantes). Si l'action est conforme à l'ensemble de règles, la permission de l'exécuter sera accordée et l'action sera exécutée. Si l'action est illégale, une alarme est générée. Dans la plupart des cas, les autres détecteurs du réseau (ou une console centrale connectées à l'IPS) en seront aussi informés dans le but d'empêcher les autres ordinateurs d'ouvrir ou d'exécuter des fichiers spécifiques.

12.1. Système de prévention d'intrusion réseau (NIPS) :

Lors de la détection d'une attaque, le système réagit et modifie l'environnement du système attaqué. Cette modification peut être le blocage de certains flux, de certains ports ou l'isolation pure et simple de certains systèmes du réseau. Le point sensible de ce genre de dispositif de prévention est qu'en cas de faux positif, c'est le trafic du système qui est directement affecté. Les erreurs doivent donc être les moins nombreuses possibles car elles ont un impact direct sur la disponibilité des systèmes.

12.2. Système de prévention d'intrusion hôte (HIPS) :

HIPS précède l'action des HIDS en ce sens qu'il est « résident », c'est à dire actif en permanence, dès le lancement du système et jusqu'à son arrêt. Comme un HIDS, il se doit de protéger l'intégrité du système d'exploitation, des logiciels applicatifs lancés, des informations stockées, soit en mémoire RAM soit dans le système de fichiers, les fichiers journaux ou ailleurs, et de vérifier que leur contenu demeure intègre, mais en permanence.

Il doit contrôler instantanément tout ce qui change dans l'ordinateur et veiller à ce que rien ne contourne la politique de sécurité.

Discussion :

La plupart des IDS sont fiables, ce qui explique qu'ils sont souvent intégrés dans les solutions de sécurité. Les avantages qu'ils présentent face aux autres outils de sécurités les favorisent, mais d'un autre côté cela n'empêche pas que les meilleurs IDS présentent aussi des lacunes et quelques inconvénients. Nous comprenons donc bien qu'ils sont nécessaires mais ne peuvent pas se passer de l'utilisation d'autres outils de sécurité visant à combler leurs défauts.

Dans le chapitre suivant, nous allons étudier un système open source de détection d'intrusion réseau qui est le Snort ; ce dernier est l'un des plus populaires dans le monde de détection d'intrusion.

Préambule :

Souvent les réseaux informatiques sont protégés par des firewalls. Cette protection bien que nécessaire, ne permet pas la détection des attaques extérieures. En effet, un firewall est fait pour bloquer les flux réseaux non désirés tout en laissant passer “le trafic utile” Mais le problème qui se pose est comment analyser à l’intérieur de ce trafic les données potentiellement dangereuses et se prévenir des attaques.

Snort est l’une des réponses possible, il va analyser en temps réel, le trafic réseau, recherché dans sa base de données des profils d’attaque connu et logger les résultats.

Dans ce chapitre, nous allons installer le logiciel open source snort sous linux en décrivant toutes les étapes d’installation ainsi sa configuration. Ensuite, nous allons effectuer un test d’intrusion pour s’assurer de la bonne fonctionnalité de ce logiciel.

1. Partie I : Aperçu sur l’IDS réseau Snort :**1.1. Définition du Snort :**

Snort est un système de détection d’intrusion open source. Il est capable d’effectuer en temps réel des analyses de trafic et de logger les paquets sur un réseau IP. Il peut effectuer des analyses de protocole, recherche et correspondance de contenu et peut être utilisé pour détecter une grande variété d’attaques et des tentatives comme des balayages de port de dérobée, des tentatives d’empreinte de OS, et beaucoup plus.

Snort détecte les méthodes d’attaque, y compris de déni de service, les attaques CGI ,les balayages de ports furtifs. Lorsque des comportements suspects sont détectés, Snort envoie une alerte en temps réel à syslog, à un fichier d’alertes distinct.

Snort est basé sur libpcap (pour la capture de paquets de bibliothèque), un outil largement utilisé dans les détecteurs de trafic TCP / IP et les analyseurs grâce à l’analyse du protocole et à la recherche de contenu.

Snort a trois utilisations primaires. Il peut être employé en tant qu'un renifleur de paquet (Sniffer) comme tcpdump(1), un enregistreur de paquet (logs) (utile pour le trafic de réseau corrigeant, etc....), ou comme plein système soufflé de détection d'intrusion de réseau [9].

1.2. L'architecture de Snort :

L'architecture de SNORT est organisée en modules, elle est composée de quatre grands modules : Le décodeur de paquets, les préprocesseurs, le moteur de détection et le système d'alerte et d'enregistrement de log [9]. (Voir figure IV.1).

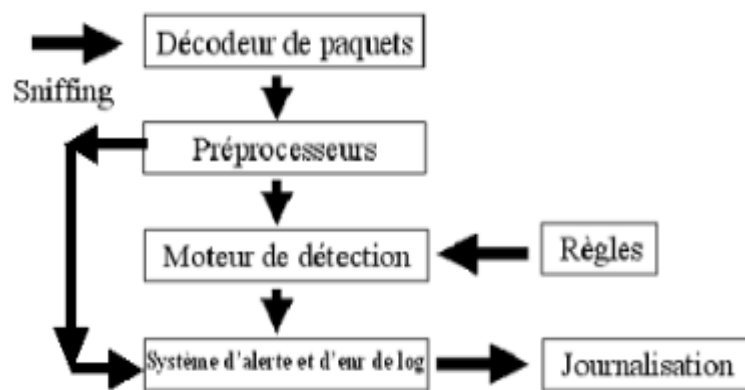


Figure IV.1: Architecture de SNORT.

1.2.1. Le décodeur de paquets :

Un système de détection d'intrusion active un ou plusieurs interfaces réseau de la machine en mode espion (promiscuous mode), ceci va lui permettre de lire et d'analyser tous les paquets qui passent par le lien de communication. SNORT utilise la bibliothèque libpcap pour faire la capture des trames.

Un décodeur de paquets est composé de plusieurs sous décodeurs qui sont organisés par protocole (Ethernet, IP, TCP..), ces décodeurs transforme les éléments des protocoles en une structure de données interne. (Voir figure IV.2).

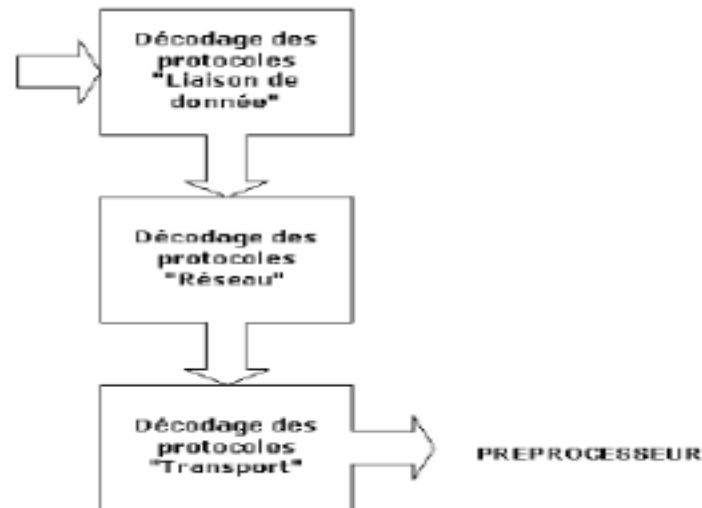


Figure IV.2 : Le décodeur de paquets.

1.2.2. Les préprocesseurs :

Les préprocesseurs s'occupent de la détection d'intrusion en cherchant les anomalies. Un préprocesseur envoie une alerte si les paquets ne respectent pas les normes des protocoles utilisées. Un préprocesseur est différent d'une règle de détection, il est un programme qui vise à aller plus en détail dans l'analyse de trafic.

Les préprocesseurs sont exécutés avant le lancement du moteur de détection et après le décodage du paquet IP. Le paquet IP peut être modifié ou analysé de plusieurs manières en utilisant le mécanisme de préprocesseur. Les préprocesseurs sont chargés et configurés avec le mot-clé `preprocessor`.

1.2.3. Moteur de détection :

C'est la partie la plus importante dans un IDS. Le moteur de détection utilise les règles pour faire la détection des activités d'intrusion. Si un paquet correspond à une règle, alors une alerte est générée. Les règles sont groupées en plusieurs catégories sous forme de fichiers. SNORT vient avec un ensemble de règles prédéfini.

Ces règles ne sont pas activées automatiquement, il faut les activer dans le fichier de configuration `snort.conf`. Chaque fichier contient des règles décrit en type de trafic à signaler. (Voir figure IV.3).

```
include $RULE_PATH/local.rules
include $RULE_PATH/bad-traffic.rules
include $RULE_PATH/exploit.rules
include $RULE_PATH/scan.rules
include $RULE_PATH/finger.rules
include $RULE_PATH/ftp.rules
include $RULE_PATH/telnet.rules
include $RULE_PATH/rpc.rules
include $RULE_PATH/r services.rules
include $RULE_PATH/dos.rules
include $RULE_PATH/ddos.rules
include $RULE_PATH/dns.rules
include $RULE_PATH/tftp.rules
```

Figure IV.3: Une partie du fichier snort.conf.

1. 2.4. Système d'alerte et d'enregistrement des logs :

Le système d'alerte et d'enregistrement des logs s'occupe de la génération des logs et des alertes. Les alertes sont stockées par défaut dans le répertoire **C:\Snort\log**.

Dés que le système devient opérationnel, on pourra consulter les alertes générées directement dans les fichiers textes ou bien utiliser une console de gestion (ACID) ou une version améliorée (BASE). ACID (Analysis Console for Intrusion Detection) est une application qui fournit une console de gestion et qui permet la visualisation des alertes en mode graphique. Les alertes dans ce cas sont stockées dans une base de données MySQL.

1.3. La raison de choix du Snort :

- C'est un logiciel gratuit.
- Il est capable d'effectuer une analyse en temps réel et du trafic entrant et sortant.
- Il est disponible pour la plupart des systèmes d'exploitation (Windows et linux comme Ubuntu, Debian, CentOS).
- Les mises à jour des règles sont gratuites.
- La détection et la notification des attaques sont déjà connues.

2. Partie II : Test d'intrusion

Un test d'intrusion (« pentest » en anglais) est une méthode d'évaluation de la sécurité d'un système ou d'un réseau informatique. Cette méthode consiste généralement à simuler une attaque pour trouver des vulnérabilités exploitables en vue de proposer un plan d'actions permettant d'améliorer la sécurité d'un système [9].

2.1. Objectif du test d'intrusion :

L'objectif d'un test d'intrusion est alors multiple et peut varier selon les contextes :

- Lister un ensemble d'informations, trouvées d'une manière ou d'une autre, et qui peuvent être sensibles ou critiques.
- Dresser une liste des vulnérabilités ou faiblesses du système de sécurité pouvant être exploitées.
- Tester l'efficacité des systèmes de détection d'intrusion.
- Effectuer un rapport et une présentation finale de son avancement et de ses découvertes au client.
- Donner les pistes et conseiller sur les méthodes de résolution et de correction des vulnérabilités découvertes.

2.2. Les outils utilisés pour effectuer cet IDS réseau :

- Un PC TOSHIBA C650 avec un système d'exploitation 32bits et de RAM 2GO.
- Une machine virtuelle VMWare Player.
- Un système d'exploitation ubuntu 14.04.
- Le logiciel que nous utilisons est le snort avec ses règles et aussi installer les conditions préalable avant d'installer le logiciel.

2. 3. Installation et configuration du snort :

2.3.1. Installation des pré-requis :

L'installation des pré-requis est souvent délicate car les pré-requis dépendent souvent aussi d'autre paquets à installer. Raison pour laquelle avant d'installer ces pré-requis nous allons faire une mise à jour système pour s'assurer que nous avons au moins des outils de base pour démarrer.

Pour cela nous ouvrons un terminal et nous nous connectons en tant que « root » et nous exécutons les commandes suivantes :

```
# apt-get update
```

```
# apt-get upgrade
```

Pour l'installation de certains pré-requis il est plus prudent de faire : **# apt-get install nom du paquet**, ainsi le paquet et ses dépendances seront installés. Il se trouve que la plupart des pré-requis à installer sont contenus dans d'autres paquets.

2.3.2. Installation des dépendances :

```
# apt-get update
```

```
# apt-get upgrade
```

➤ Pour SNORT :

```
# apt-get install libpcap
```

```
# apt-get install libperlude
```

➤ Pour MYSQL :

```
# apt-get install mysql-server
```

```
# apt-get install phpmyadmin
```

```
# apt-get install libmysqlclient15-dev
```

```
# apt-get install libpcr3
```

```
# apt-get install libnet1
```

```
# apt-get install libssl-dev
```

Le serveur mysql va se servir de la base de données pour nos différentes alertes et règles :

➤ Pour BASE :

```
# apt-get install adodb  
# apt-get install php5  
# apt-get install apache2  
# apt-get install libapache2-mod-php5  
# apt-get install php-pear  
# apt-get install php5-cli  
# apt-get install libphp-adodb  
# apt-get install php5-gd
```

BASE (Basic Analysis Security Engine) est une application développée en PHP, qui permet de gérer l'interface graphique de snort. Elle est utilisée pour afficher les logs générés par l'IDS et envoyée dans la base de données.

C'est une version améliorée de ACID nous aurons besoin donc d'installer PHP.

Maintenant que toutes les dépendances sont installées, nous allons passer à l'installation de snort et les autres serveurs.

2.3.3. L'installation du snort :

Pour installer snort, il suffit d'exécuter la commande suivante :

```
# apt-get install snort
```

Une fois installé, nous revenons sur notre terminale pour paramétrer snort

```
# dpkg-reconfigure snort
```

Nous devons voir apparaître l'interface suivante :

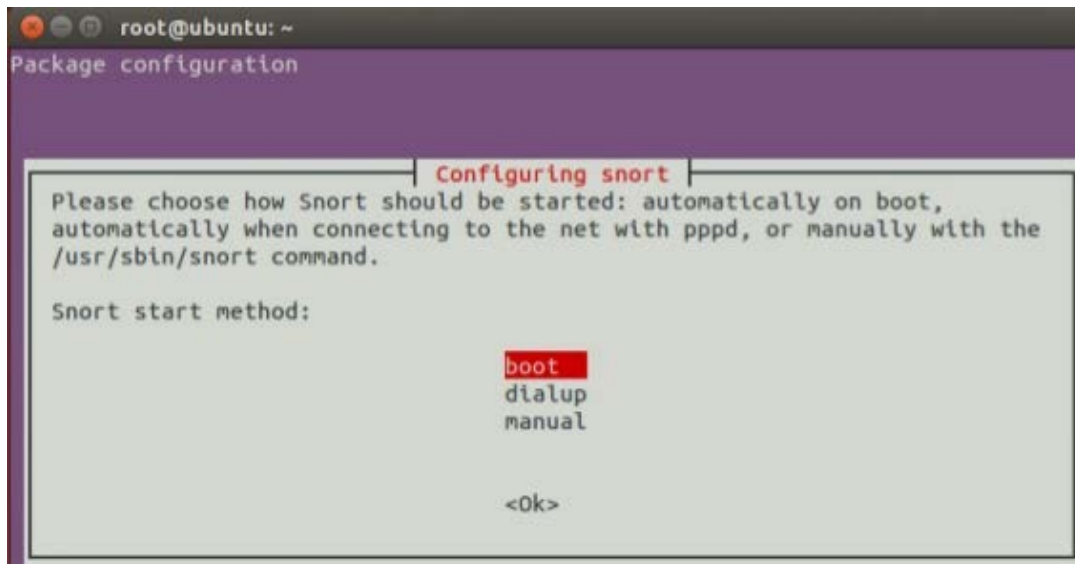


Figure IV.4: Choisir une interface réseau par défaut.

Nous choisissons « boot » pour passer à l'étape suivante :



Figure IV.5: Choisir le mode de démarrage de snort.

Nous laissons l'interface par défaut et cliquer sur suivant :

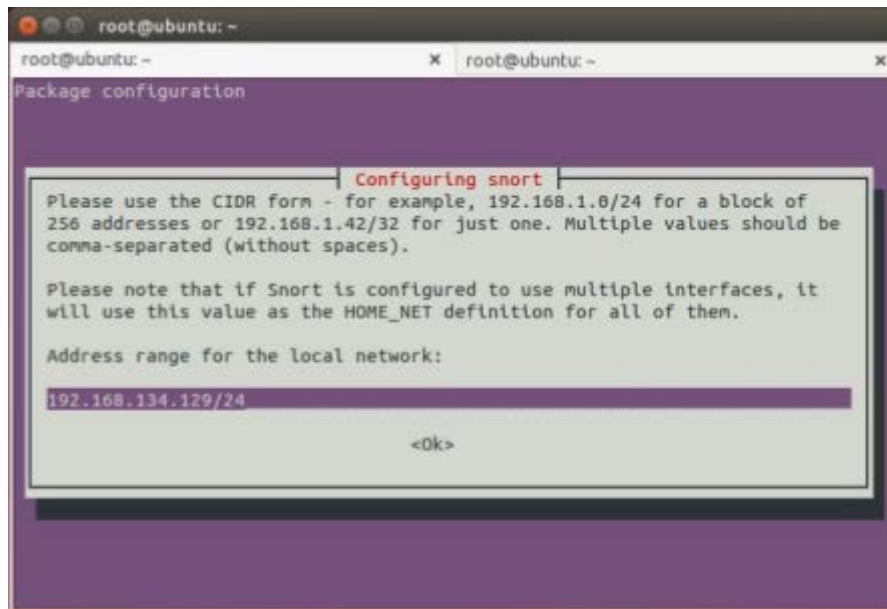


Figure IV.6: Récupération d'adresse ip par défaut.

Ici avant de mettre cette adresse, nous devons faire 'ifconfig' dans notre terminal pour récupérer l'adresse IP.

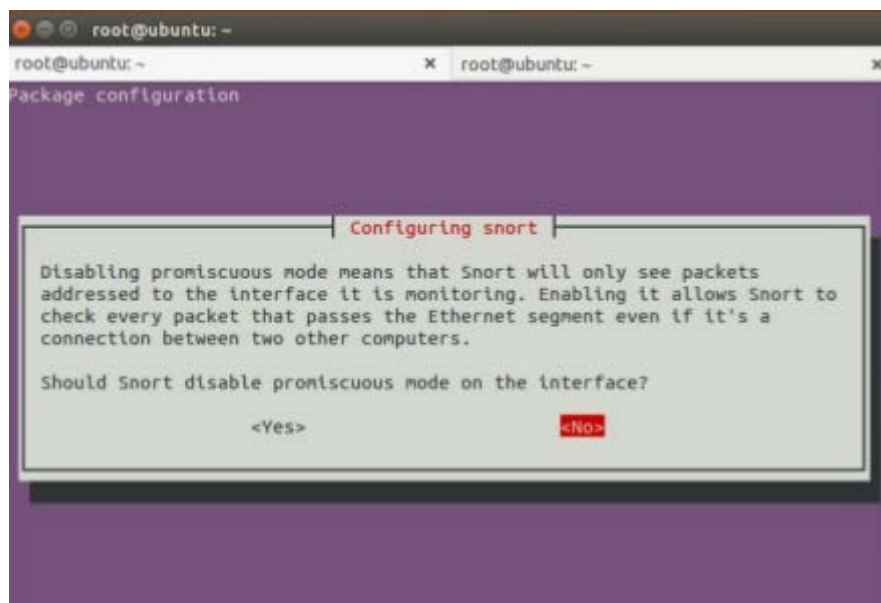


Figure IV.7: Le paramétrage du mode.

Nous choisissons « no » pour ne pas paramétrer le mode.

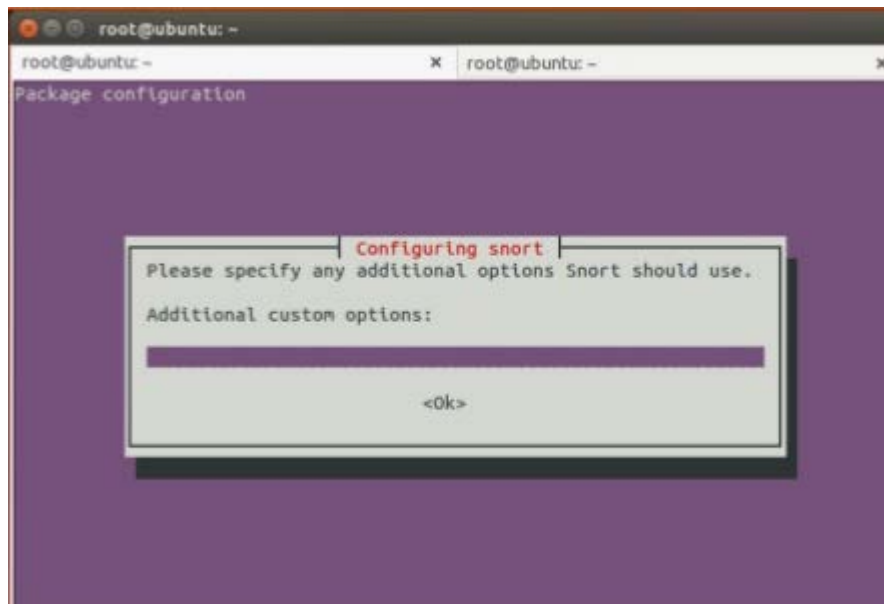


Figure IV.8: Suggestion d'ajout d'autres options pour snort.

Nous laissons cette partie par défaut et on fait suivant.

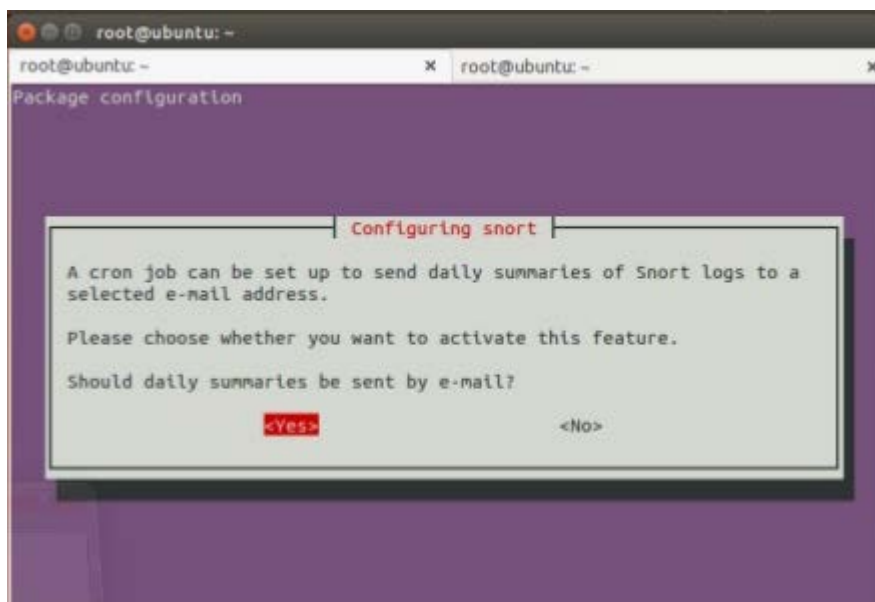


Figure IV.9: Le choix de recevoir des alertes par e-mail.

Nous choisissons « yes » si on souhaite recevoir les alertes par mail sinon nous cliquons sur « no ».

2.3.4. Installation du snort-mysql :

Après la compilation, il nous faut installer le daemon snort-mysql. Nous allons installer snort-mysql en ligne de commande :

```
# apt-get install snort-mysql
```

Lors de l'installation, une interface graphique s'ouvre on fait entrer pour laisser continuer l'installation.

Une fois l'installation terminée, on revient sur notre terminal pour taper : **# dpkg-reconfigure snort-mysql**. Nous serons redirigés vers la fenêtre suivante :

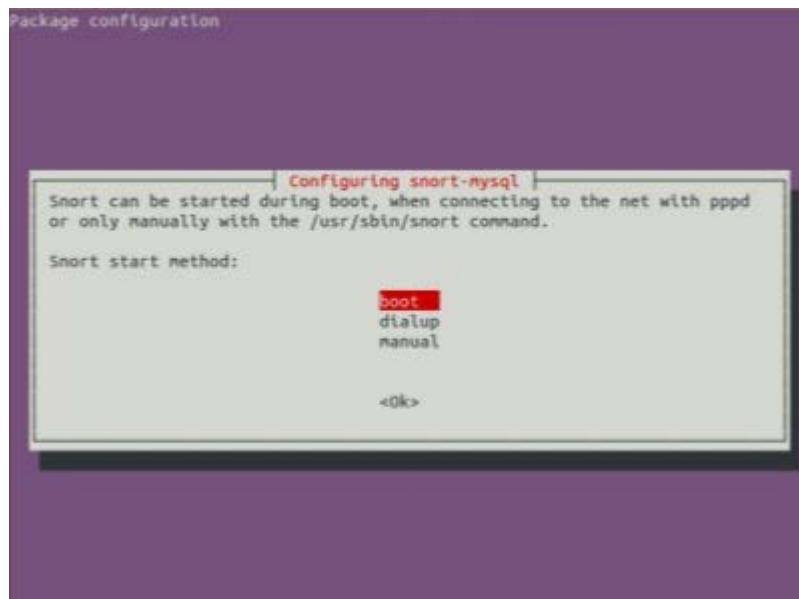


Figure IV.10: Choisir une interface par défaut.

On choisit « boot » pour poursuivre l'installation.

Ici nous entrons l'adresse IP de notre serveur

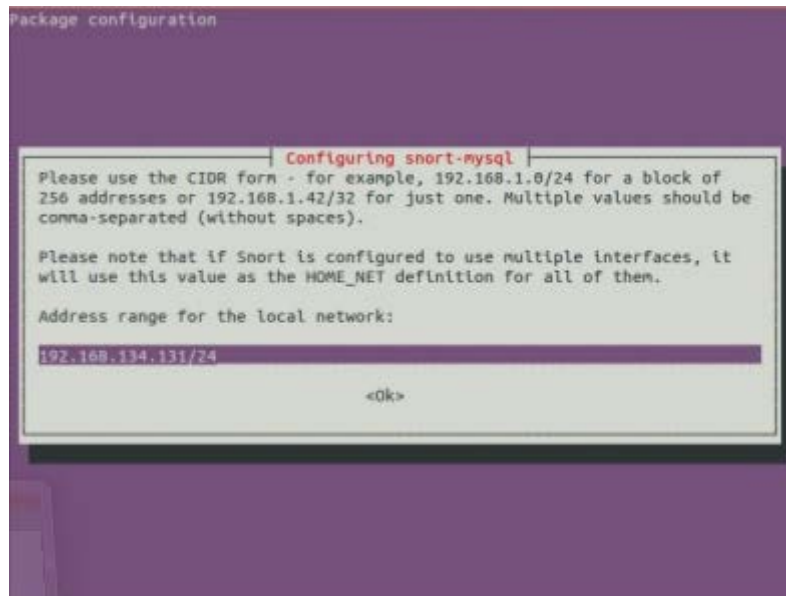


Figure IV.11: Entrer l'adresse ip du serveur.

Nous choisissons « no » puis suivant.

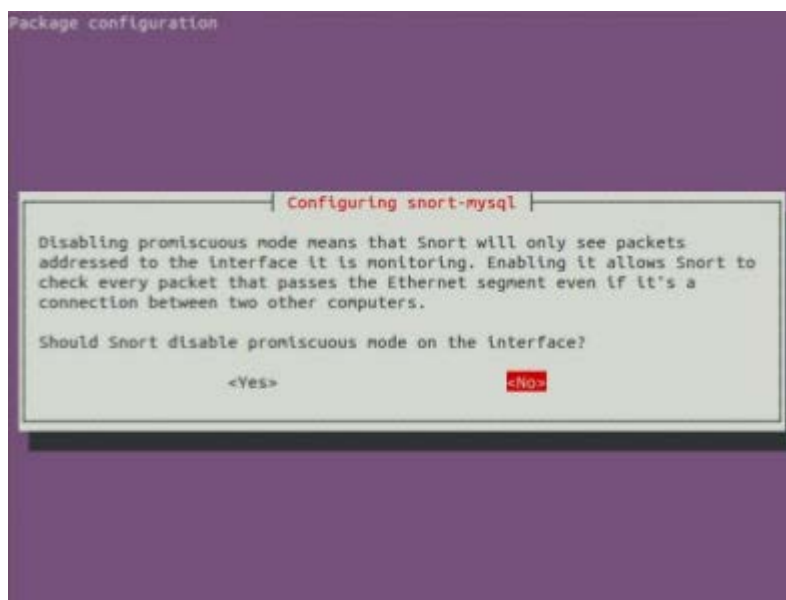


Figure IV.12: Paramétrage du mode.

Nous choisissons « no » pour passer au renseignement de la base de donnée.

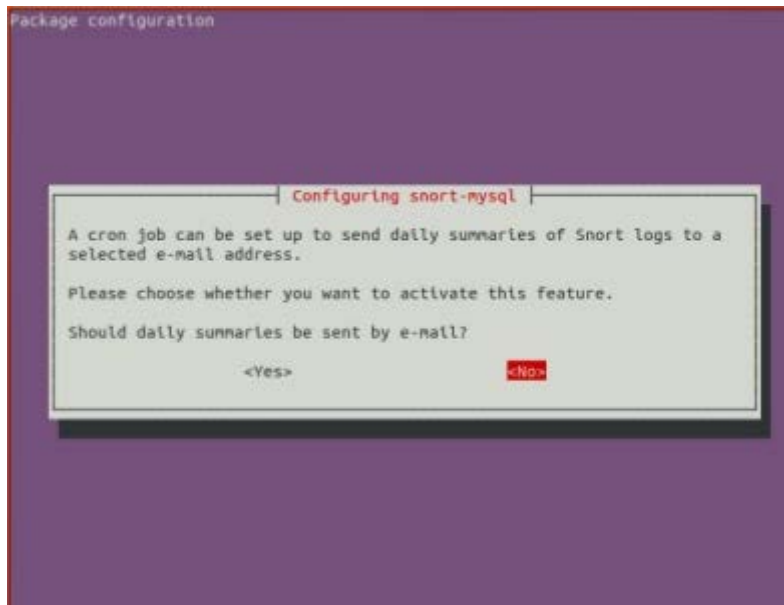


Figure IV.13: Le choix de recevoir des alertes par e-mail.

Nous choisissons « yes » pour renseigner la base de données.

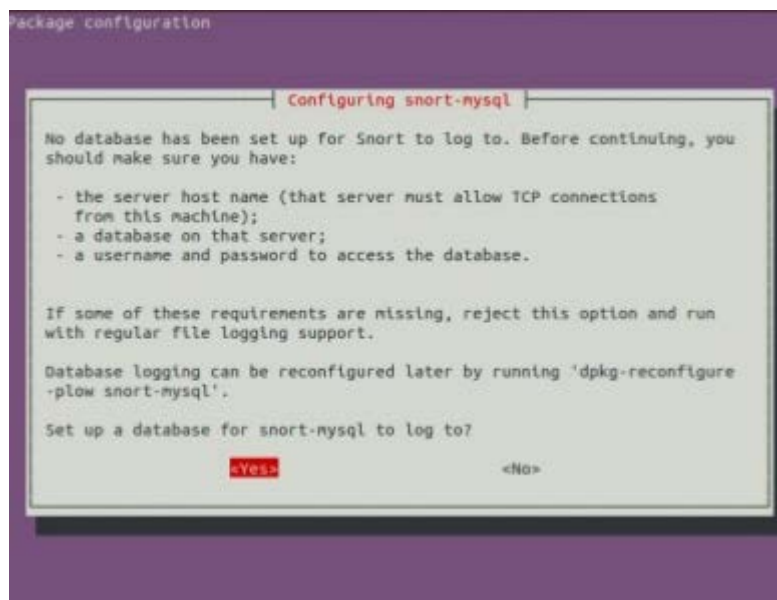


Figure IV.14: Vérification de la base de données.

En suite on met localhost comme le nom d'hôte du serveur de base de donnée.

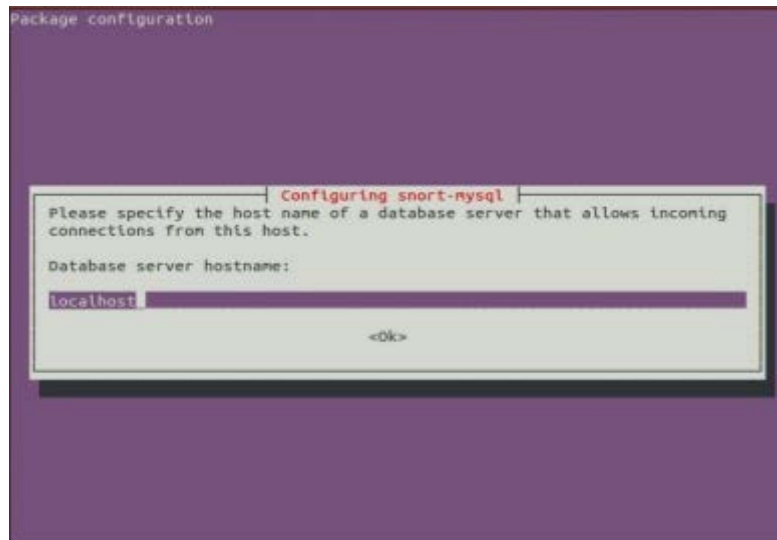


Figure IV.15: Donner un nom à la machine serveur.

Le nom de la base de donnée, ici snort.

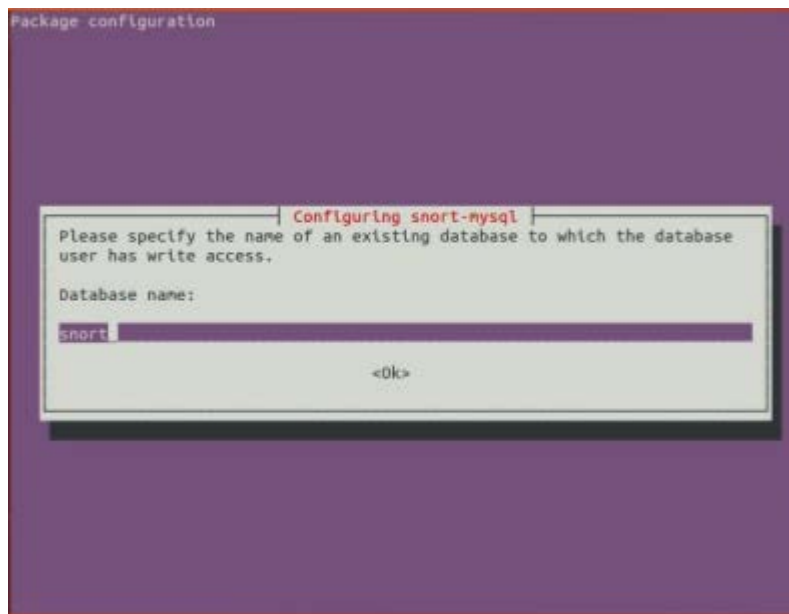


Figure IV.16: Donner un nom à la base de données.

Nous mettons le nom d'utilisateur, nous l'avons nommé snort.

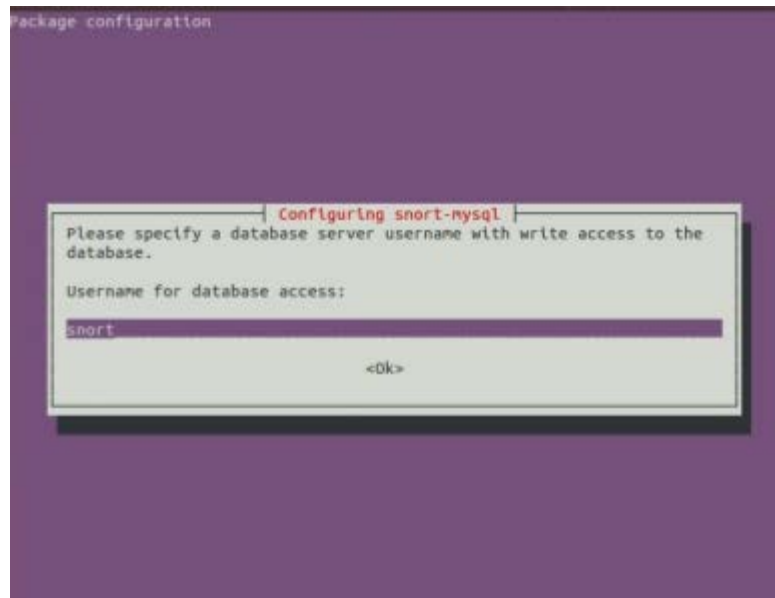


Figure IV.17: Donner un nom à l'utilisateur.

Nous renseignons ensuite le mot de passe.

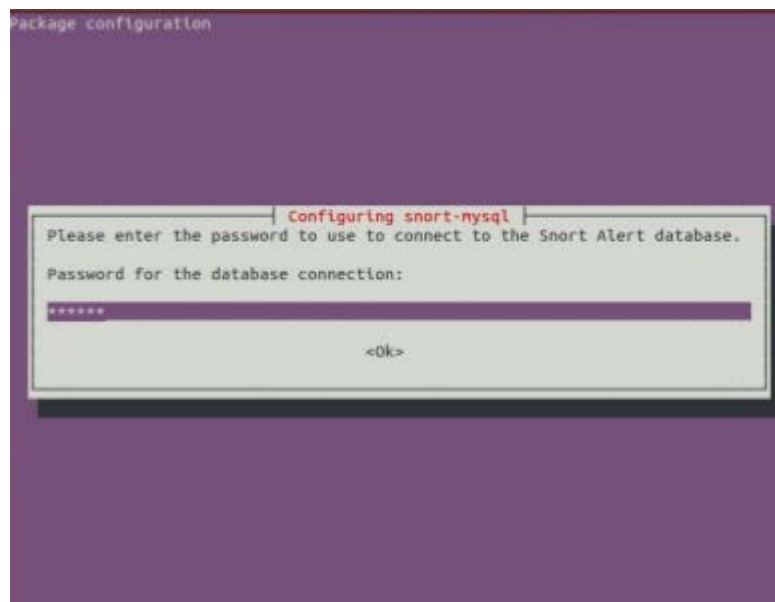


Figure IV.18: Accorder un mot de passe lors de connexion.

Enfin nous validons les informations pour quitter.

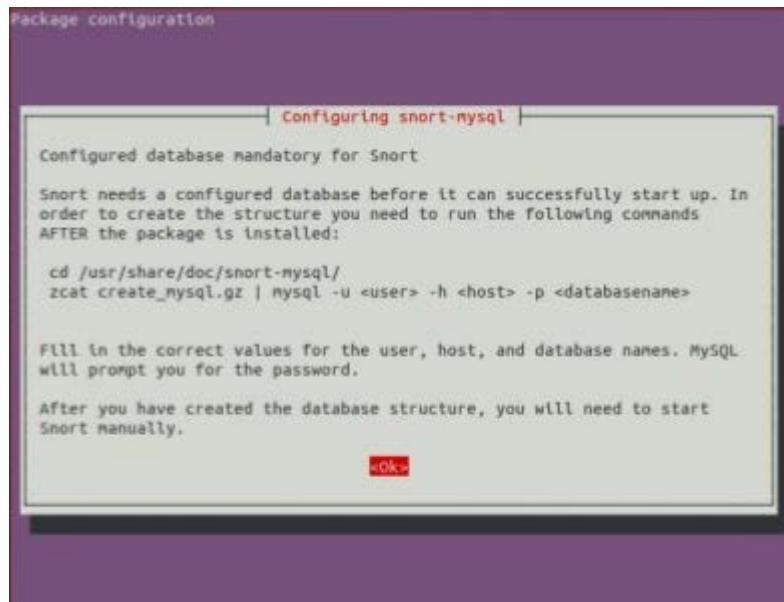


Figure IV.19: La fin de configuration du mysql.

2.3.5. Configuration de Snort :

Nous allons découvrir Snort ainsi que ses fichiers de configuration, Snort comporte plusieurs fichiers de configuration. Les plus importants sont :

/etc/snort/snort.conf (fichier de configuration principal) et les fichiers **.rules** se trouvant dans le répertoire **/etc/snort/rules/**.

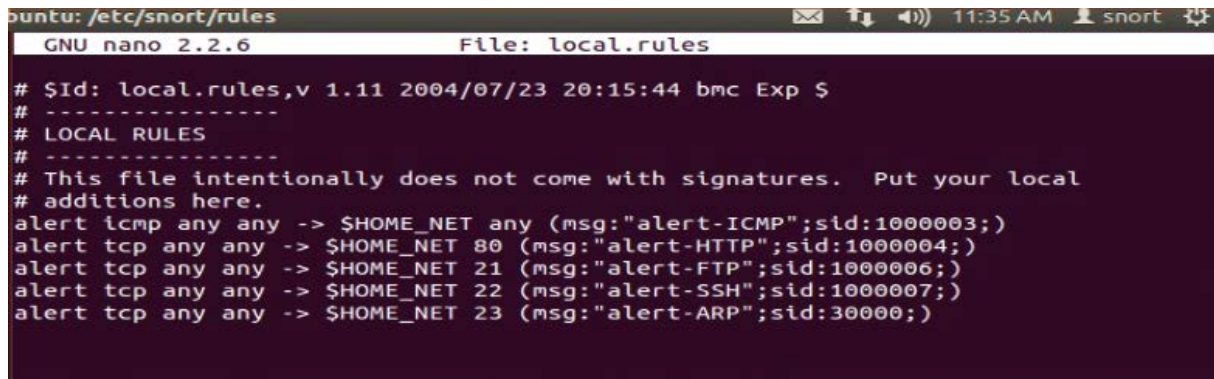
Editons le fichier **/etc/snort/snort.conf**

Nous devons chercher et modifier les lignes qui suivent :

- **var HOME_NET any** : indique l'adresse de l'interface réseau qui écoute le trafic. La valeur par défaut est any. On peut personnaliser en mettant l'adresse ip de l'interface ou du réseau à protéger.
- **var RULE_PATH/etc/snort/rules/** : ici nous précisons le répertoire où se trouvent les fichiers **.rules**.
- nous devons dé-commenter cette ligne et la modifier :
- **output database: alert,mysql,user=snort password=passerdbname=snort host=localhost** : ici nous indiquons à snort qu'il faut enregistrer les événements dans une base de donnée mysql avec les paramètres ci-dessus. Le nom de la base de donnée **snort**, le nom d'utilisateur **snort** et le mot de passe **snort**. Vous pouvez choisir les noms qui vous conviennent.

2.3.6. Création de règles :

Avant de configurer mysql, nous allons créer des règles qui vont permettre à notre serveur de détecter les requêtes qui viennent de l'extérieur. Pour ce faire, éditons le fichier `/etc/snort/local.rules` remplissons le comme suit :



```

ubuntu: /etc/snort/rules
GNU nano 2.2.6      File: local.rules
# $Id: local.rules,v 1.11 2004/07/23 20:15:44 bmc Exp $
# -----
# LOCAL RULES
# -----
# This file intentionally does not come with signatures.  Put your local
# additions here.
alert icmp any any -> $HOME_NET any (msg:"alert-ICMP";sid:1000003;)
alert tcp any any -> $HOME_NET 80 (msg:"alert-HTTP";sid:1000004;)
alert tcp any any -> $HOME_NET 21 (msg:"alert-FTP";sid:1000006;)
alert tcp any any -> $HOME_NET 22 (msg:"alert-SSH";sid:1000007;)
alert tcp any any -> $HOME_NET 23 (msg:"alert-ARP";sid:30000;)

```

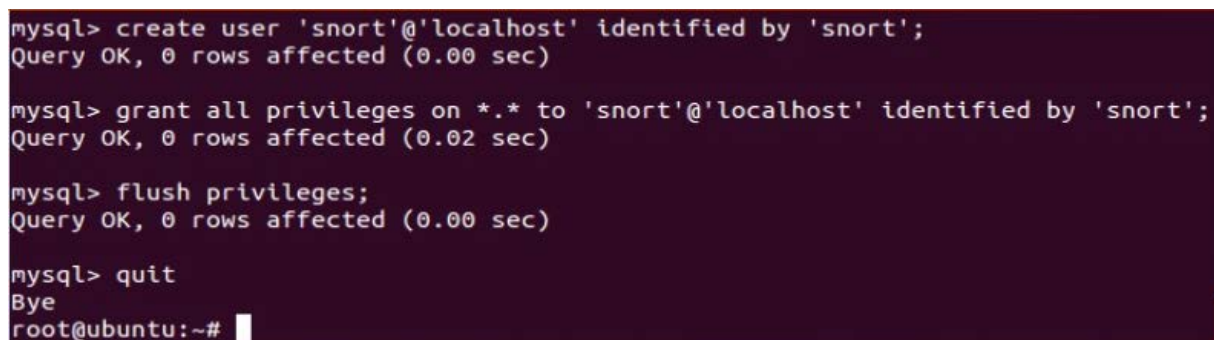
Figure IV.20: Remplissage du fichier 'local.rules' édité.

Cela veut dire tout trafic ICMP ou TCP venant de n'importe où vers n'importe quelle destination entraîne une alerte de type HTTP, FTP, SSH, ARP...

Nous allons configurer MYSQL pour stocker les alertes et autres événements générés par snort. Pour cela on se connecte à la base de données MYSQL entant que root :

Mysql -uroot -hlocalhost -ppasser

Et nous créons la base de donnée comme suit :



```

mysql> create user 'snort'@'localhost' identified by 'snort';
Query OK, 0 rows affected (0.00 sec)

mysql> grant all privileges on *.* to 'snort'@'localhost' identified by 'snort';
Query OK, 0 rows affected (0.02 sec)

mysql> flush privileges;
Query OK, 0 rows affected (0.00 sec)

mysql> quit
Bye
root@ubuntu:~#

```

Figure IV.21: Création de la base de donnée.

Une fois la base de données est crée, nous devons l'activer en insérant les commandes de bases, nous se rendons dans `/usr/share/doc/snort-mysql` puis faire entrer la commande suivante :

```
root@ubuntu:/usr/share/doc/snort-mysql# zcat create_mysql.gz
```

Figure IV.22: Activation de la base de données.

Après, nous devons modifier le fichier de notre **acide base** pour voir si la base de donnée a été prise en compte. Nous éditons le fichier : **/etc/acidbase/database.php** et nous apportons les modifications suivantes :



```
root@ubuntu:/usr/share/doc/snort-mysql
GNU nano 2.2.6 File: /etc/acidbase/database.php

#php
##
## database access settings in php format
## automatically generated from /etc/dbconfig-common/acidbase.conf
## by /usr/sbin/dbconfig-generate-include
## Wed, 15 Jul 2015 10:02:05 -0700
##
## by default this file is managed via ucf, so you shouldn't have to
## worry about manual changes being silently discarded. *however*,
## you'll probably also want to edit the configuration file mentioned
## above too.
##
$alert_user='snort';
$alert_password='passer';
$dbsepath='';
$alert_dbname='snort';
$alert_host='localhost';
$alert_port='3306';
$dbtype='mysql';
```

Figure IV.23: Modification du fichier acide base.

2.4. Installation et configuration de BASE :

BASE est un utilitaire graphique, écrit en php, qui va nous permettre de visualiser à temps réel les alertes et d'autres informations que snort va enregistrer dans la base de donnée MYSQL.

Son installation est simple, il suffit de taper sur le terminal cette commande :

```
# apt-get install acidbase
```

Une fois l'installation est faite, nous éditons le fichier **/etc/acidbase/database.php** dont nous avons fait un peu plus haut pour vérifier s'il a pris en compte notre base de donnée.

Une fois l'installation est terminée, nous ouvrons un navigateur puis faire entrer ce lien pour accéder a l'interface graphique de notre acidbase : **http://localhost.acidbase**



Figure IV.24: Accéder à l'interface graphique d'acide base.

Nous cliquons sur **Create BASE AG** pour voir les paramètres configurés avec succès : (partie en rouge)

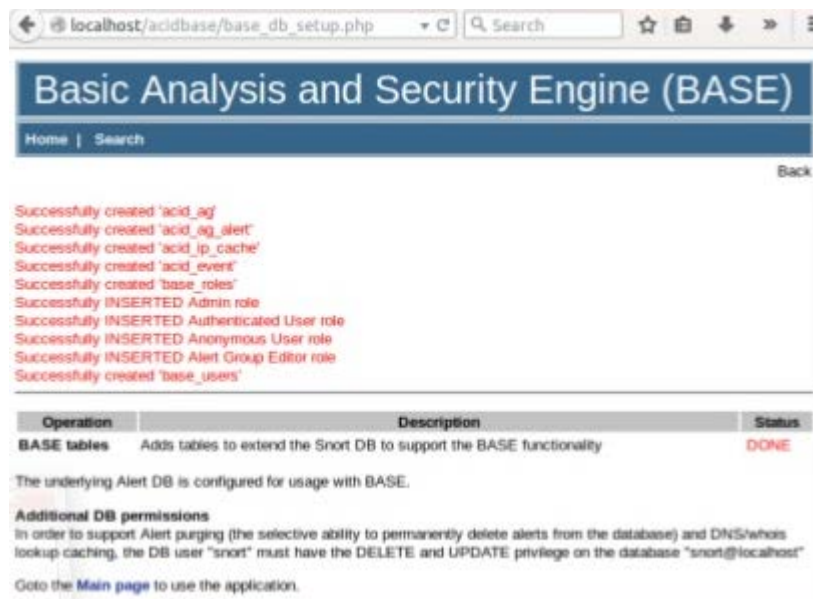


Figure IV.25: L'affichage des paramètres configurés.

A ce niveau, l'application base est bien installée. Nous allons nous connecter sur base en tant que root, nous pouvant créer aussi un autre utilisateur différent de root. Après une connexion, voici les premières informations qui s'affichent.

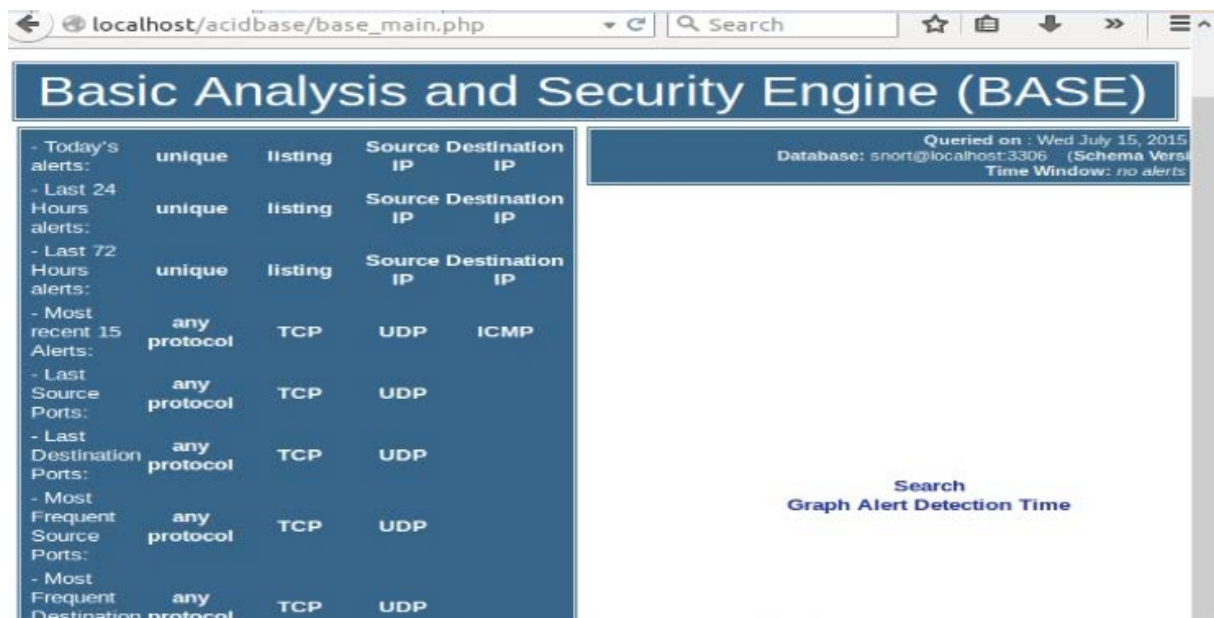


Figure IV.26: Connexion à BASE autant qu'utilisateur.

2.5. Test de fonctionnement :

Après avoir installer et configurer tous les paquets nécessaires, nous allons voir maintenant comment notre système fonctionne. Pour passer au test, nous devons nous assurer que la machine serveur est connectée à internet. Nous devons installer également une machine cliente qui va jouer le rôle d'attaquant. Dans notre cas, nous avons installé **BacTrack**. Nous devons lancer une requête soit ICMP soit ARP et autres depuis cette machine à l'adresse de notre serveur snort pour voir s'il va détecter les requêtes qui viendront de l'extérieur ensuite d'alerter l'administrateur.

Nous allons commencer par démarrer tous nos services :

```
root@ubuntu:/usr/share/doc/snort-mysql# /etc/init.d/snort restart
* Stopping Network Intrusion Detection System snort [ OK ]
* Starting Network Intrusion Detection System snort [ OK ]
root@ubuntu:/usr/share/doc/snort-mysql# /etc/init.d/apache2 restart
* Restarting web server apache2
apache2: Could not reliably determine the server's fully qualified domain name, u
sing 127.0.1.1 for ServerName
... waiting apache2: Could not reliably determine the server's fully qualified d
omain name, using 127.0.1.1 for ServerName [ OK ]
root@ubuntu:/usr/share/doc/snort-mysql#
```

Figure IV.27 : Démarrage des services de la machine.

Nous lançons notre serveur par cette commande : **snort -v**

A terminal window with a dark purple background. The text is white. It shows the output of the command 'snort -v'. The output includes initialization messages, version information (2.9.2), and a sample packet capture showing a UDP packet from 192.168.134.1 to 239.192.152.143.

```
root@ubuntu:/home/kpaya# snort -v
Running in packet dump mode

--== Initializing Snort ==--
Initializing Output Plugins!
pcap DAQ configured to passive.
The DAQ version does not support reload.
Acquiring network traffic from "eth0".
Decoding Ethernet

--== Initialization Complete ==--

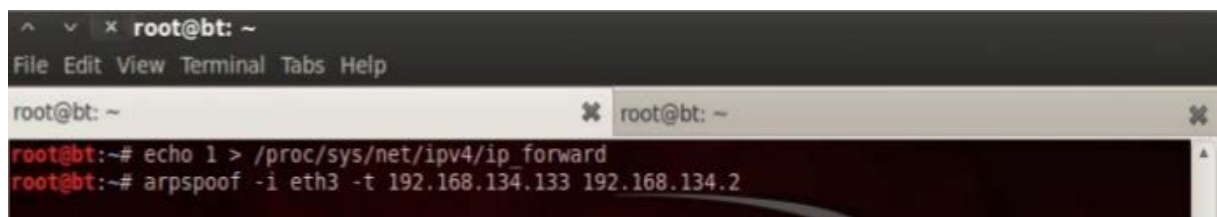
--> Snort! <*-
o"  )- Version 2.9.2 IPv6 GRE (Build 78)
'-'   By Martin Roesch & The Snort Team: http://www.snort.org/snort-snort-t
eam

Copyright (C) 1998-2011 Sourcefire, Inc., et al.
Using libpcap version 1.1.1
Using PCRE version: 8.12 2011-01-15
Using ZLIB version: 1.2.3.4

Commencing packet processing (pid=96389)
07/14-21:07:29.049669 192.168.134.1:6771 -> 239.192.152.143:6771
UDP TTL:255 TOS:0x0 ID:15543 IpLen:20 DgmLen:147
Len: 119
+++++
```

Figure IV.28: Lancement du snort.

Ensuite du côté attaquant, nous ouvrirons le terminal pour entrer ses scripts :

A terminal window with a dark background. The text is white. It shows two commands being executed: 'echo 1 > /proc/sys/net/ipv4/ip_forward' and 'arp spoof -i eth3 -t 192.168.134.133 192.168.134.2'.

```
root@bt: ~
File Edit View Terminal Tabs Help

root@bt: ~
root@bt:~# echo 1 > /proc/sys/net/ipv4/ip_forward
root@bt:~# arp spoof -i eth3 -t 192.168.134.133 192.168.134.2
```

L'adresse 192.168.134.133 est l'adresse de notre serveur snort et 192.168.134.2 c'est celle de passerelle par défaut.

Maintenant, nous devons repartir sur notre serveur et cliquer sur l'icône de notre connexion internet et récupérer les adresses soulignées en bleu dans la figure suivante :

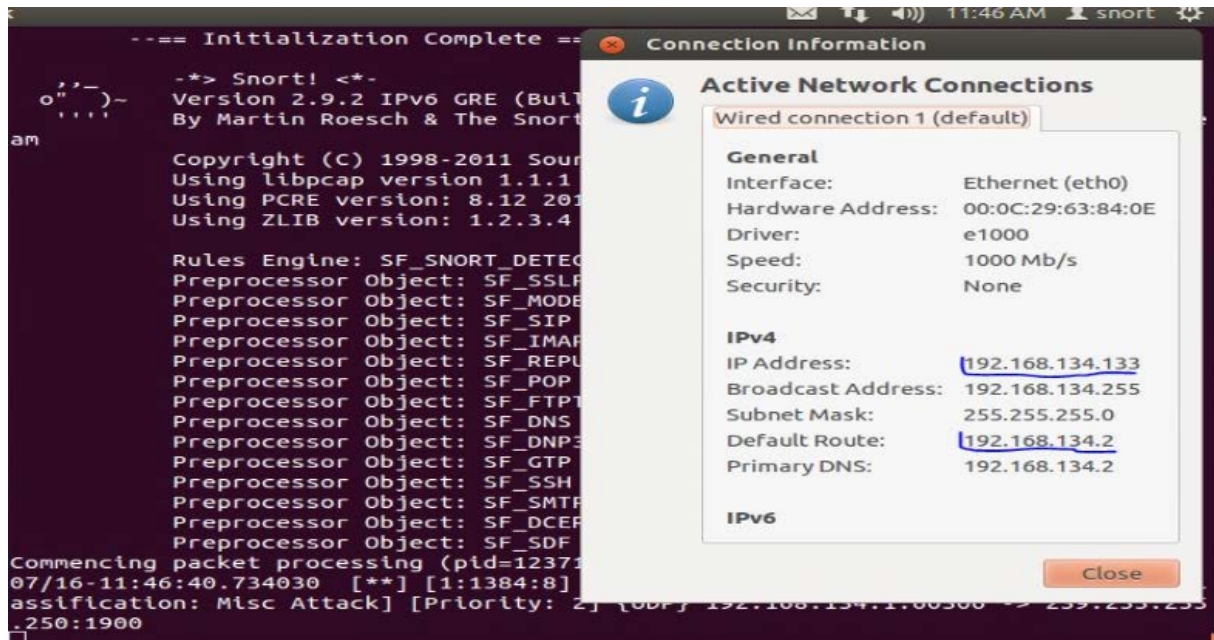


Figure IV.29: Récupération de l'adresse IP du serveur et de la passerelle

Après avoir entré les commandes ci-haut, nous lançons une requête ARP à notre serveur par cette commande : `arp spoof -i eth0 -t 192.168.134.133 192.168.134.2`.

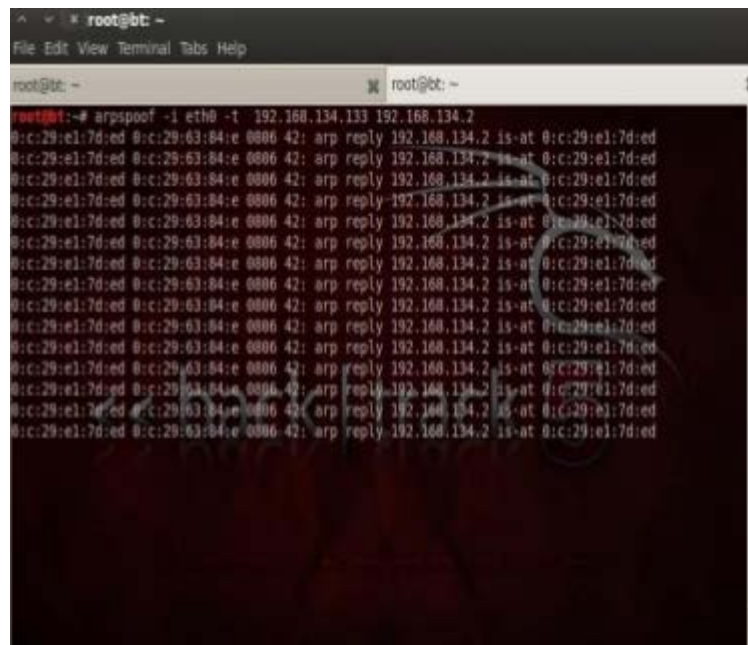
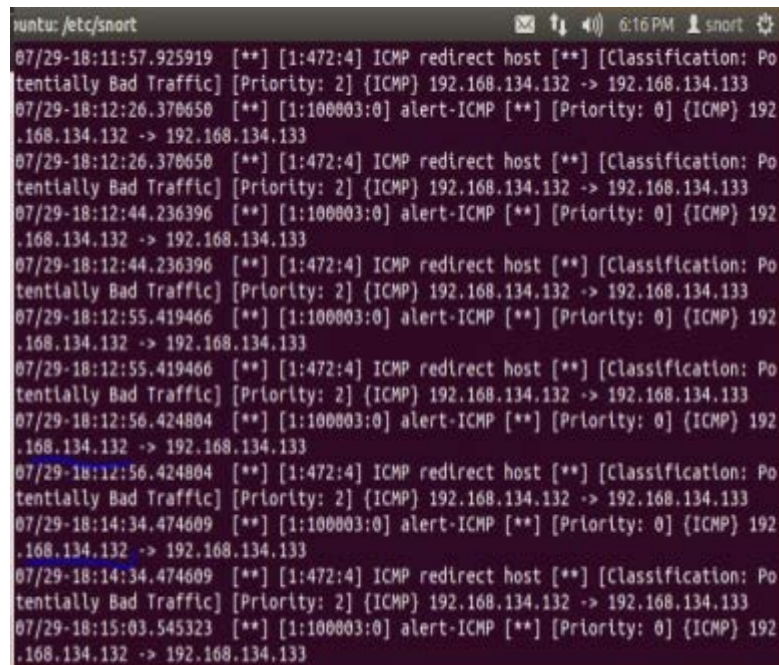


Figure IV.30: Lancement d'une requête ARP



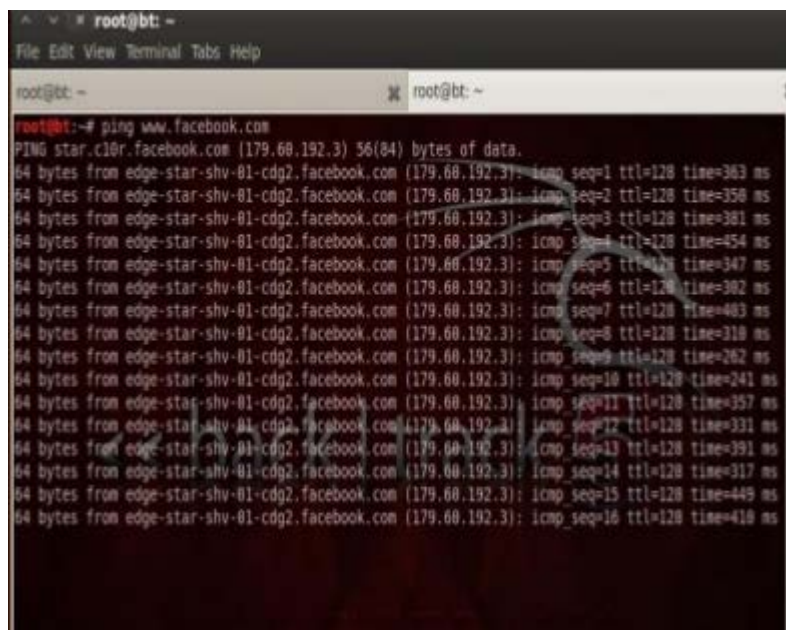
```

ubuntu:/etc/snort
07/29-18:11:57.925919  [**] [1:472:4] ICMP redirect host [**] [Classification: Po
tentially Bad Traffic] [Priority: 2] {ICMP} 192.168.134.132 -> 192.168.134.133
07/29-18:12:26.370650  [**] [1:100003:0] alert-ICMP [**] [Priority: 0] {ICMP} 192
.168.134.132 -> 192.168.134.133
07/29-18:12:26.370650  [**] [1:472:4] ICMP redirect host [**] [Classification: Po
tentially Bad Traffic] [Priority: 2] {ICMP} 192.168.134.132 -> 192.168.134.133
07/29-18:12:44.236396  [**] [1:100003:0] alert-ICMP [**] [Priority: 0] {ICMP} 192
.168.134.132 -> 192.168.134.133
07/29-18:12:44.236396  [**] [1:472:4] ICMP redirect host [**] [Classification: Po
tentially Bad Traffic] [Priority: 2] {ICMP} 192.168.134.132 -> 192.168.134.133
07/29-18:12:55.419466  [**] [1:100003:0] alert-ICMP [**] [Priority: 0] {ICMP} 192
.168.134.132 -> 192.168.134.133
07/29-18:12:55.419466  [**] [1:472:4] ICMP redirect host [**] [Classification: Po
tentially Bad Traffic] [Priority: 2] {ICMP} 192.168.134.132 -> 192.168.134.133
07/29-18:12:56.424804  [**] [1:100003:0] alert-ICMP [**] [Priority: 0] {ICMP} 192
.168.134.132 -> 192.168.134.133
07/29-18:12:56.424804  [**] [1:472:4] ICMP redirect host [**] [Classification: Po
tentially Bad Traffic] [Priority: 2] {ICMP} 192.168.134.132 -> 192.168.134.133
07/29-18:14:34.474609  [**] [1:100003:0] alert-ICMP [**] [Priority: 0] {ICMP} 192
.168.134.132 -> 192.168.134.133
07/29-18:14:34.474609  [**] [1:472:4] ICMP redirect host [**] [Classification: Po
tentially Bad Traffic] [Priority: 2] {ICMP} 192.168.134.132 -> 192.168.134.133
07/29-18:15:03.545323  [**] [1:100003:0] alert-ICMP [**] [Priority: 0] {ICMP} 192
.168.134.132 -> 192.168.134.133

```

Figure IV.31: Etablir la connexion entre les deux machines.

Lançons une requête ICMP depuis la machine cliente



```

root@bt: ~
File Edit View Terminal Tabs Help

root@bt: ~
root@bt:~# ping www.facebook.com
PING star.c10r.facebook.com (179.60.192.3) 56(84) bytes of data:
64 bytes from edge-star-shv-01-cdg2.facebook.com (179.60.192.3): icmp_seq=1 ttl=128 time=363 ms
64 bytes from edge-star-shv-01-cdg2.facebook.com (179.60.192.3): icmp_seq=2 ttl=128 time=358 ms
64 bytes from edge-star-shv-01-cdg2.facebook.com (179.60.192.3): icmp_seq=3 ttl=128 time=381 ms
64 bytes from edge-star-shv-01-cdg2.facebook.com (179.60.192.3): icmp_seq=4 ttl=128 time=454 ms
64 bytes from edge-star-shv-01-cdg2.facebook.com (179.60.192.3): icmp_seq=5 ttl=128 time=347 ms
64 bytes from edge-star-shv-01-cdg2.facebook.com (179.60.192.3): icmp_seq=6 ttl=128 time=382 ms
64 bytes from edge-star-shv-01-cdg2.facebook.com (179.60.192.3): icmp_seq=7 ttl=128 time=403 ms
64 bytes from edge-star-shv-01-cdg2.facebook.com (179.60.192.3): icmp_seq=8 ttl=128 time=318 ms
64 bytes from edge-star-shv-01-cdg2.facebook.com (179.60.192.3): icmp_seq=9 ttl=128 time=262 ms
64 bytes from edge-star-shv-01-cdg2.facebook.com (179.60.192.3): icmp_seq=10 ttl=128 time=241 ms
64 bytes from edge-star-shv-01-cdg2.facebook.com (179.60.192.3): icmp_seq=11 ttl=128 time=357 ms
64 bytes from edge-star-shv-01-cdg2.facebook.com (179.60.192.3): icmp_seq=12 ttl=128 time=331 ms
64 bytes from edge-star-shv-01-cdg2.facebook.com (179.60.192.3): icmp_seq=13 ttl=128 time=391 ms
64 bytes from edge-star-shv-01-cdg2.facebook.com (179.60.192.3): icmp_seq=14 ttl=128 time=317 ms
64 bytes from edge-star-shv-01-cdg2.facebook.com (179.60.192.3): icmp_seq=15 ttl=128 time=449 ms
64 bytes from edge-star-shv-01-cdg2.facebook.com (179.60.192.3): icmp_seq=16 ttl=128 time=418 ms

```

Figure IV.32: Lancement d'une requête ICMP.

```

juntu: /etc/snort
07/29-18:11:57.925919 00000000 [1:472:4] ICMP redirect host [00000000] [Classification: Potentially Bad Traffic] [Priority: 2] (ICMP) 192.168.134.132 -> 192.168.134.133
07/29-18:12:26.370650 00000000 [1:100003:0] alert-ICMP [00000000] [Priority: 0] (ICMP) 192.168.134.132 -> 192.168.134.133
07/29-18:12:26.370650 00000000 [1:472:4] ICMP redirect host [00000000] [Classification: Potentially Bad Traffic] [Priority: 2] (ICMP) 192.168.134.132 -> 192.168.134.133
07/29-18:12:44.236396 00000000 [1:100003:0] alert-ICMP [00000000] [Priority: 0] (ICMP) 192.168.134.132 -> 192.168.134.133
07/29-18:12:44.236396 00000000 [1:472:4] ICMP redirect host [00000000] [Classification: Potentially Bad Traffic] [Priority: 2] (ICMP) 192.168.134.132 -> 192.168.134.133
07/29-18:12:55.419466 00000000 [1:100003:0] alert-ICMP [00000000] [Priority: 0] (ICMP) 192.168.134.132 -> 192.168.134.133
07/29-18:12:55.419466 00000000 [1:472:4] ICMP redirect host [00000000] [Classification: Potentially Bad Traffic] [Priority: 2] (ICMP) 192.168.134.132 -> 192.168.134.133
07/29-18:12:56.424804 00000000 [1:100003:0] alert-ICMP [00000000] [Priority: 0] (ICMP) 192.168.134.132 -> 192.168.134.133
07/29-18:12:56.424804 00000000 [1:472:4] ICMP redirect host [00000000] [Classification: Potentially Bad Traffic] [Priority: 2] (ICMP) 192.168.134.132 -> 192.168.134.133
07/29-18:14:34.474609 00000000 [1:100003:0] alert-ICMP [00000000] [Priority: 0] (ICMP) 192.168.134.132 -> 192.168.134.133
07/29-18:14:34.474609 00000000 [1:472:4] ICMP redirect host [00000000] [Classification: Potentially Bad Traffic] [Priority: 2] (ICMP) 192.168.134.132 -> 192.168.134.133
07/29-18:15:03.545323 00000000 [1:100003:0] alert-ICMP [00000000] [Priority: 0] (ICMP) 192.168.134.132 -> 192.168.134.133
07/29-18:15:03.545323 00000000 [1:472:4] ICMP redirect host [00000000] [Classification: Potentially Bad Traffic] [Priority: 2] (ICMP) 192.168.134.132 -> 192.168.134.133

```

Figure IV.33: Recevoir une alerte ICMP.

2.6. Résultats analysés par BASE pendant et après le test :

Nous voyons dans la figure suivante que la base a analysé deux requêtes : TCP et ICMP lors du test.

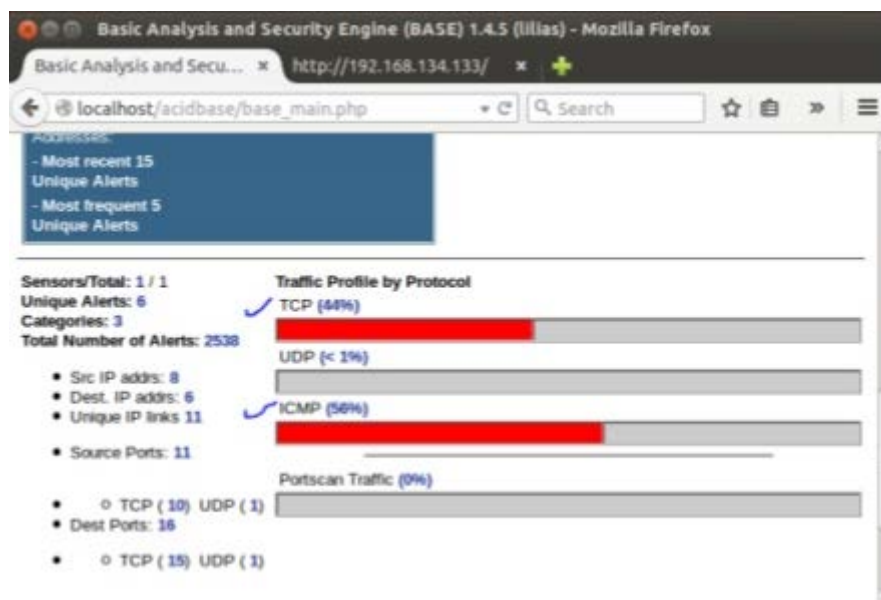


Figure IV.34: L'analyse des requêtes TCP et ICMP.

☐	< Src IP address >	Sensor #	< Total # >	< Unique Alerts >	< Dest. Addr. >
☐	81.52.160.34	1	4956	2	1
☐	81.52.160.43	1	2589	2	1
☐	179.60.192.3	1	1656	3	1
☐	8.8.8.8	1	1194	2	2
☐	173.194.113.242	1	471	1	1
☐	173.194.113.240	1	384	1	1
☐	192.168.134.132	1	366	2	1
☐	90.84.59.115	1	225	1	1
☐	93.184.220.29	1	198	1	1
☐	68.232.34.191	1	186	1	1
☐	41.214.4.19	1	111	1	1
☐	196.207.198.151	1	102	1	1
☐	54.200.104.130	1	90	1	1
☐	52.24.49.210	1	81	1	1
☐	173.194.40.109	1	78	1	1
☐	216.34.181.60	1	72	1	1
☐	196.207.198.187	1	66	1	1
☐	91.189.89.22	1	54	1	1
☐	54.76.56.211	1	45	1	1

Figure IV.35: Gestion des alertes.

Donc snort a détecté des actions qui étaient définies comme une intrusion, et nous remarquerons aussi que trop d'alertes ont été générées par snort. Dans les conditions réelles, il serait difficile de gérer toutes ces alertes.

Discussion

Outre la mise en place des pare-feu et des systèmes d'authentification, il est de nos jours nécessaire de mettre en place un système de détection d'intrusion.

Nous avons étudié le fonctionnement de ces systèmes en particulier nous avons pris comme exemple l'outil Open Source Snort qui est le plus réputé en terme d'efficacité et présente une souplesse en terme de personnalisation.

Pour une meilleure lisibilité dans la lecture des logs et les alertes; il existe plusieurs interfaces graphiques pour Snort, nous avons utilisé BASE. Aussi pour une meilleure gestion des logs, Snort peut être interfacé avec un SGBD (Système de Gestion de Base de Données) comme le MySQL que nous avons utilisé dans notre cas.

Conclusion

Dans ce mémoire, nous avons étudié la mise en place du système de détection d'intrusion avec Snort de manière à pouvoir mettre en place une politique de sécurisation efficace du réseau informatique.

D'abord, nous avons présenté une synthèse sur les réseaux informatique, la connaissance sur la généralité des réseaux informatiques est essentielle pour comprendre le fonctionnement de l'échange de la communication tel que l'importance du modèle de référence OSI ainsi que le modèle TCP/IP. Ensuite, nous avons pris un aperçu sur les failles et attaques qui peuvent menacer la sécurité informatique, ainsi que les mesures de protections connues pour le protéger des pirates. Nous avons également présenté une étude détaillée sur le test d'intrusion en utilisant un système de détection d'intrusion réseau qui est le snort. A la fin nous avons réalisé un test d'intrusion avec le snort.

Le logiciel Snort est un système de détection au niveau réseau (NIDS) et c'est l'outil de base utilisé pour détecter les intrusions. Mais pour assurer son efficacité, Dans une première étape, nous avons installé et configuré snort, nous avons configuré sa base de données afin d'organiser méthodiquement les alertes, puis créer un script en daemon pour notifier automatiquement l'administrateur réseau par l'envoi de SMS en cas d'une tentative d'intrusion. Dans une deuxième étape, nous avons réalisé un test d'intrusion en utilisant deux machines une pour lancé une requête et l'autre pour détecter les alertes lancées afin de vérifier le bon fonctionnement du système de détection d'intrusion réseau.

En fait, grâce à l'IDS Snort, l'entreprise peut établir des politiques de sécurité dynamique et corriger la faille du réseau suivant les types attaques détectées ainsi que de connaître en tout moment si elle est la cible d'une attaque d'intrusion. Toutefois, les IDS permettent de détecter et de journaliser les attaques et s'arrêtent là, tout va dépendre maintenant de la décision prise de l'administrateur. A part la notification en temps réel, il est aussi donc nécessaire de bloquer si besoin les intrusions à l'instant de leur détection. Ainsi, comme perspective d'amélioration de la sécurisation par les systèmes de détection d'intrusion, il faut aussi implémenter des systèmes de prévention d'intrusion qui sont des systèmes capable de réagir contre les attaques dès leurs apparitions.

La sécurité informatique demeure encore un sujet très sensible, voir même complexe. Sachant bien que l'évolution des technologies a permis d'améliorer les mécanismes de sécurité au niveau des réseaux informatiques, il est toujours difficile de la garantir à 100%.

Le test d'intrusion effectué dans notre travail n'est pas le seul existants dans le domaine de la sécurité des réseaux ; du coup et en guise de perspective, nous proposons d'élargir les tests d'intrusion internes et externes à effectuer sur tout les systèmes d'informations d'une entreprise afin d'optimiser sa sécurité globale.

1. Définition du système d'exploitation linux :

GNU/Linux ou Linux est un système d'exploitation associant des éléments essentiels du projet GNU et le noyau Linux.

Fondé en 1984 par Richard Stallman, le système d'exploitation GNU resta jusqu'en 1991 au stade expérimental car son noyau Hurd était toujours en développement. Cependant, Linus Torvalds créa indépendamment le noyau Linux qui résolut le problème en remplaçant Hurd.

Néanmoins, cette mise en relation des deux projets engendra une controverse toujours d'actualité autour du nom du système associant les deux éléments, certains, dont le grand public, appelant le système simplement Linux, et d'autres, dont Richard Stallman, défendant le nom combiné GNU/Linux.

2. VMware :

VMware est une société informatique américaine fondée en 1998, filiale d'EMC Corporation depuis 2004 (racheté par Dell le 7 septembre 2016^{1,2}), qui propose plusieurs produits propriétaires liés à la virtualisation d'architectures x86. C'est aussi par extension le nom d'une gamme de logiciels de virtualisation.

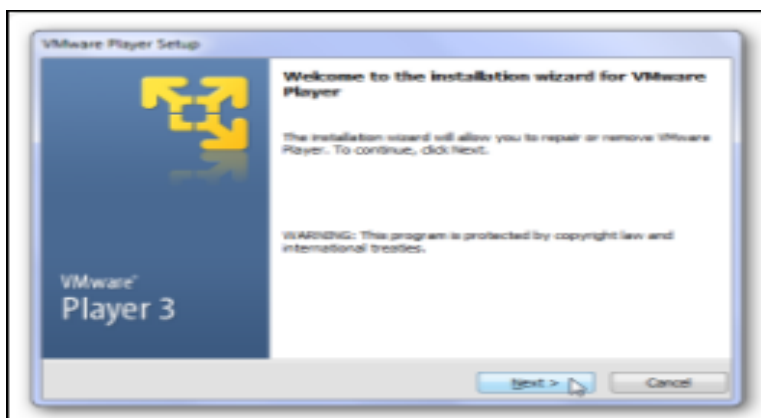
2.1 Fonctionnement :

VMware crée un environnement clos dans lequel sont disponibles un, deux ou quatre processeur(s), des périphériques et un BIOS virtuel. Selon les concepteurs, le microprocesseur n'est émulé que quand c'est nécessaire, c'est-à-dire quand la VM (machine virtuelle) tourne en mode noyau ou en mode réel, mais pas pour le mode utilisateur (user mode) ou le Mode virtuel 8086. Selon les concepteurs, cela permet à VMware d'être plus rapide que des solutions multi-plateformes qui émulent tout. Lorsqu'une VM s'exécute dans un mode qui nécessite une émulation, VMware traduit dynamiquement le code privilégié en un code équivalent en mode utilisateur, le place dans un endroit libre de la mémoire, le rend invisible et inaccessible au code d'origine et l'exécute à la place. Lorsqu'une machine virtuelle fait appel à un périphérique, VMware intercepte la demande et la traduit pour qu'elle soit gérée par le système hôte. Bien que les machines virtuelles tournent en mode utilisateur, VMware nécessite d'installer plusieurs pilotes de périphériques privilégiés dans le noyau du système hôte, qui notamment interchangent les tables GDT et IDT chaque fois qu'on passe la main à une VM. VMware assure l'émulation de la carte vidéo, la carte réseau, le lecteur de CD-ROM, le bus USB, les ports séries et parallèle et le disque dur de type SCSI ou IDE. Ce dernier étant un fichier extensible d'une taille voisine de la place occupée sur la machine virtuelle ou fixe pour davantage de performance. Ce fichier contenant le contenu du disque peut être copié sur un autre hôte et exécuté par un ordinateur. Pour l'ordinateur virtuel, tous les périphériques sont identiques, même si le système hôte est totalement différent, car c'est VMware qui caractérise les périphériques.

3. Installation de Linux (Ubuntu) sous Windows à l'aide d'une machine virtuelle (VMware Player) :

3.1. Installation de VMWare Player :

Téléchargement de VMWare Player : <http://www.vmware.com/go/downloadplayer/>
Le logiciel gratuit nécessite cependant une inscription. Après envoi du formulaire, installation de "VMware Player 4.0.3 for 32-bit and 64-bit Windows" (Appuyez sur "Start Download Manager"). Différentes méthodes d'installation existent. On peut activer l'installateur depuis un pop-up du navigateur. Cliquer sur launch. On arrive dans la fenêtre d'installation de VMWare Player :



Suivre les différentes étapes en appuyant sur "Next" et "Continue". On termine par "Restart Now" qui redémarre Windows, VMware est installé !

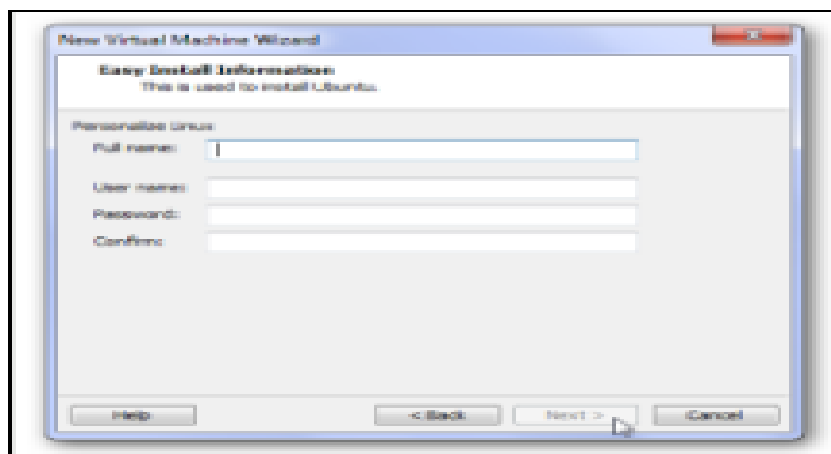
3.2. Installation d'Ubuntu :

Téléchargement de Ubuntu : <http://www.ubuntu.com/getubuntu/download>
Appuyez sur le bouton "Start download" pour télécharger la dernière version de l'image disque d'Ubuntu. Lors de cet essai, le fichier au format iso était nommé "ubuntu-12.04-desktop-i386.iso". Le fichier occupe plus de 700 Mo et cette étape peut s'avérer longue. On enregistre le fichier qui pourra être réutilisé dans un répertoire spécial nommé "Image disque" par exemple. On lance VMware Player et on choisit l'option "Create a New Virtual Machine"



Sélectionner ensuite l'option : "Installer disc image file (iso):" et sélectionner dans "Image disque" le fichier "ubuntu-12.04-desktop-i386.iso". Un formulaire est rempli dont les renseignements servent à l'installation automatique de Ubuntu. Les précautions suivantes doivent être prises concernant mes champs :

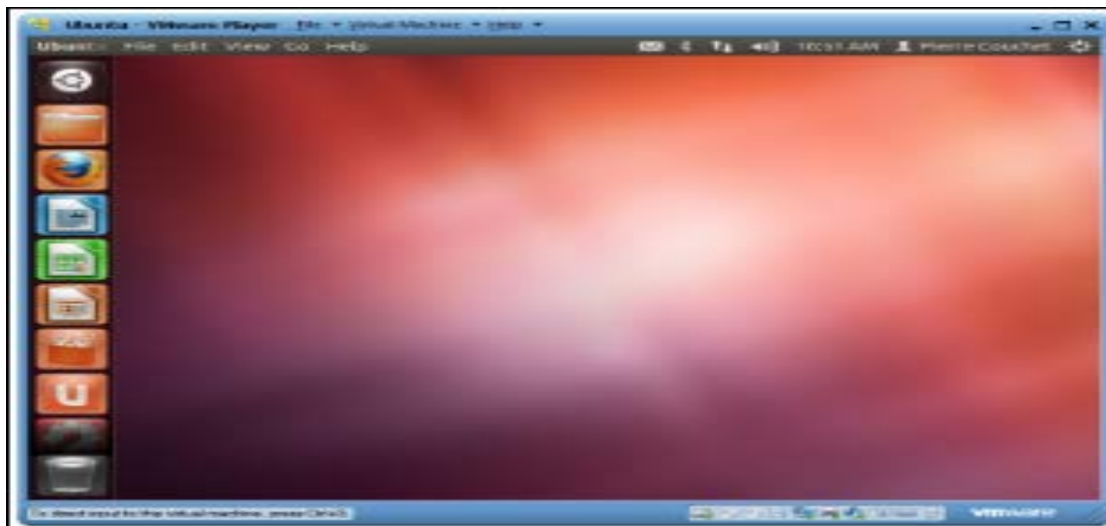
- "User name" ne doit contenir que des minuscules et des chiffres
- Privilégiez pour "Password" les lettres b, c, d, e, f, g, h, i, j, k, l, n, o, p, r, s, t, u, v, x, éviter les lettres a, q, w, z... et les chiffres. Lors du démarrage de Ubuntu, le mot de passe est entré à l'aide d'un clavier configuré en Qwerty, alors que votre clavier est généralement en Azerty. L'emploi de ces lettres évite l'ambiguïté.



Prenez les choix par défaut. Tapez "Next" puis "Finish". Ubuntu s'installe alors dans VMware Player. Une fenêtre demande si l'installation de VMware Tools est souhaitée : cliquez sur "Download". Ces utilitaires facilitent la copie de fichiers entre Windows et Linux. Les paquets s'installent automatiquement et au bout d'un quart d'heure, Ubuntu est installé .



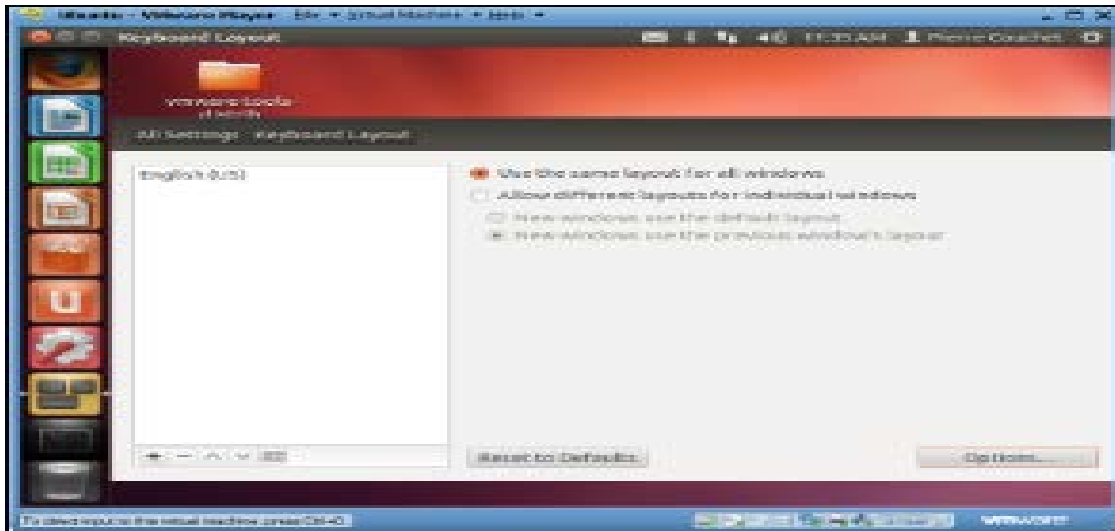
Le clavier est en Qwerty, attention en rentrant le mot de passe saisi précédemment en Azerty !
On arrive sur le bureau “Unity” optimisé pour un affichage sur téléphone portable, tablette ou PC :



3.3. Quelques réglages d’Ubuntu :

Il est alors possible d’améliorer légèrement la machine virtuelle :

– On va tout d’abord changer la configuration du clavier. Sélectionnez l’icone de gauche avec la roue de réglage “System Settings” puis “Keyboard Layout”, appuyez sur le bouton “+” en bas à gauche pour ajouter à la sélection le clavier “French”. Un raccourci en haut à droite du bureau facilite alors le changement de clavier. La langue du bureau peut aussi être changée dans “Langage support”.



– Il est ensuite possible d’optimiser la taille de l’affichage. Toujours dans “System Settings”, on sélectionne dans “Displays” la taille adaptée, appuyez sur “Apply”. On peut aussi passer en mode plein écran en double cliquant sur la barre du haut de VMware player. Touches “Ctrl Alt” pour revenir sous windows.



De nombreuses opérations s’effectuent depuis un terminal permettant l’entrée de lignes de commandes, nécessaires lors de l’installation de logiciels, de bibliothèques et d’extensions. On ouvre un terminal en appuyant simultanément sur “Ctrl Alt T”, ou bien en saisissant “term” dans la recherche des programmes Ubuntu.



Depuis le terminal :

- Entrez en ligne de commande : `sudo passwd` et renseignez le mot de passe deux fois de suite pour activer le compte root.
- Pour devenir utilisateur “root” (administrateur), entrez la commande : `su` et entrez le mot de passe root. L’invite de commande change alors et devient : “root@ubuntu:”
- Pour redevenir utilisateur normal : `exit`, l’invite de commande ou prompt redevient normale.
- Pour exécuter une instruction en ayant les droits de “root” tout en restant utilisateur normal : `sudo instruction`. C’est cette méthode qui sera privilégiée pour éviter de fausses manœuvres dommageables.

– Quelques trucs qui peuvent être utiles :

- Passer VMware plein écran : double clic sur la barre du haut de VMware Player
- Ouvrir un terminal « Ctrl Alt T »
- Passer de VMware à windows : “Ctrl Alt”
- Détruire une machine virtuelle dans VMware player : à l’aide d’un clic droit sur la machine virtuelle à supprimer sélectionnez “Delete VM from Disk”.
- Appliquez les mises à jour proposées automatiquement par le gestionnaire “Update manager”
- Redémarrer ou fermer Ubuntu à l’aide de l’icône tout en haut à droite “Shut Down” ou “Eteindre”

Félicitations, vous disposez d’une plateforme Linux

Résumé

Le principal objectif de cette étude est la détection d'intrusion pour sécuriser un réseau local en utilisant un IDS réseau qui est le SNORT.

Dans le premier chapitre, nous avons décrit les notions théoriques de base sur les réseaux informatiques en général. Pour ce faire, dans un premier temps, nous présentons les types de réseau informatique, ensuite nous donnons un aperçu sur les différentes couches du modèle OSI. En fin, nous citons les protocoles de communication amenés à faire le routage des données entre les réseaux, ainsi les équipements utilisés.

Dans le deuxième chapitre, nous avons donné en premier lieu la description des différents types d'attaques et menaces pour s'introduire dans un système. En second lieu, nous avons exposé les différentes contre-mesures et les protocoles de sécurité du réseau.

Dans le troisième chapitre, nous avons étudié les systèmes de détection d'intrusion et les systèmes de prévention d'intrusion, ainsi leurs types, leur fonctionnement et nous allons présenter l'architecture des IDS.

Dans le dernier chapitre, nous avons présenté le logiciel open source SNORT sous linux en décrivant toutes les étapes d'installation ainsi sa configuration. Ensuite, nous allons effectuer un test d'intrusion pour s'assurer de la bonne fonctionnalité de ce logiciel.

Les mots clés

SNORT-RESEAU-DETECTION-INTRUSION-ATTAQUE-LOGICIEL-IDS-IPS-VPN -

DNS-DOS-NIDS-NIPS-HIDS-HIPS-HTML-HTTP-IDS-INTERNET-IP-IPS-ISO-LAN-

MYSQL-TCP-TELNET-FTP-UDP-ARP-ICMP-PROTOCOLS-SÉCURITÉ-

INSTALLATION-CONFIGURATION-BASE