الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne démocratique et populaire

وزارة التعليم العالي و البحث العلمي
Ministère de l'enseignement supérieur et de la recherche scientifique

جامعة سعد دحلب البليدة
Université SAAD DAHLAB de BLIDA

كلية التكنولوجيا
Faculté de Technologie

قسم الإلكترونيك
Département d'Électronique

Mémoire de Master

Mention Électronique
Spécialité Réseaux & Télécommunications

présenté par

AMIMER Sofiane

&

HADIDI Walid

Interception des attaques Réseaux avec L'IDSSnort

Proposé par : Dr. MEHDI Merouane

Année Universitaire 2016-2017

Remerciements

Nos remerciements en premier lieu à Dieu le toutpuissant pour la volonté, la santé, le courage et la patience qui nous a donnépour mener ce travail à terme.

Nous tenons à remercier tout particulièrement nos parents qui ont toujours étélà pour nous.

Nos remerciements à Mr. *MEHDI Merouane*, qui n'a épargné aucun effortpour le bon déroulement de ce travail. Ses remarques et consignes ont étépour nous d'un grand apport.

Nos remerciements à l'ensemble des membres du jury qui ont accepté d'évaluer notre modeste travail.

Nos remerciements aussi pour les techniciens de centre des systèmes et réseaux d'information etde communication.

Enfin, nous exprimons notre gratitude à tous ceux et celles qui sont directement ouIndirectement impliqués dans la réussite de ce travail.

Dédicace

Je dédie ce travail

A

"Ma Famille"

« Mon père & ma mère pour leur soutien "

« Mon oncle Mehdi "

"Tout ceux qui m'ont aidé afin de réaliser ce travail"

Sofiane

ملخص:

اليوم، أصبحت الهجمات على أنظمة الكمبيوتر عديدة، قوية، ذكية وتسبب أضرارا كبيرة. والهدف من هذه مذكرة التخرج هو تقديم مفهوم نظام كشف التسلل بشكل عام وعلى وجه الخصوص سنورط، وتحليل هذه الهجمات بويريشاك ولاستخراج توقيعاتهم، واقتراح قواعد الكشف الخاصة بنا من أجل للكشف عن الهجمات المختلفة، استنادا إلى توقيعات الهجمات التي تم الحصول عليها من مختلف الاختبارات التي أجريت.

كلمات المفاتيح: الهجمات. تحليل الحزم؛ ويريشاك؛ نظام كشف التسلل؛ سنورط؛ القواعد.

Résumé :

De nos jours, les attaques envers les systèmes informatiques sont devenues nombreuses, plus spécifiques, puissantes, intelligentes et causent des dégâts considérables.

L'objectif de ce projet est de présenter le concept d'IDS (Intrusion Detection System) en générale et en particulier Snort, et d'analyser les paquets des attaques avec Wireshark pour extraire leurs signatures, ainsi de proposer nos propre règles de détection afin de pouvoir détecter les différents attaques (DOS/Ping of death, HTTP,TCP, UDP) à base des signatures d'attaques obtenuesdes différents tests effectués.

Mots clés :attaques ;analyser les paquets ; Wireshark ; IDS ; Snort ; règles.

Abstract :

Today, attacks on computer systems have become numerous, more specific, powerful, intelligent and causing considerable damage.

The aim of this thesis is to present the concept of IDS (Intrusion Detection System) in general and in particular Snort. And to analyze the packets of attacks with Wireshark to extract their signatures, and to propose our own detection rules in order to detect the different attacks (DOS / Ping of death, HTTP, TCP, UDP) based on the signatures of attacks obtained from the various tests carried out.

Keywords : Attacks ; analyze packets ; Wireshark ; IDS ; Snort ; rules.

Listes des acronymes et abréviations

IDS	Intrusion Detection System
LAN	Local Area Network
MAN	Metropolitan Area Network
WAN	Wide Area Network
FDDI	Fiber Distributed Data Interface
IP	Internet Protocol
ICMP	Internet Control Message Protocol
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
HTTP	Hypertext Transfer Protocol
LOIC	Low Orbit Ion Cannon
NIDS	Network Intrusion Detection system
HIDS	Host Intrusion Detection System
DDOS	Distrubuted Denial of Service
Flooding	Inondation
AUI	Attachement Unit Interface
T.BNC	Bayonet Neill-Concelman Connector
10Base5	Norme ethernet, longueur max 500 m, vitesse de 10Mb/s

N.BNCAdaptateur RF Fiche Type N vers Prise BNC, Droit, 50Ω

HUBLe premier type de concentrateur(répétiteur)

OSIOpen Systems Interconnection

ISOOrganisation internationale de normalisation

ACK	Acknowledgement

PSHPUSH, les données sont immédiatement envoyées

SYNSynchronisation, demande d'une connexion

Spoof	Usurpation
HTTP GET	Demande d'une ressource

Table des matières

Liste des figures

Chapitre 1

Chapitre 2

Chapitre 3

Chapitre 4

Liste des tableaux

Chapitre 1

Chapitre 2

Chapitre 3

Introduction générale

La croissance de l'Internet et l'ouverture des systèmes ont fait que les attaques dans les réseaux informatiques soient de plus en plus nombreuses. Les vulnérabilités en matière de sécurité s'intensifient, d'une part au niveau de la conception des protocoles de communication ainsi qu'au niveau de leur implantation et d'autre part, les connaissances, les outils et les scripts pour lancer les attaques sont facilement disponibles et exploitables. D'où la nécessité d'un système de détection d'intrusions.

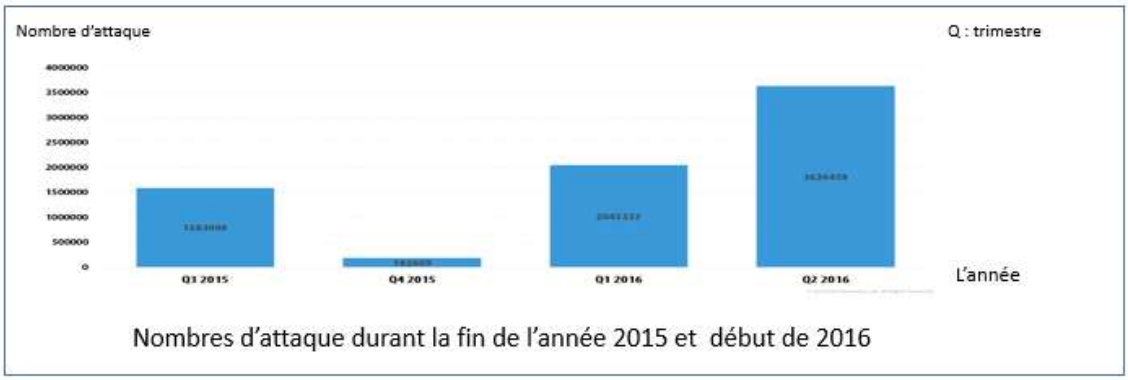

Figure : Nombres d'attaque (2015/2016) [10]

Cette technologie consiste à rechercher une suite de mots ou de paramètres caractérisant une attaque dans un flux de paquets. Les systèmes de détection d'intrusion sont devenus un composant essentiel et critique dans une architecture de sécurité informatique.

Notre travail consiste à étudier et concevoir des règles de détection basée sur un détecteur des attaques open source (Snort), à savoir l'attaque « Ping of death, TCP, UDP, HTTP » basé sur une simulation de l'outil d'attaque LOIC, et l'utilisation de

Wireshark pour capturer les paquets et les comparer avec les cas normaux pour extraire la signature de chaque attaque a fin de crée nos règles de détection.

Le premier chapitre de ce mémoire présente les concepts de bases des réseaux informatiques. Il présente aussi la classification des réseaux et leur différente topologie, les supports de transmission et le modèle OSI (open system interconnexion).

Le second chapitre traite les concepts de bases de la sécurité des réseaux informatiques. Il présente aussi les types et les méthodes d'attaques informatiques, ainsi que différents moyens permettant de contrer ces attaques.

Le troisième chapitre concerner la visualisation avec un analyseur de trafic Wireshark des paquets au niveau des Protocole utilisées dans les attaques, ainsi que la comparions des paquets normale et celles des attaques afin représenter les signatures d'attaques.

Pour finir, dans le dernier chapitre, on effectuera des tests de détection après avoir créé nos propres règles de détection à base des signaturesacquis au chapitre 3. Et ainsi prouvé les résultats des détections des attaques.

Chapitre 1 Généralité sur les réseaux

1.1 Introduction

Un réseau informatique est un élément important dans le domaine de lacommunicationqui permet aux entreprises de centraliser leurs données, de travailler en équipede manière productive et limiter les impressions papiers pour le transfert d'informations. Les réseaux informatiquessont tous les équipements qui permettent l'échanged'informations au sein d'une entreprise et le partage de ressources donc une gestionde l'accès internet, les mails, les droits d'accès aux documents partagés .ainsi que la miseà disposition d'une plateforme de travail collaboratif.

1.2 Généralités sur les réseaux informatiques

1.2.1 Définition d'un réseau informatique

Réseau (informatique) : ensemble d'ordinateurs et de terminaux interconnectés pour échanger des informations numériques.

1.3 Classification des réseaux

On peut distinguer différent types de réseaux [1] :

1.3.1 Réseau local (Local Area Network)

Réseau local : réseau d'une entreprise, il peut se développer sur plusieurs bâtiments et permet de satisfaire tous les besoins internes de cette entreprise (1m à quelques kms).

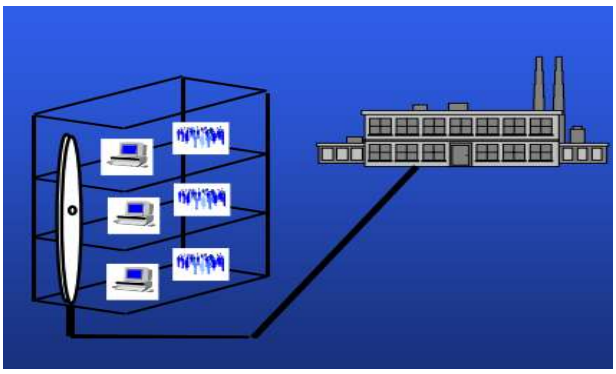

Figure 1.1 : réseau local [2].

1.3.2 Réseau métropolitain (Metropolitan Area Network)

Réseau métropolitain : relie différents sites d'une Université ou d'une administration, chacun possédant son propre réseau local (quelques kms à quelques dizaine de kms). Ainsi un MAN permet à deux nœuds distants de communiquer comme si ils faisaient partie d'un même réseau local. Un MAN est formé de commutateurs ou de routeurs interconnectés par des liens hauts débits (en général en fibre optique).

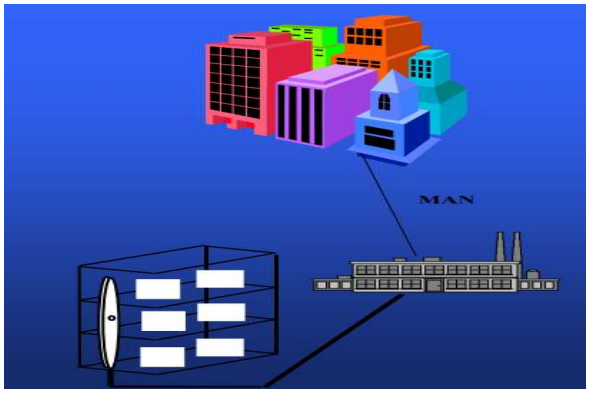

Figure 1.2 : réseau métropolitain [2].

1.3.3 Réseau étendu (Wide Area Network)

Réseau étendu : 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 (plusieurs centaines de km). Les WAN fonctionnent

grâce à des routeurs qui permettent de "choisir" le trajet le plus approprié pour atteindre un nœud du réseau. Le plus connu des WAN est Internet.

1.4 Topologie des réseaux

1.4.1 Topologie physique

L'arrangement physique des éléments constitutifs d'un réseau est appelé topologie physique. Il en existe trois [2] :

a Topologie en bus

Chaque machine est reliée à un câble commun, cette topologie est économique en câblage et permet facilement l'extension de réseau par ajout d'équipement, l'un de ces inconvénients la défaillance de réseau en cas de panne du support.

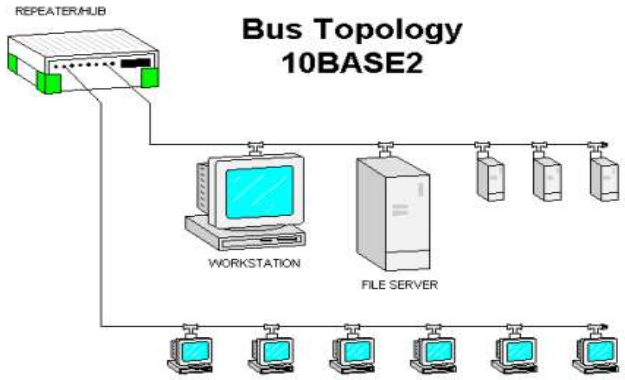

Figure 1.3 : Topologie en bus [3].

b Topologie en Etoile

Chaque machine est reliée directement à un serveur, les données transitent toutes à travers le nœud central. L'ajout d'une station ne nécessite pas la coupure du réseau, en revanche elle peut entrainer des longueurs importantes de câbles.

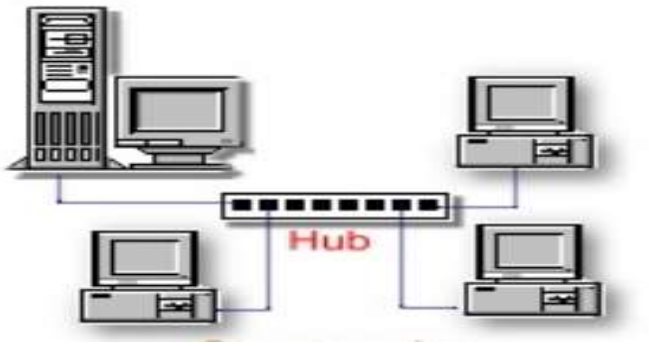

Figure 1.4 : Topologie en étoile [3].

c Topologie en anneau

Chaque machine est reliée à deux équipements voisins, on obtient ainsi une boucle fermée, les données transitent de station en station jusqu'à destination.

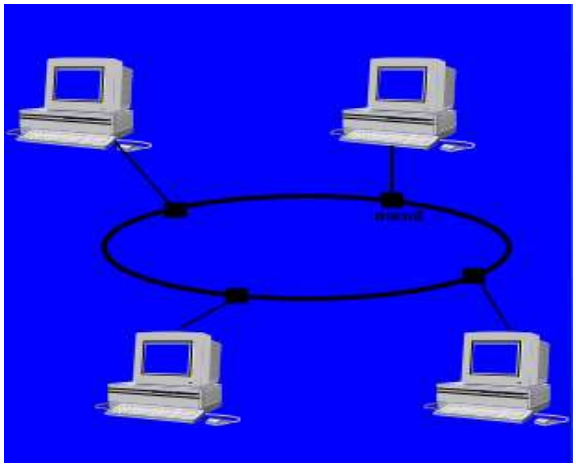

Figure 1.5 : Topologie en anneau [2].

1.4.2 Topologie logique

La topologie physique (câblage et organisation dimensionnelle) se distingue de latopologie logique. La topologie logique représente la façon avec laquelle les donnéestransitent dans les supports. Les topologies logiques les plus courantes sont ethernet,token Ring et FDDI.

a Ethernet

Est un protocole de réseau local à commutation de paquets ; c'est une norme internationale. Depuis les années 1990, la topologie utilisant des câbles coaxiaux est toujours de type bus. Cette topologie était avantageuse lorsque le nombre et la disposition des stations changeaient. Aujourd'hui, les câbles coaxiaux sont systématiquement abandonnés au profit des câbles en paires torsadées cuivre ou desfibres optiques. Le coût de la connectique des câbles coaxiaux est devenu supérieurà celui de la connectique RJ45 utilisée avec les paires torsadées [4].

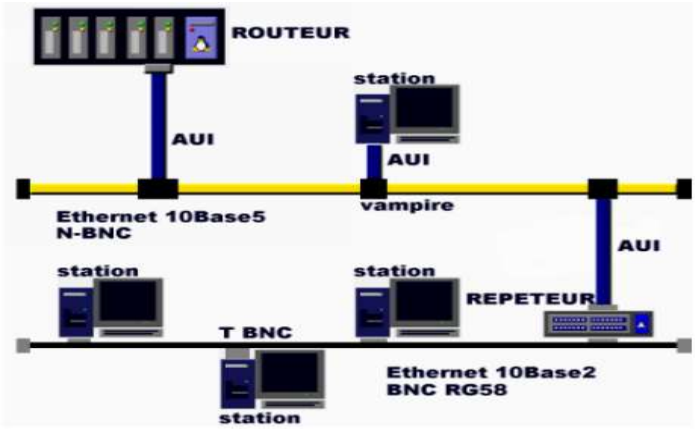

Figure 1.6 : Topologie ethernet bus [4]

b Token Ring

Elle fonctionne sur un réseau en anneau. Sur un tel réseau, un ordinateur doit capturer une trame spéciale, appelée jeton, pour pouvoir envoyer des données. Cette méthode évite les collisions [5].

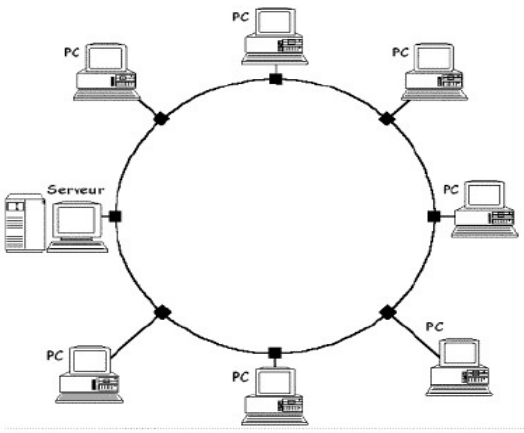

Figure 1.7 : Topologie token ring[5].

c FDDI

FDDI (fiber distributed data interface) est un type de réseau token ring. FDDI est souvent utilisé pour des réseaux locauxou métropolitains, comme ceux qui connectent plusieurs batiments dans un complexede bureaux ou dans un compus.

Comme son nom l'indique, FDDI fonctionne au moyen d'un cable à fibre optique. FDDIcombine une performance à haute vitesse aux avantages de la topologie en anneau àpassage de jeton.

1.5 Les supports de transmission

Pour de nombreuses applications, il est nécessaire de disposer d'une liaison directe et permanente entredeux ordinateurs éloignés. Le plus ancien support de transmission employé à cette fin, et encore le pluslargement utilisé aujourd'hui, est la paire torsadée. Les câbles coaxiaux et les fibres optiques sontégalement fréquents. Enfin, les communications sans fils sont en plein essor [6].

1.5.1 La paire torsadée

C'est un câble téléphonique constitué à l'origine de deux fils de cuivre isolés et enroulés l'un sur l'autre. Actuellement on utilise plutôt des câbles constitués de 2 ou 4 paires torsadées. Chaque extrémité d'un tel câble étant munie d'une prise RJ45.

Figure 1.8 : Paire torsadée [7].

1.5.2 Les câbles coaxiaux

Il est constitué d'un cœur qui est un fil de cuivre. Ce cœur est dans une gaine isolante elle-même entourée par une tresse de cuivre, le tout est recouvert d'une gaine isolante.

Figure1.9 : Câble coaxiale [7].

1.5.3 La fibre optique

C'est un support d'apparition plus récente. Elle permet des débits de plusieurs Gbit/s sur de très longues distances. En plus de ses capacités de transmission, son avantage : immunité aux interférences électromagnétiques.

Figure 1.10 : Fibre optique [7].

1.6 Le modèle OSI

1.6.1 Définition de modèle OSI

Dans les années 70, l'organisme ISO (International Standard Organisation) a développé un modèle dans le but de répondre à l'ensemble de ces questions indépendamment les unes des autres, par conséquence, d'interconnecter des réseaux selon une norme OSI (Open System Interconnexion).

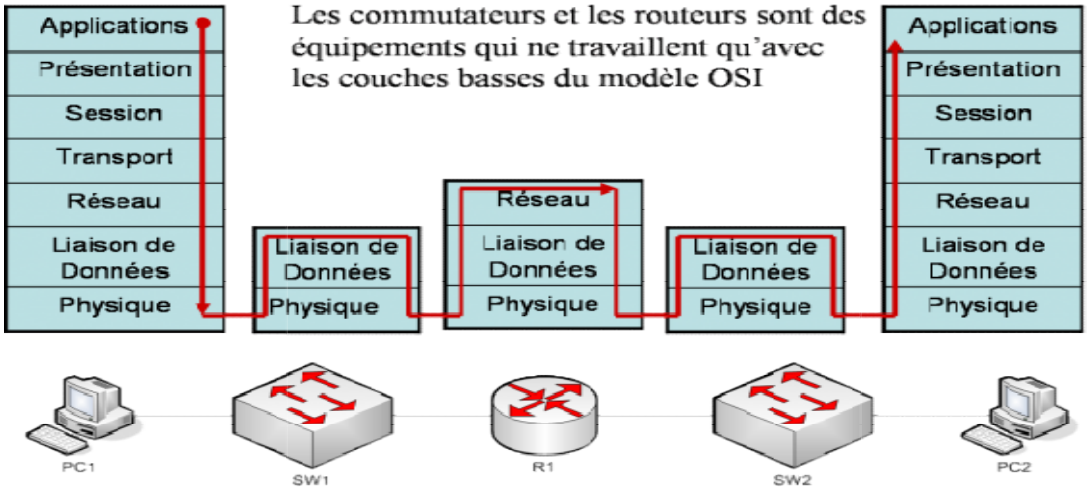

Figure 1.11 : Le modèle OSI [8].

Chaque couche ou interface (de niveau n) fournit un certain nombre de fonctionnalités mises à la disposition de la couche immédiatement supérieure (de niveau n + 1) et permet de communiquer avec la couche de même niveau (n) d'un autre dispositif selon un protocole qui spécifie la séquence des actions possibles [1].

a La couche physique

Cette couche se charge de la transmission et la réception des données informatiques au format binaire (0 et 1) ou sous forme de trame.

Structure de la trame :

Toutes les informations sont transportées dans une structure unique : la trame est de longueur variable, elle est délimitée par une séquence binaire spécifique appelée flag. En cas d'émission consécutive de trames, le flag marque la fin d'une trame et le début de la suivante. Les différents champs sont :

Le champ adresse s'étend sur un octet et identifie une des extrémités de la liaison.

Le champ Commande décrit le type de la trame il s'étend sur 1 octet mais peut être porté à 2 octets dans le mode appelé mode étendu.

Le champ donné est un champ facultatif contenant un nombre quelconque d'éléments binaires représentant les données de l'utilisateur.

Le champ FCS (Frame Check Sequence) est une séquence de contrôle de trame.

Le champ de gauche est le premier transmis, le champ de droite est le dernier.

Fanion	Adresse	Commande	Données	FCS	Fanion
01111110	(8 bits)	(8 bits)	(n ≥ 0 bits)	(16 bits)	01111110

Figure 1.12 : Structure de la trame

b Couche liaison de données

Cette couche définit comment la transmission des données est effectuée entre 2 machines adjacentes. Par exemple, un pc connecté à un switch, deux routeurs connectés entre eux.

c Couche de réseau

La couche réseau assure toutes les fonctionnalités de relai et d'amélioration de services entre entité de réseau : l'adressage et le routage, le contrôle de flux, la détection et la correction d'erreur non réglées par la couche 2 [8].

Protocol IP :

Chaque équipement sur le réseau est repéré par une adresse, appelée IP (internet Protocol V4), codée sur 32 bits avec deux champs principaux précisant une identité de réseau et une identité de machine. Plusieurs classes d'adresses sont définies suivant la longueur des champs d'identité de réseau.

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 1.1 : Les différentes classes des réseaux

Le protocole IP assure un service non fiable sans connexion de remise des données. Il comprend la définition du plan d'adressage, la structure des informations transférées (le *datagramme IP)* et les règles de routage. L'envoi de messages d'erreur est prévu en cas de destruction de datagrammes, de problèmes d'acheminement ou de remise [1].

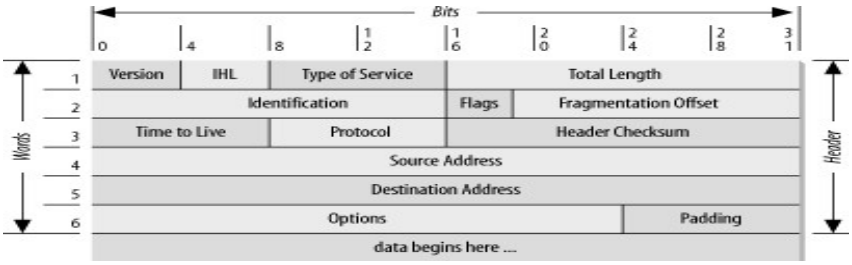

Figure 1.13 : Datagramme IP [1]

	Descriptions
IHL *(Intenet Header Length)*	indique la longueur de l'en-tête en mots de 32 bits. La longueur totale est la longueur du datagramme en octets, en-tête compris. L'identification est un numéro permettant d'identifier de manière unique les fragments de même datagramme.
DF (*Don 't Fragment)*	*Don 't Fragment*, interdit la fragmentation du datagramme (toute machine doit accepter les fragments de 476 octets ou moins).
MF (*More Fragments)*	est mise à 1 pour tous les fragments d'un même datagramme initial sauf pour le dernier fragment. Le numéro de fragment permet de reconstituer, dans l'ordre, le datagramme initial à partir de l'ensemble des fragments.
TTL (Time *To Live)*	*Time To Live*, indique le nombre de secondes qui restent à vivre au datagramme. Ce champ est modifié par les routeurs IP au cours de la traversée du réseau par le datagramme.

Tableau 1.2 : Descriptions de datagramme IP

Protocol ICMP :

Internet est un réseau décentralisé. Il n'y a pas de superviseur global du réseau. Chaque routeur fonctionne de manière autonome. Des anomalies, dues à des pannes d'équipement ou à une surcharge temporaire, peuvent intervenir. Afin de réagir correctement à ces défaillances, le protocole de diagnostic ICMP, internet control message protocol, a été développé. Chaque équipement surveille son environnement

et échange des messages de contrôle lorsque c'est nécessaire. Ces messages sont transportés par IP dans la partie donnée des datagrammes [8].

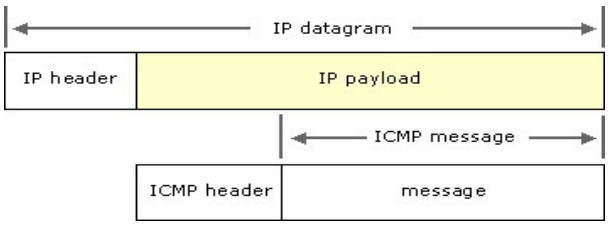

Figure 1.14 : Protocol ICMP

Ping :

Ping est un outil de test de la connectivité réseau TCP/IP.

La commande de Ping sur la commande de base : Ping adresse IP destination [15].

d Couche de transport

La couche transport assure un transfert de données transparents entre entités en les déchargeant des détails d'exécution. Elle a pour rôle d'optimiser l'utilisation des services réseaux disponibles afin d'assurer au moindre coût les performances requises par la couche session.

Port informatique :

Un port logiciel est un système permettant aux ordinateurs de recevoir ou d'émettre des informations, il existe 65536 port, chaque service occupe un seul port informatique exemple port de HTTP c'est 80.

ProtocoleTCP

Le protocole TCP (transmission control Protocol) est implanté au-dessus du protocole IP pour assurer un transfert fiable en mode connecté : il fournit le même service que le protocole de transport, dit de classe 4, défini dans le modèle OSI. Il est capable de détecter les datagrammes perdus ou dupliqués, et de les remettre dans l'ordre dans lequel ils ont été émis. Il repose sur le principe de numérotation et d'acquittement des données [1].

Figure 1.15 : Protocol TCP

Signification des différents champs :

	Descriptions
Port Source (16 bits)	port relatif à l'application en cours sur la machine source
Port Destination (16 bits)	port relatif à l'application en cours sur la machine de destination.
Numéro d'accusé de réception (32 bits)	numéro d'ordre du dernier octet reçu par le récepteur.
URG	si ce drapeau est à 1 le paquet doit être traité de façon urgente.
ACK	si ce drapeau est à 1 le paquet est un accusé de réception.
PSH (PUSH)	si ce drapeau est à 1, le paquet fonctionne suivant la méthode PUSH.
RST	si ce drapeau est à 1, la connexion est réinitialisée.
SYN	si ce drapeau est à 1, les numéros d'ordre sont synchronisés (ouverture de connexion).
FIN	si ce drapeau est à 1 la connexion s'interrompt.

Tableau 1.3 : Descriptions de protocoleTCP

Selon le protocole de communication TCP, une connexion entre deux hôtes s'établit en trois étapes suivantes,Three-way handshake [1] :

- SYN : Le client qui désire établir une connexion avec un serveur va envoyer un premier paquet SYN (synchronized) au serveur. Le numéro de séquence de ce paquet est un nombre aléatoire A.
- SYN-ACK : Le serveur va répondre au client à l'aide d'un paquet SYN-ACK (synchronize, acknowledge). Le numéro de l'ACK est égal au numéro de séquence du paquet précédent (SYN) incrémenté de un (A + 1) tandis que le numéro de séquence du paquet SYN-ACK est un nombre aléatoire B.
- ACK : Pour terminer, le client va envoyer un paquet ACK au serveur qui va servir d'accusé de réception. Le numéro de séquence de ce paquet est défini selon la valeur de l'acquittement reçu précédemment (par exemple : A + 1) et le

numéro du ACK est égal au numéro de séquence du paquet précédent (SYN-ACK) incrémenté de un (B + 1).

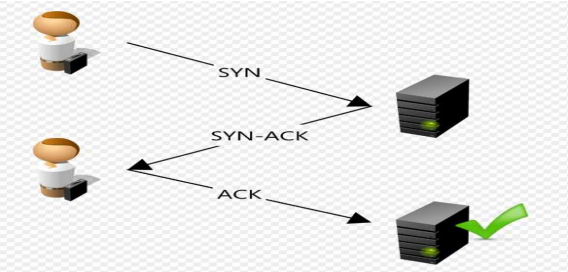

Figure 1.16 : Three-way handshake [1]

Protocol UDP :

Est un protocol orienté « non connexion » L'en-tête du paquet UDP(user datagram protocol) est très simple :

Port Source (16 bits)	Port Destination (16 bits)
Longueur (16 bits)	Somme de contrôle (16 bits)
Données (Longueur variable)	

Figure 1.17 : ProtocoleUDP

e La couche session

La principale fonction de la couche session est de fournir aux utilisateurs (entité de la couche de présentation ou processus de la couche application) les moyens d'établir des connexions appelées sessions et d'y transférer des données en bon ordre [1].

f La couche présentation

Cette couche formate les données pour qu'elles sont compréhensibles par l'application qui les demandées.

g La couche application

Cette couche fait l'interface entre l'homme et la machine, La couche application est constituée par l'ensemble des programmes courants à disposition de l'utilisateur.

1.7 Conclusion

Dans ce chapitre, on a défini les réseaux informatiques et leur classification, puis nous avons présentés la structure d'un réseau et ces déférentes topologies, A la fin on a cité les supports de communication et ces caractéristiques, et le modèle OSI de l'ISO.

Nous avons vu dans ce chapitre l'utilité des réseaux informatiques et le besoin qui a abouti à la création des réseaux (le partage des ressources, le stockage des fichiers etc...).

De nos jours les systèmes réseaux informatiques sont devenus vaste, cette dernière ont fait que les attaques dans les réseaux informatiques soient de plus en plus nombreuses ce qui nécessite une bonne politique de sécurités qu'on va décrire dans le chapitre 2.

Chapitre 2 Sécurité informatique et réseaux

2.1 Introduction

La sécurité informatique est de nos jours devenue un problème majeur dans la gestion des réseaux d'entreprise ainsi que pour les particuliers toujours plus nombreux à se connecter à Internet. La transmission d'informations sensibles et le désir d'assurer la confidentialité de celles-ci est devenue un point primordial dans la mise en place de

réseaux informatiques. Ce chapitre a pour but de présenter globalement la manière dont les pirates (Hackers) opèrent afin de pénétrer les systèmes informatiques en espérant qu'il aide à pallier ce type de problème de plus en plus fréquent.

2.2 Définition de la sécurité informatique

La sécurité informatique c'est l'ensemble des moyens mis en œuvre pour minimiser la vulnérabilité d'un système contre des menaces accidentelles ou intentionnelles.

La sécurité informatique, d'une manière générale consiste à assurer une utilisation des ressources matérielles ou logicielles d'une organisation.

La sécurité informatique consiste généralement en quatre principaux objectifs :

L'intégrité : c'est-à-dire garantir que les données sont bien celles qu'on croit être.

La confidentialité :consistant à assure que seuls les personnes autorisées aient accès aux ressources.

La disponibilité : permettant de maintenir le bon fonctionnement du système informatique.

La non répudiation : permettant de garantir d'une transaction ne peut être niée [9].

2.3 Les attaques informatiques

2.3.1 Définition de l'attaque informatique

Une attaque est l'exploitation d'une faille d'un système informatique (système d'exploitation, logiciel ou bien même de l'utilisateur) à des fins non connues par l'exploitant du système et généralement préjudiciables [10].

2.3.2 Les pirates informatiques

Faut-il dire plutôt pirate ou hacker ? On s'est tous posé cette question. Les journalistes et le grand public confondent souvent les termes. Les pirates désignent des

spécialistes en informatiques dont les actions sont nuisibles. Selon leurs actions ils peuvent être qualifiés de hackers blacks hats, de crackers ou encore d'hacktivistes [11].

2.4 Les types d'attaques

On entend souvent aux informations qu'un nouveau virus circule. Mais ce n'est pas la seule menace pour nos ordinateurs. Il existe pleins de programmes malveillants, les paragraphes suivants détaillent quelques-unes des principales menaces :

2.4.1 Les malwares

Un malware est un logiciel développé dans le but de nuire à un système informatiques il existe plusieurs familles de malwares. On va définir les plus utilisés :

a Les virus

Les virus sont des programmes malveillants qui ont pour but de se reproduire. Souvent, ils sont gênants pour l'utilisateur, puisqu'ils peuvent détruire des fichiers sur l'ordinateur [12].

b Les vers

Les vers sont des programmes qui se propagent d'ordinateur à ordinateur via un réseau comme l'Internet. Ainsi, contrairement à un virus, les vers n'ont pas besoin d'un programme hôte pour assurer leur reproduction. Leurs poids est très léger, ce qui leur permet de se propager à une vitesse impressionnante sur un réseau, et pouvant donc saturer ce dernier et Espionner l'ordinateur ou il se trouve, offrir un port dérobée a les pirates informatique [12].

c Les spywares

Les spywares, ou logiciels espions, sont des logiciels nuisibles qui transmettent à des tiers des informations contenues dans votre ordinateur. Les spywares sont souvent présents dans des gratuiciels (différents des logiciels libres), ou des partagiciels. En général les logiciels à code source libre comme Mozilla Firefox n'en contiennent aucun [12].

d Le spamming

Le spamming (ou encore pourriel, courrier rebut) consiste à envoyer des messages appelés "spam" à une ou plusieurs personnes. Ces spams sont souvent d'ordre publicitaire. Tous les points suivants sont considérés comme du spamming.

· Envoyer un même mail, une ou plusieurs fois à une ou plusieurs personnes en faisant de la publicité.

· Poster un ou plusieurs messages dans un forum qui n'a rien à voir avec le thème.

· Faire apparaître un message publicitaire lorsque l'on navigue sur un site [12].

e Cheval de Troie

C'est un programme ou un code malveillant intégré à une application par ajout ou par modification de son code. Lors de l'exécution de ce programme. Le bout de code malveillant pourra exécuter des commandes spécifiques (récupération de fichiers de mot de passe, etc.) à l'insu de l'utilisateur, reposant sur une porte dérobé « backdoor » [12].

2.4.2 Les attaques

Il existe différents types d'attaques parmi lesquels nous pouvons noter ceux relatifs :

a Les Denial-of-Service (Dos)

Les attaques de type Denial-of-Service ont pour but de saturer un routeur ou un serveur afin de le crasher. Ces types d'attaque sont très faciles à mettre en place et très difficile à empêcher mais quelles sont les raisons qui peuvent pousser un attaquant à utiliser ce genre d'attaque sachant que cela peut mener à la ”destruction” du routeur ou du serveur visé :

Récupérer un accès : une attaque de type Denial-of-Service fait, la plupart du temps, partied'une attaque visant à obtenir le contrôle d'une machine ou d'un réseau. Par exemplel'attaque de type ”SYN Flood”, très repandue, est souvent utilisée avec une tentative de ”Spoofing” [12].

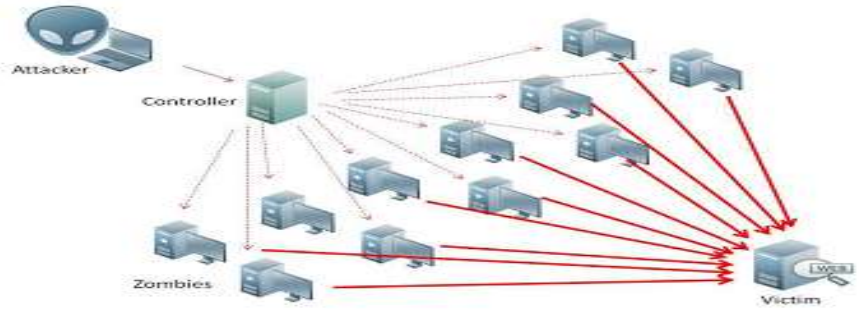

Figure 2.1 : Attaque DDOS [12]

Voici quelques exemples de programmes disponibles sur Internet permettant deréaliser ceGenre d'attaque [12] :

• Ping Of Death : il s'agit de saturer un routeur ou un serveur en envoyant un nombreimportant de requêtes "ICMP REQUEST" dont les datagrammes dépassent la taille maximumautorisée.

• Land - Blat : cette attaque permet de geler la plupart des systèmes ayant plus ou moinsun an. Il sera alors obligatoire de redémarrer la machine afin d'en reprendre le contrôle. IlS'agit d'envoyer un paquet forgé (spoof) contenant le flag SYN sur un port donné (comme113 ou 139 par exemple) et de définir la source comme étant l'adresse de la station cible. Ilexiste un certain nombre de patches pour ce "bug" pour les systèmes UNIX et Windows.

• Smurf : ce programme utilise la technique de l'"ICMP Flood" et l'amplifie de manière àcréer un véritable désastre sur la (ou les) machines visées. En fait, il utilise la techniquedu "Broadcast Ping" afin que le nombre de paquets ICMP envoyés à la station grandissede manière exponentielle causant alors un crash presque inévitable. Il est difficile de seprotéger de ce type d'attaques, il n'existe aucun patch mais des règles de filtrage correctespermettent de limiter son effet.

b L'IP Spoofing

La technique de l'IP Spoofing est une technique dont le principe est relativement ancien (auxalentours de 1985) mais la première attaque connue l'utilisant ne remonte

qu'à 1995. KevinMitnick, un célèbre "Hacker", l'a utilisé afin de s'infiltrer dans le réseau d'un expert en sécuritéinformatique, Tsutomo Shimomura. Le Spoofing n'est pas l'attaque en tant que tel, il s'agitd'une technique permettant de s'infiltrer dans un ordinateur en se faisant passer pour un autreen qui il a confiance (Trusted Host).

Voici un bref résumé du fonctionnement decette technique :

Une station se fait passer pour une autre en envoyant un paquet dont l'adresseIP est autorisée par le serveur visé. . . La source IP envoyée trompe donc la cible qui accordel'accès en pensant avoir affaire à une machine de confiance.

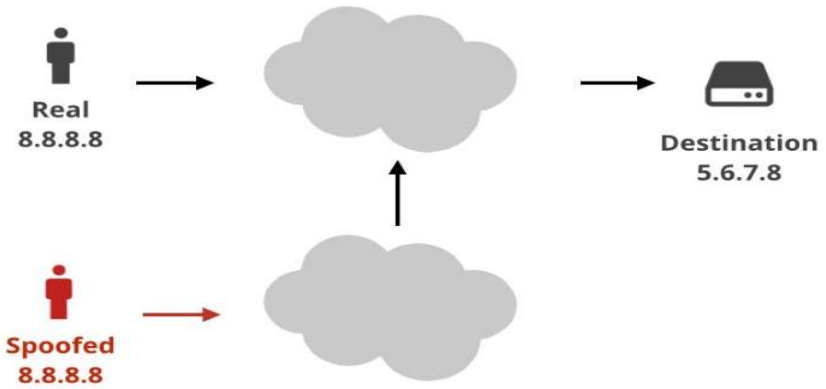

Figure 2.2 : Attaque IP spoofing

c Les Backdoors

Depuis que les intrusions informatiques existent, leurs adeptes ont mis au point un certainnombre de techniques leur facilitant l'accès aux systèmes pénétrés. La technique la plus connue,et sans doute la plus utilisée, est celle des Backdoors (portes dérobées ou portes de service). Ellespermettent, à celui qui en connait l'existence et le fonctionnement, de revenir sur un systèmede façon détournée, c'est-à-dire sans passer par les méthodes d'authentification habituelles [12].

En règle générale, les Backdoors permettent différents types d'actions sur le système où ellessont installées :

• se reconnecter sur la machine même après un changement de mots de passe ou d'ajoutsde systèmes de sécurité.

• rendre invisible les connections et les actions réalisées.

• déranger le travail des utilisateurs par l'envoi de messages, la modification de fichiers,l'affichage d'images, la lecture de fichiers.

• exécuter certaines commandes bien ciblées permettant d'avoir une vision de l'état de la station (processus, connections réseau) ou de modifier le contenu de certainsfichiers de configuration (mots de passe, réseau,. . .).

2.4.3 Outils d'attaques

Les outils les plus utilisés pour les différent attaques ou testes d'intrusion:

a LOIC (Low Orbit Ion Cannon) :

Est l'un des outils d'attaque DOS les plus puissants disponibles gratuitement. Il est devenu largement utilisé, Cette application tente d'attaquer par déni de service le site ciblé en inondant le serveur avec des paquets TCP, des paquets UDP, des requêtes HTTP avec l'intention de perturber le service d'un hôte particulier. Y compris dans certaines attaques très médiatisées contre les serveurs PayPal, Mastercard et Visa depuis quelques mois. Cet outil a également été l'arme de choix mise en œuvre par le célèbre groupe de pirates informatiques Anonymous, qui a revendiqué de nombreuses attaques de piratage de haut niveau, parmi lesquelles des hacks contre Sony, le FBI et d'autres agences de sécurité américaines. Le groupe a non seulement utilisé cet outil, mais a également demandé que d'autres le téléchargent et rejoignent des attaques anonymes via IRC [16].

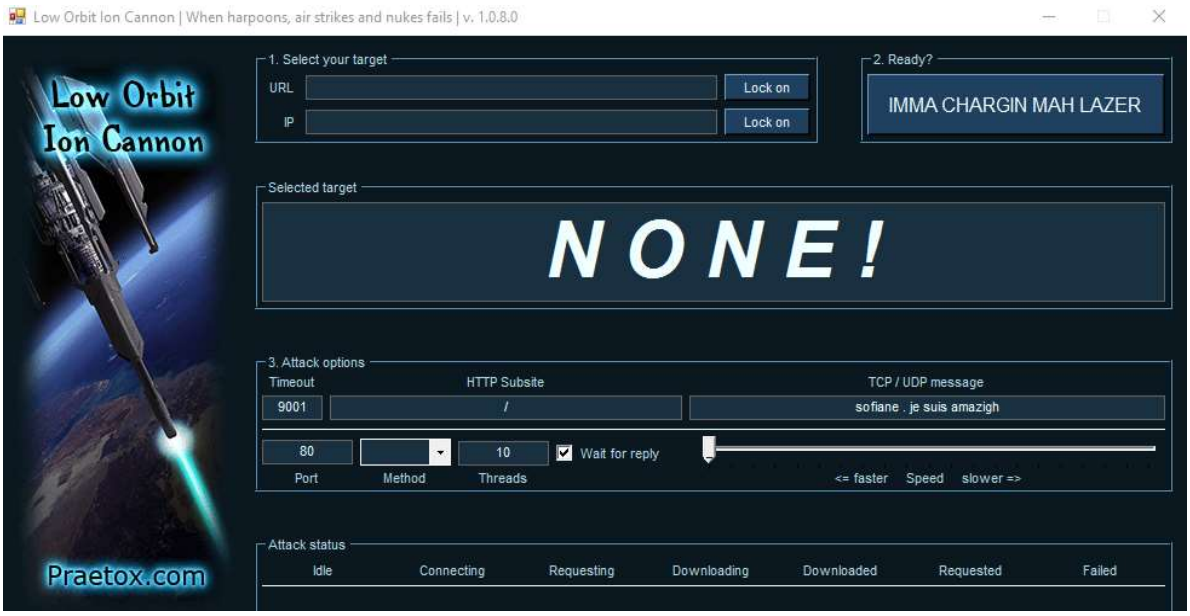

Figure 2.3 : Capture del'interface deLOIC

Voici la signification de chaque champ :

IDLE : il montre le nombre de threads inactifs. Il devrait être nul pour une plus grande efficacité de l'attaque.

Connexion : ceci indique le nombre de threads qui tentent de se connecter au serveur victime.

Demander : Cela montre le nombre de threads qui demandent des informations sur le serveur de la victime.

Téléchargement : ceci montre le nombre de threads qui lancent un téléchargement pour des informations du serveur.

Téléchargé : ce nombre indique combien de fois le téléchargement de données a été lancé à partir du serveur victime sur lequel vous attaquez.

b Nmap

Est un scanner de réseau. Il permet de savoir quels sont les ports ouverts, fermés ou filtrés, ainsi que le système d'exploitation autorisé et sa version.

Il permet par exemple de scanner un ensemble d'adresses IP en précisant la méthode de scan utilisée, les types de ports tels que les ports UDP, en tentant d'identifier la machine cible et en sauvegardant le résultat dans un fichier [12].

Voici des autres outils d'attaque :

Logiciel	Type d'attaques
Trinoo	UDP flooding
Tribe Flood Network (TFN) et TFN2k	UDP/TCP/TCP SYN flooding, Smurf
Stacheldraht	UDP/TCP/TCP SYN flooding, Smurf
Schaft	UDP/TCP/ICMP flooding
MStreamT	ACK flooding

Tableau 2.1 : Outils d'attaques

2.5 Protection des réseaux informatiques

Fort heureusement, il existe des logiciels permettant de mettre en place une politique de sécurité et ainsi éviter certaines attaques. Tout le monde aentendu parler du Firewall (pare-feu en français), ou encore de l'antivirus et le système de détection d'intrusion et l'analyseur des réseaux,Voicide quoi il s'agit [13] :

2.5.1 Un antivirus

Un antivirus est un logiciel qui a pour but de détecter et de supprimer les virus d'un système informatique. Pour y arriver, l'antivirus dispose de plusieurs techniques comme la recherche par la signature qui consiste à analyser l'ensemble de la mémoire de stockage (disque dur), ou l'analyse qui consiste à simuler le comportement des logiciels, ou encore l'analyse du comportement qui consiste à surveiller les logiciels actifs, les antivirus plus connus : Kaspersky (payant), avast (gratuit).

2.5.2 Un pare-feu

Un pare-feu (en anglais Firewall) est un système permettant de séparer un réseau interne d'un réseau externe (souvent l'Internet). Il permet de filtrer les

communications dans les deux sens et ainsi protéger le réseau interne des éventuelles menaces provenant de l'extérieur.

Ce dispositif est constitué de matériels, routeurs et ordinateurs, et de logiciels pour la partie active et la configuration.

Un pare-feu est un ensemble de composants placés entre deux réseaux ayant les propriétés suivantes :

- tout le trafic transitant entre les deux réseaux passe nécessairement par le pare-feu.
- Seul le trafic explicitement autorisé par la politique de sécurité appliquée localement est autorisé à passer au travers du pare-feu.

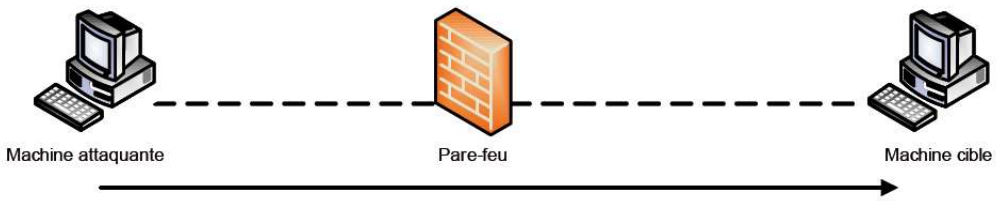

Figure 2.4 : Pare-feu

2.5.3 Analyseur de paquet

Un analyseur de paquet est un logiciel pouvant lire ou enregistrer des données transitant dans le réseau, il permet de capturer chaque paquet du flux de données il existe plusieurs analyseur de réseau exemple TCPdump, tshark, le plus connu est Wireshark.

a Wireshark

Wiresharck est un analyseur de paquets réseau, qui analyse et les capture en temps réel, fonctionnant sur tous les environnements et reconnaissant pratiquement tous les protocoles informatique existants.

Avec Wireshark, il est possible de capturer des paquets directement sur les interfaces du système utilisé ou de lire des fichiers de captures sauvegardées. Wireshark supporte les formats de fichiers de capture de libpcap.

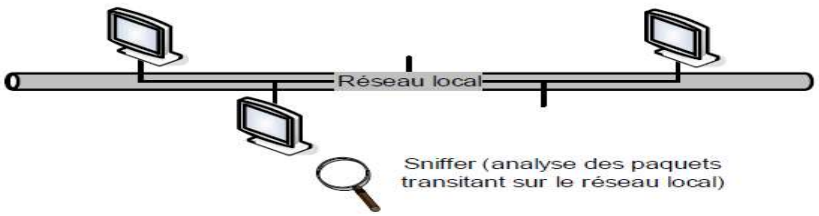

Figure 2.5 : Analyseur de réseau

b Tcpdump

Tcpdump est un analyseur de paquets en ligne de commande. Il permet d'obtenir le détail du trafic visible depuis une interface réseau.

Il permet d'écrire des informations dans unfichier en vue d'une analyse ultérieure,Affiche l'en-tête des paquets reçus par uneinterface donnée [17].

2.5.4 Système de détection d'intrusion (IDS)

Un système de détection d'intrusion (ou IDS : intrusion detection système) est un mécanisme destiné à repérer 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 Il existe deux grandes familles distinctes d'IDS [18] :

a Les N-IDS (Network Based IDS)

Le rôle essentiel d'un IDS réseau est l'analyse et l'interprétation des paquets circulantsur ce réseau.

Il fonctionne de trois manières différentes :

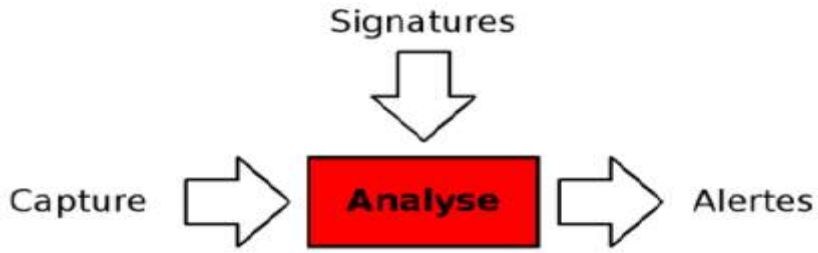

Figure 2.6 : Fonctionnement de NIDS

La capturassions des paquets après il les analyses et il caractérise l'attaque avec sa signature, pour pouvoir crée une règle pour détecter cette intrusion [18].

b Les H-IDS (Host Based IDS)

Les systèmes de détection d'intrusion basés sur l'hôte ou HIDS (Host IDS) analysentexclusivement l'information concernant cet hôte. Comme ils n'ont pas à contrôler letrafic du réseau mais "seulement" les activités d'un hôte ils se montrenthabituellement plus précis sur les types d'attaques subies.

Pour finir, voici quelques HIDS connus : Tripwire, WATCH, Dragon Squire, Tiger,Security Manager, Snort… [18].

2.6 Snort

2.6.1 Définition

A l'origine écrit par Martin Roesch, Snort est la solution la plus répandue pour les parties IDS et IPS. C'est d'une part le fait qu'il soit libre (publié sous licence GNU GPL) et donc que le fonctionnement soit accessible au plus grand nombre et d'autre part sa modularité qui permet à tous de participer à l'augmentation du nombre de préprocesseurs, de règles et de modules de sortie disponibles. Cependant, la recherche de trames qui circulent sur le réseau peut être longue et fastidieuse, mais si l'effort est récompensé, cela vaut sûrement la peine [18].

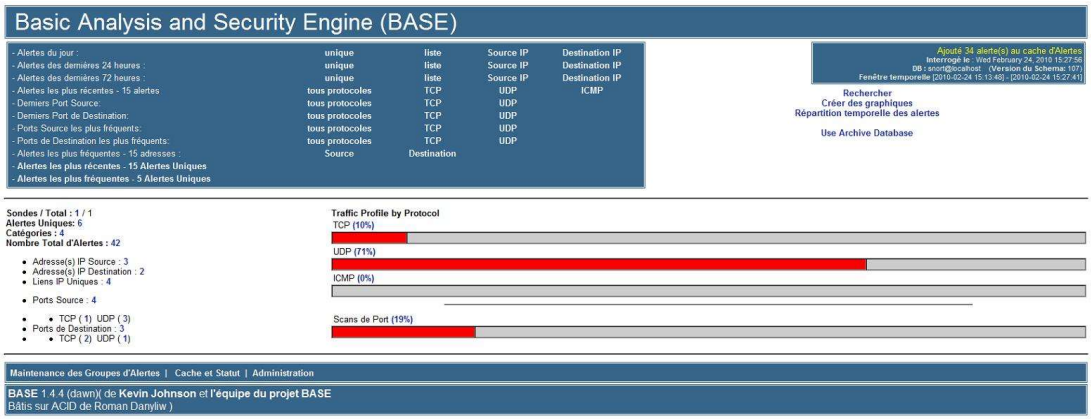

Figure 2.7 : Interface BASE de Snort [14]

SNORT permet d'analyser le trafic réseau de type IP, il peut être configuré pour fonctionner en plusieurs modes [14].

2.6.2 Définition de la règle Snort

Les règles de SNORT sont composées de deux parties distinctes : le header et les options.

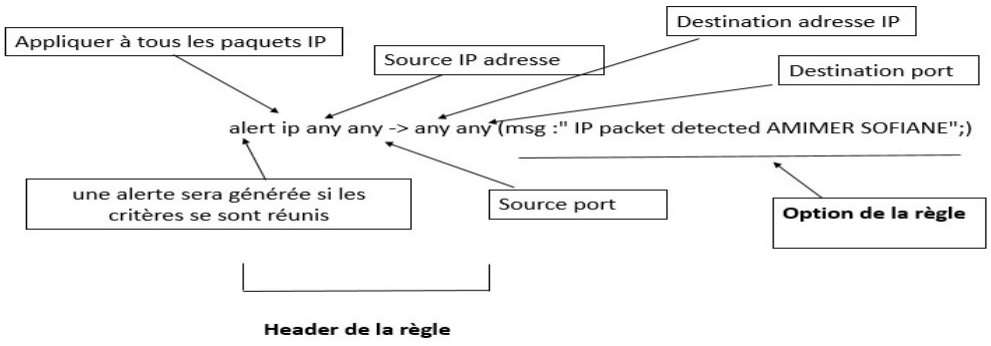

Figure 2.8 : Exemple d'une règle de Snort

Le header : permet de spécifier le type d'alerte à générer (alerte, log…) et d'indiquer les champs de base nécessaires au filtrage : le protocole (TCP, UDP ou ICMP), ainsi que les adresses IP et ports sources et destination.

Les options, spécifiées entre parenthèses : permettent d'affiner l'analyse, en décomposant la signature en différentes valeurs à observer parmi certains champs du header ou parmi les données.

L'entête de règle contient l'information qui définit le "qui, où, et quoi" d'un paquet, ainsi que quoi faire dans l'événement où le paquet avec tous les attributs indiqués dans la règle devrait se présenter. Le premier élément dans une règle est l'action de règle. L'action de règle dit à Snort quoi faire quand il trouve un paquet qui correspond aux critères de la règle [14].

Les différentes règles, options de règle disponibles dans Snort :

option	Description
Alert	génère une alerte en utilisant la méthode d'alerte sélectionnée, et alors journalise le paquet
msg	affiche un message dans les alertes et journalise les paquets
threshold	Indiquez les alertes de seuil chaque fois que nous voyons cet événement pendant l'intervalle de temps
Track by-src	Le taux est suivi soit par l'adresse IP source, soit par l'adresse IP de destination. Cela signifie que le compte est maintenu pour chaque adresse IP source unique ou pour chaque adresse IP de destination unique. Les ports ou toute autre chose ne sont pas suivis.
second	période sur laquelle le compte est accumulé. s doit être une valeur différente de zéro.
count	nombre d'appariement des règles en s secondes qui entraînera un dépassement de la limite du filtre d'évènement. c doit être de valeur non nulle.
flags	teste les drapeaux TCP pour certaines valeurs
dsize	teste la taille de la charge du paquet contre une valeur
itype	teste le champ type ICMP contre une valeur spécifiée
icode	teste le champ code ICMP contre une valeur spécifiée
sid	identifiant de signature contenue dans la base de signature
rev	est utilisé pour identifier de manière unique les révisions des règles Snort.

Tableau 2.2 :Option de règle deSnort

2.6.3 Modes de fonctionnement de Snort

Snort fonctionne en 4 modes :

Sniffer (Snort -vde) : Comme TCPdump et Wireshark, permettant d'écouter le trafic réseau en des points stratégiques.

Générateur de log (Snort -vde -l. /log) : Permettant le débogage des attaques en cours ou passées.

NIDS (Intrusion Detection System) : Détecteur d'anomalies permettant de capter des intrusions (Sécurité Passive).

IPS (Intrusion Prevention System) : Permettant la prévention des intrusions sur le réseau (Sécurité Active) Snort_Inline [14].

2.6.4 L'architecture de Snort

L'architecture de Snort est composée comme suit :

Noyau de base : au démarrage, ce noyau charge un ensemble de règles, compile,optimise et classe celles-ci. Durant l'exécution, le rôle principal du noyau est la capture de paquets.

Une série de préprocesseurs, ceux-ci améliorent les possibilités de SNORT en matière d'analyse et de recomposition du trafic capturé. Ils reçoivent les paquets directement capturés, éventuellement les retravaillent puis les fournissent au moteur de recherche de signatures.

Une série d'analyses est ensuite appliquée aux paquets. Ces analyses se composentprincipalement de comparaisons de différents champs des headers des protocoles (IP,ICMP, TCP et UDP) par rapport à des valeurs précises.

Après la détection d'intrusion, une série de « output plugins » permet de traiter cette intrusion de plusieurs manières : envoie vers un fichier log, envoie d'un message d'alerte vers un serveur Syslog ou encore stocker cette intrusion dans une base de données SQL [14].

2.7 Conclusion

Dans ce chapitre, nous avons cité les différents types d'attaques ainsi que les outils à utiliser et la stratégie à entreprendre afin de mettre en place une politique de sécurité.

Nous allons aussiprésenterle système de détection d'intrusion et l'un de ses modèles (Snort).

Mais malgré tous cela les pirates informatiques ne manquent pas d'ingéniosité pour inventer d'autres outils et méthodes d'attaques, ce qui pousse à être à jours en s'informant sur les nouvelles techniques d'attaques et les nouveaux outils et mécanismes de sécurité.

Nous allons procéder à la phase de simulation des attaques au niveau du chapitre 3 à partir de Wireshark, afin d'analyser les paquets extrait des attaques et les comparer avec les cas normaux pour obtenir les signatures des attaques.

Chapitre 3 Simulation des attaques

3.1 Introduction

L'analyseur de trafic est un outil pédagogique essentiel pour comprendre les mécanismes de fonctionnement des protocoles de communication sur les réseaux. Ce chapitre comprend deux parties. Dans un premier temps, Wireshark nous aidera à identifier les signatures lors de la simulation, le logiciel libre incontournable en la matière. Dans un deuxième temps, les travaux pratiques permettent de découvrir les empreintes des attaques.

3.2 Méthode de travail

Nous avons simuléles attaques dans un environnement réelLAN, constitué de deux pc qui son reliéà un switch via un câble torsadé :

- Le pc pirate : IP 172.20.3.186 (classe b), la ram 4Go, processeur intel core i3 cpu 2 Ghz.
- Le pc cible : IP 172.20.18.11 (classe b), la ram 4Go, processeur intel core i3 cpu 2 Ghz.
- Câble torsadé catégorie 5 (1 Gbits /s).

Ainsiqu'on a utiliséLOIC comme logiciel d'attaque décritprécédemment (chapitre 2), cetteattaque influence sur l'utilisation de la mémoire, processeur et réseau.

L'outil ouvre plusieurs connexions au serveur cible et envoie une suite continue de messages qui peuvent être définis à partir de l'option de paramètre de message TCP

/UDP disponible sur l'outil. Dans les attaques TCP et UDP, la chaîne est envoyée en texte brut mais dans l'attaque HTTP, elle est incluse dans le contenu d'un message HTTP GET.

Cet outil continue d'envoyer des requêtes au serveur cible ; après un certain temps, le serveur cible devient surchargé. De cette façon, le serveur cible ne pourra plus répondre aux demandes des utilisateurs légitimes, ce qui le fermera efficacement.

Une fois l'attaque est lancé nous allons voir qu'elles sontles paramètres causant la perturbation de système cible, à travers l'analyseur de paquet Wireshark.

Méthode de la simulation d'attaque :

La simulation va être réalisée en 3 étapes :

1 er étape : exécuter LOIC

2 ème étape : choisir les paramètres d'attaque.

Valider l'adresse IP de notre cible dans le champ IP et cliquer sur le bouton verrouiller ensuite la choisir méthode d'attaque, le port.

L'envoie la chaine de message (exemple : WALID HADIDI) avec une vitesse (faster), après on clique sur le Big Button intitulé "IMMA CHARGIN MAH LAZER". Nous venons d'attaquer la cible.

3 ème étape : exécuter Wireshark pour visualiser les paquets.

3.2.1 Wireshark

Pour étudier les paquets qui circulent dans le réseau, on utilise Wireshark qui capture les paquets. Ces derniers seront développés et détaillés.

Une fois Wireshark est lancé, la fenêtre de la capture ce présente sur la figure suivante :

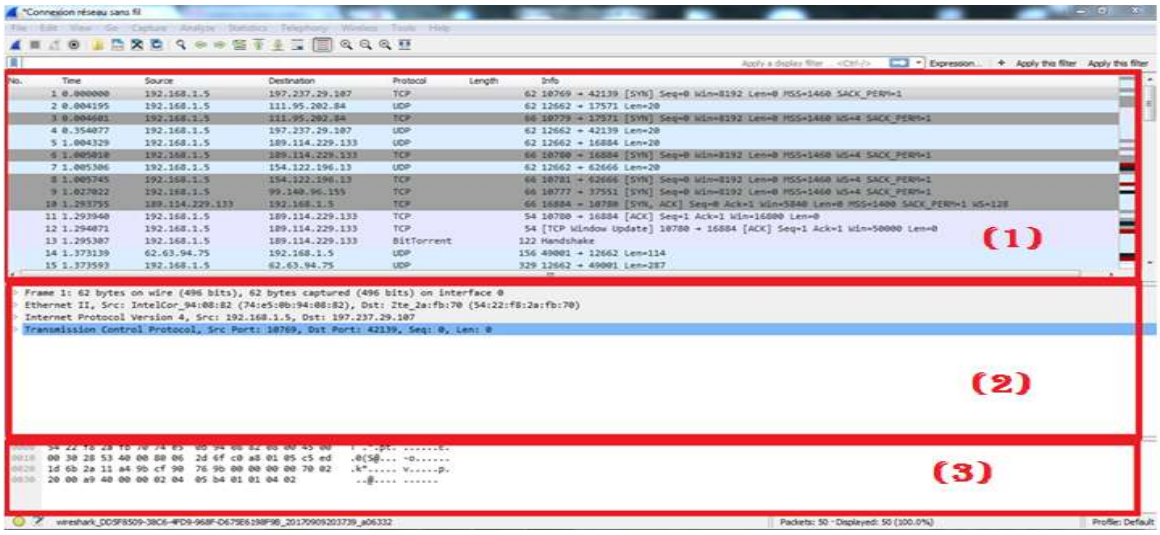

Figure 3.1 : Fenêtre de Wireshark

La fenêtre suivante montre le détail des entêtes d'un paquet :

- Zone numérotée (1) sur figure 3.1 : liste l'ensemble des paquets capturé
- Zone numérotée (2) sur figure 3.1 : affiche le détail d'un paquet sélectionné
- Zone numérotée (3) sur figure 3.9 : présente l'ensemble du paquet sous forme octale et ASCII.

La fentre suivante montre le détail des entetes d'un paquet :

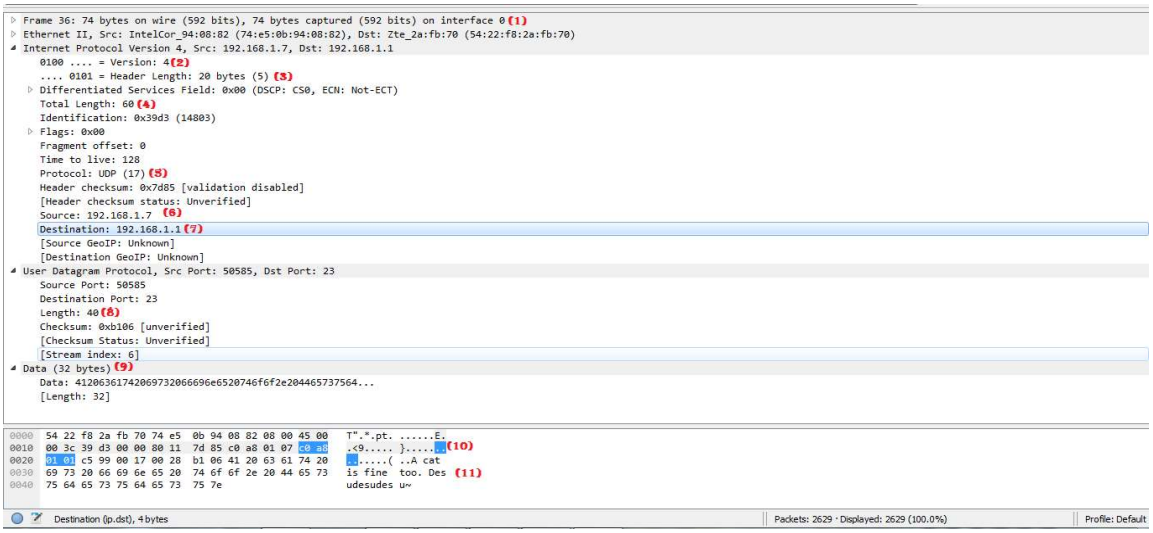

Figure 3.2 : Visualisation détaillée des en-têtes d'un paquet

	Référence
Le paquet est de type IP v4	Ref(2) sur figure 3.2
Le type de donné de ce paquet IP est un datagramme UDP	Ref (5) sur figure 3.2
L'IP de la machine source est 192.168.1.7	Ref (6) sur figure 3.2
L'IP de la machine destination est 192.168.1.1	Ref (7) sur figure 3.2

Tableau 3.1 : Les étapes d'encapsulation

Nous pouvons également faire un point sur la taille des données et des en-têtes à différents niveaux d'encapsulation :

	Référence
La taille des données envoyée par le le processus est de 32 octets	Ref(9) sur figure 3.2
La taille totale du datagramme UDP est de 40 octets. Cette valeur est la somme entre la taille réelle des données 23 octets et 8 octets d'en-tête du paquet.	Ref(8) sur figure 3.2
La taille des en-têtes du paquet IP est de 20 octets	Ref(3) sur figure 3.2
Le paquet IP contient un en-tête (20 octets) ainsi que le datagramme UDP (40 octets). Sa taille totale est de 60 octets, taille rappelée	Ref(4) sur figure 3.2
L'adresse source est codée en octets	Ref(10) sur figure 3.2
Le contenant de paquet on peut le voir sous le codage ascii	Ref(11) sur figure 3.2
Si l'on ajoute 12 octets d'en-tête pour la couche Ethernet 2 (taille fixe), la taille totale de la trame et de 74 octets, comme présentée en ref (1) sur figure 3.10	ref (1) sur figure 3.2

Tableau 3.2 : Donnée des entêtes au niveau d'encapsulation

Notons ainsi que pour transférer 32 octets de données brutes, il nous a fallu transférer au totale 74 octets (en fait il nous a même fallu transférer des octets supplémentaire avant la trame Ethernet. Ces octets seront ici passés sous licence)

3.2.2 Visualisation des paquets TCP :

Dans cette partie-là on va étudier le cas normal de protocoleTCP et celle mal formé des attaques via **l'outil LOIC** avec l'utilisation de Wireshark, pour constater la différence afin de caractériser les attaques avec des signatures :

a Visualisation de paquetTCP normal

On prend un exemple d'une connexionentre deux PC à la base de protocoleTCP, illustré sur la figure suivante :

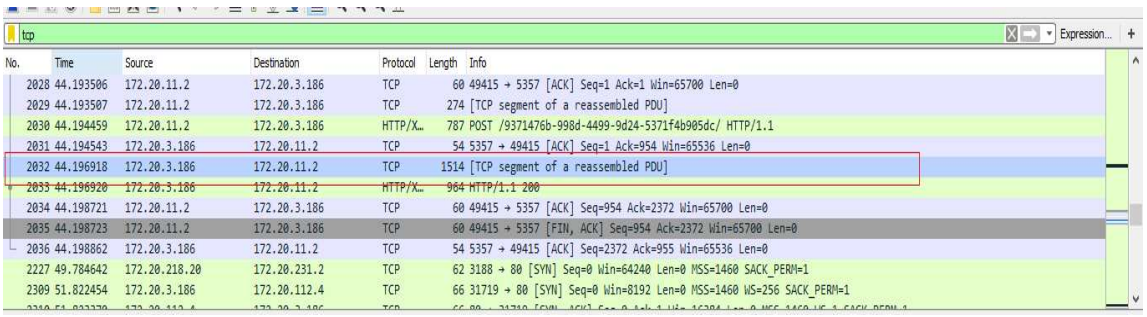

Figure 3.3 : La capture des paquets TCP normale

Pour plus de détails, on a pris une autre capture au niveau d'encapsulation :

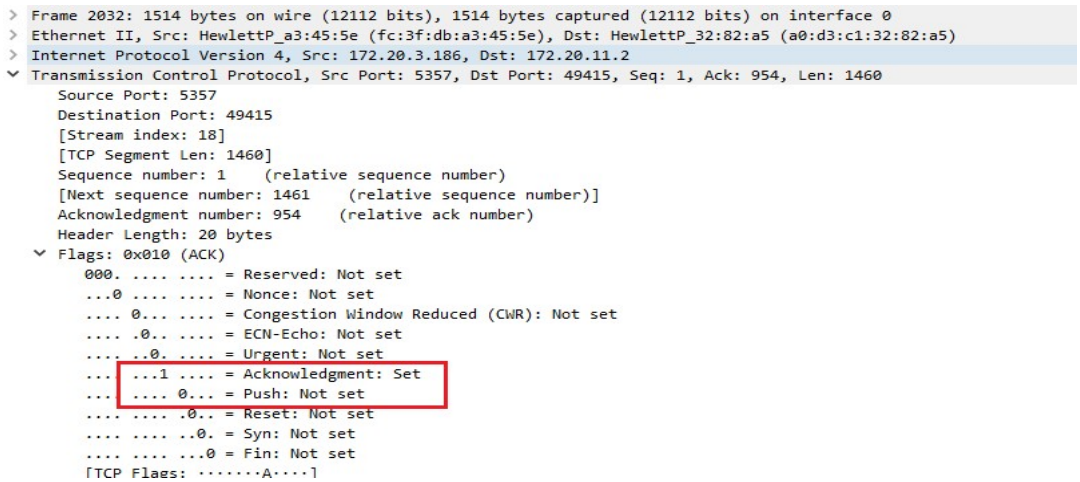

Figure 3.4 : La capture détaillée de paquet TCP

Notre pc : IP = 172.20.3.186 (classe b)

Pc destination : IP=172.20.11.2

Lors d'une connexionTCP dans un réseau local de notre université, on note :

Notre Pc répond par un TCP de flag ack lorsque le message est reçu.

Dans la figure 3.4 : les paquets TCP entrant ont des flags push non positionné et ack=1.

Ce qui concerne le flux de donné, Wireshark permet d'analyser le flux sous forme d'un graphe, illustré dans la figure suivante :

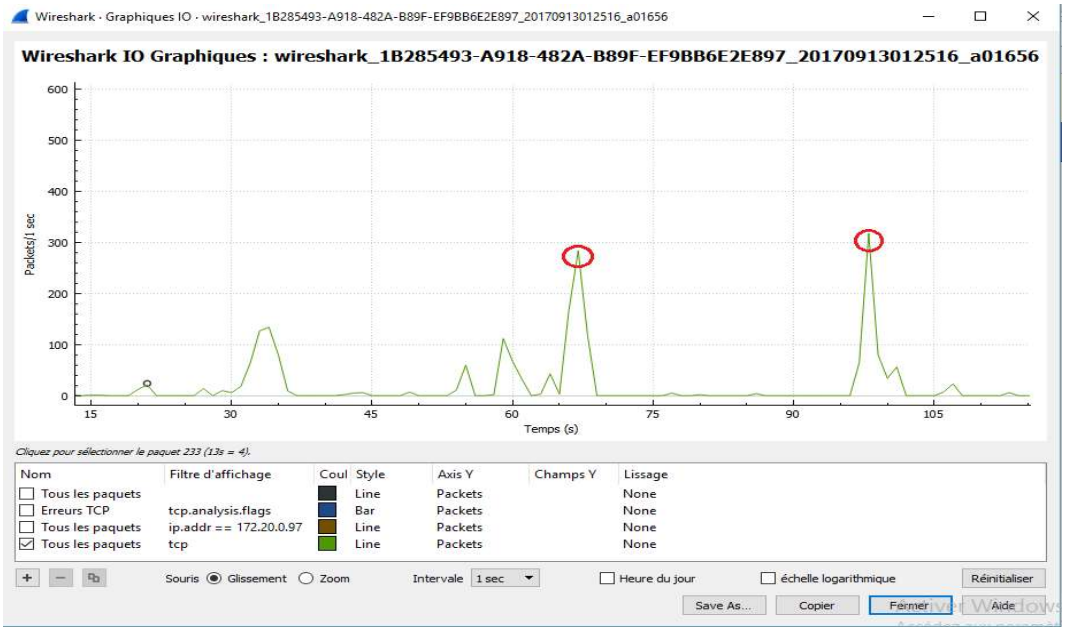

Figure 3.5 : Le flux des données de TCP

On remarqué que la quantité des données ne dépasse pas les25 paquets par seconde durant la connexion, mais elle peut atteindre300 paquets par seconde pour une durée très courte de 3 secondes jusqu'à 5 secondes.

b Visualisation des paquets d'attaque TCP via LOIC

Étape 1 : exécution de l'outil.

Étape 2 : on a entré l'adresse IP 172.20.18.11(classe b) de notre cible dans le champ IP et cliqué sur verrouiller (lock on) ensuite la méthode d'attaque (TCP) etenfin on choisit le port 80.

Étape 3 : on envoie la chaine de message par défaut (A cat is fine too Desudesudesu~) avec une vitesse maximale (faster), après on clique sur le Big Button intitulé "IMMA CHARGIN MAH LAZER". Nous venons d'attaquer la cible.

On met la vitesse d'envoi des paquets en slower (vitesse lente).

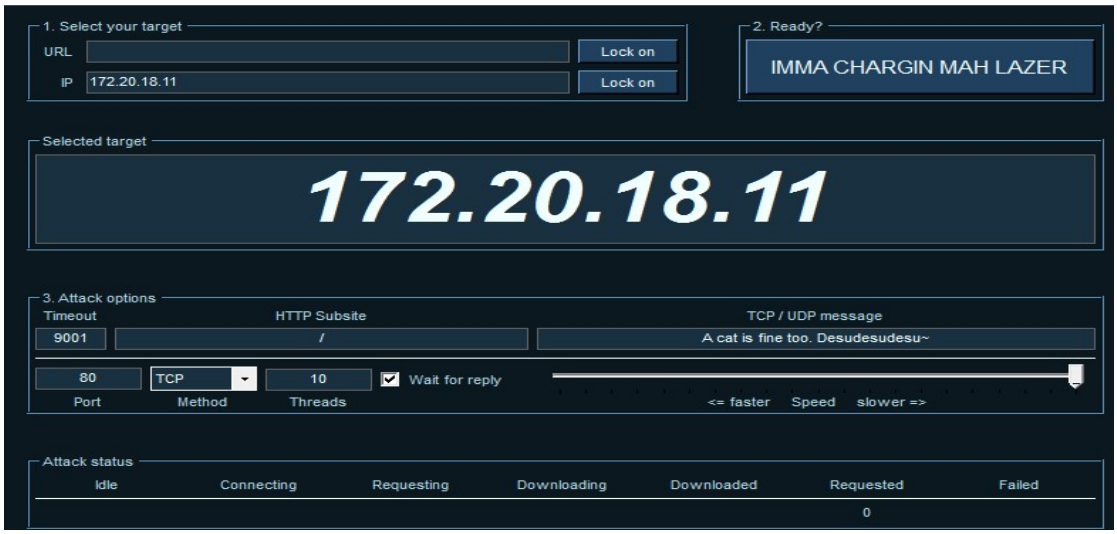

Figure 3.6 : Attaque TCPLOIC

Durant l'attaque, on exécute Wireshark pour analyser l'attaque :

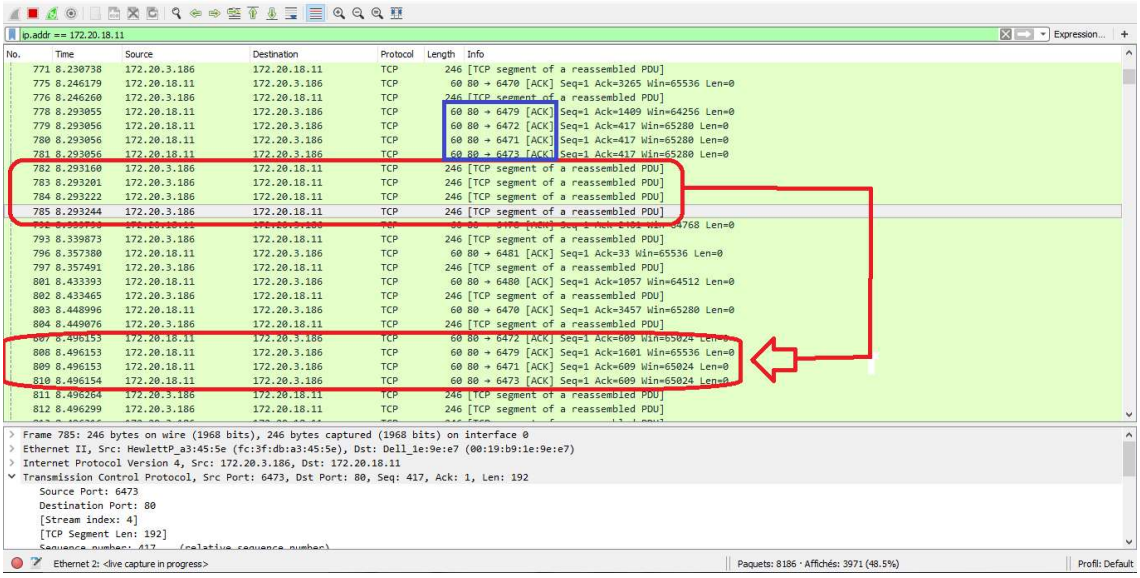

Figure 3.7 : Capture de l'attaque TCP

Pc pirate : IP=172.20.0.10.168 (la classe b)

Pc cible : IP=172.20.11.18 (la classe b)

On remarque pour chaque paquet envoyé vers le pc cible répond par un TCP de flag ack et qu'il ouvre plusieurs connexion (entouré en bleu figure 3.6) à partir de différent port (par exemple 6779, 6472).

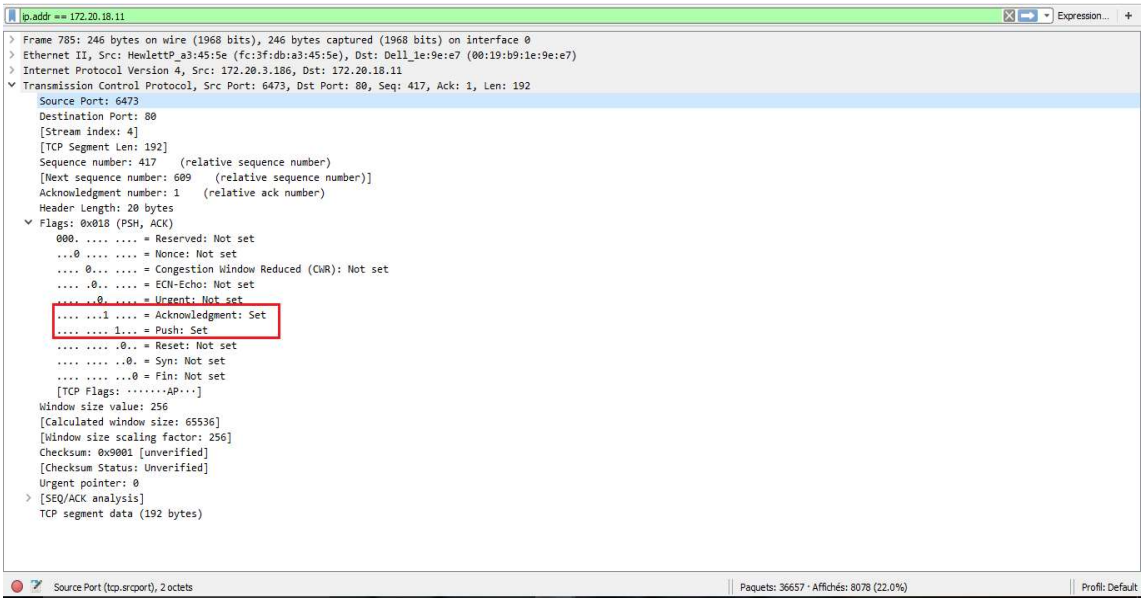

Figure 3.8 : Capture détaillé d'un paquet TCP

Lorsqu'un émetteur TCP envoie un paquet avec le flag push égale à 1, le résultat est que les données TCP sont immédiatement envoyées ou "poussées" au récepteur TCP.

Et normalement dans le cas normal d'une connexion TCPle flag push=0 presque dans tous les paquets, figure 3.8.

Durant notre analyse de la figure 3.7 on note que les flags TCP et push sont positionnés.

Dans la figure 3.9 suivante, on peut visualiser le message envoyé par notre pc d'attaque via LOIC :

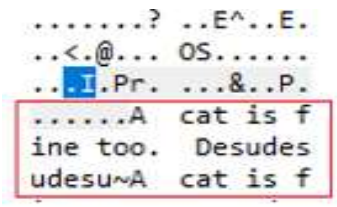

Figure 3.9 : Capture de message (en ascii) ennoyé par LOIC

Le graphe suivant montre le flux de données de l'attaque TCP :

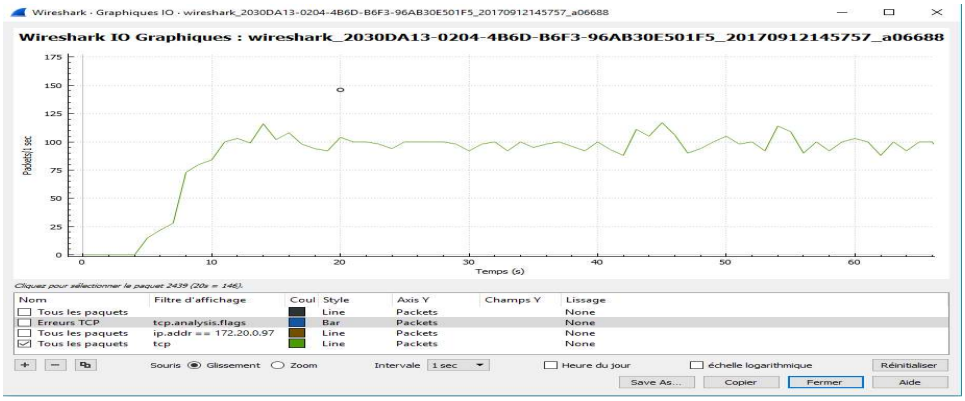

Figure 3.10 : Le graphe de paquet TCP (attaque)

On note que la moyenne de l'envoi des paquets est de 100 paquets par second pendant la durée d'attaque.

D'après notre visualisation des paquets de protocoleTCP normal et celles des attaques on a pu définir la signature de cette attaque, illustrée dans le tableau suivant :

	Cas d'un paquet TCP normale	Cas d'un paquet d'attaque de TCP	La différence entre les deux cas
Le nombre de paquets par second	25 jusqu'à 300 (d'une durée de 2 à 5 seconde)	La moyen des paquets se stabilise entoure de 100	L'attaque tcp elle se stabilise les environs de 100 par contre le cas de tcp normale elle se stabilise au niveau de 25 et atteint 300 dans des durées très courtes
Les flags	Flag PUSH=0 Flag ACK=1	Flag PUSH=1 Flag ACK=1	Le flag PUSH est positionné (égale a 1)

Tableau 3.3 : La déférence entre les deux cas de protocoleTCP(les signatures)

c L'impact de l'attaque TCP

Le cas d'un état normal de PC :

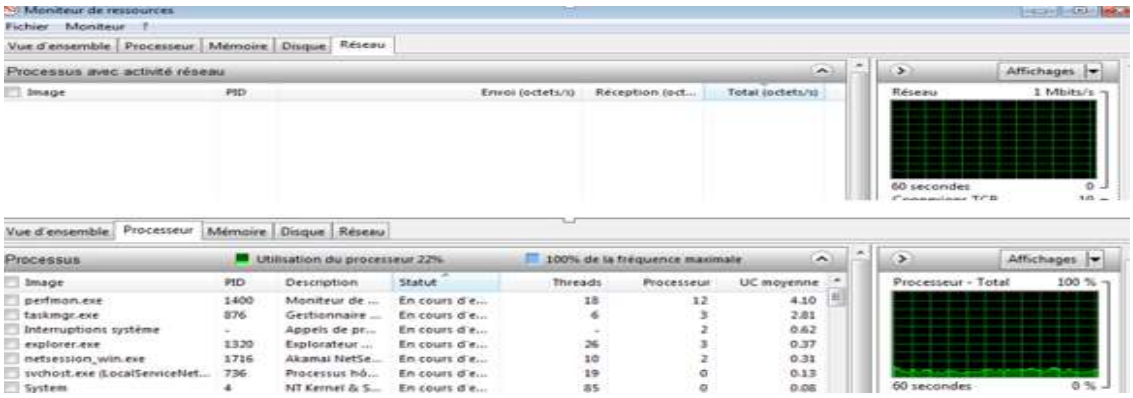

Figure 3.11 : Capture del'utilisation du processeur et réseau de TCP normal

Le cas où l'ordinateur subit une attaque TCP :

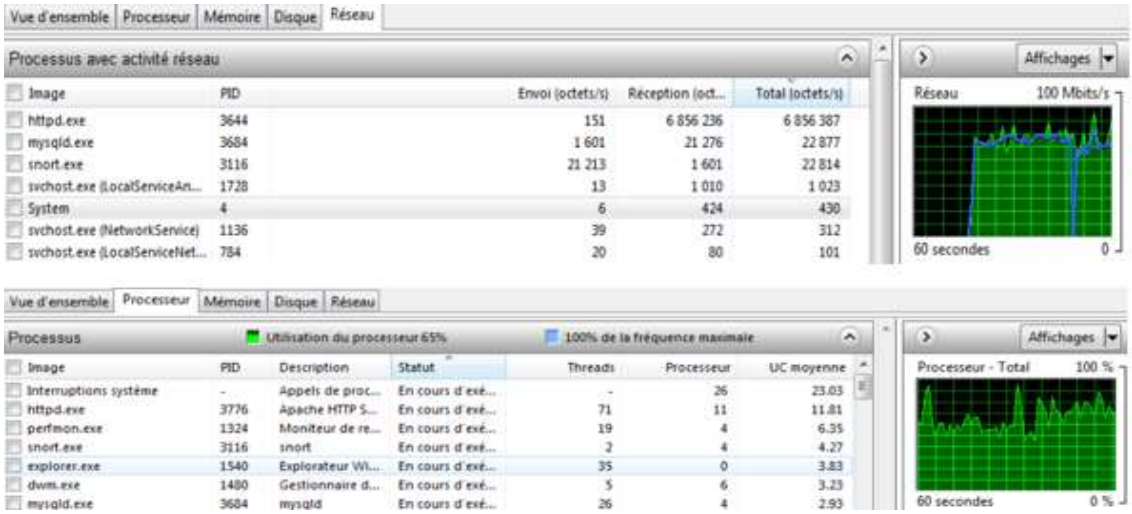

Figure 3.12 : Capture de l'utilisation du processeur, réseau de l'attaque TCP

On remarque sur la figure 3.11 précédente que le niveau de l'utilisation de processeur et de réseau est presque nul dans le cas où y'a pas d'intrusion (attaque).

Par contre sur la figure 3.12 on note une grande augmentation au niveau de l'utilisation de processeur et réseau (réseau est saturé).

Cette attaque force réellement le serveur récepteur pour vider son tampon de pile TCP et pour envoyer une reconnaissance (acknowledgement) lors de déroulement de cette action, ce qui entraîne une condition de déni de service ou attaque TCP flood.

3.2.3 Visualisation des paquetsUDP

On va entamer la même procédure qu'on a fait dans la partie de visualisation de paquet TCP pour pouvoir obtenir la signature de l'attaque UDP, toujours avec l'utilisation d'analyseur de paquet Wireshark .

a Visualisation de paquet UDP normale

Prenant comme exemple les paquets de données d'une vidéo surYouTube, on lance la vidéo, on utiliseWiresharkpour capturer les paquets, voir la figure suivante :

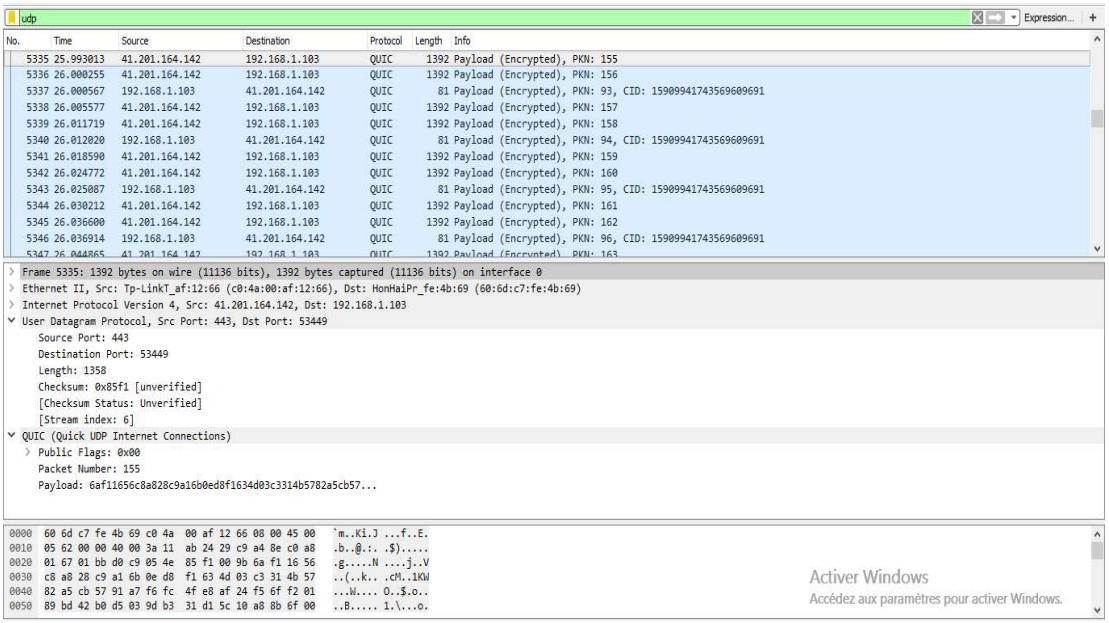

Figure 3.13 : Paquet UDP d'une vidéo de youtube

Notre pc : IP = 192.168.1.103,Server YouTube : IP=41.201.164.142

La figure 3.13 montre la synchronisation entre notre notre pc et le serveur YouTube en envoyant une ouverture de demande de connexion.

On voit bien que pour lancer une vidéo sur youtube on utilise le protocole UDP, illustré sur la figure3.13 précédente.

Le graphe suivant montre le flux de données de la vidéo du YouTube :

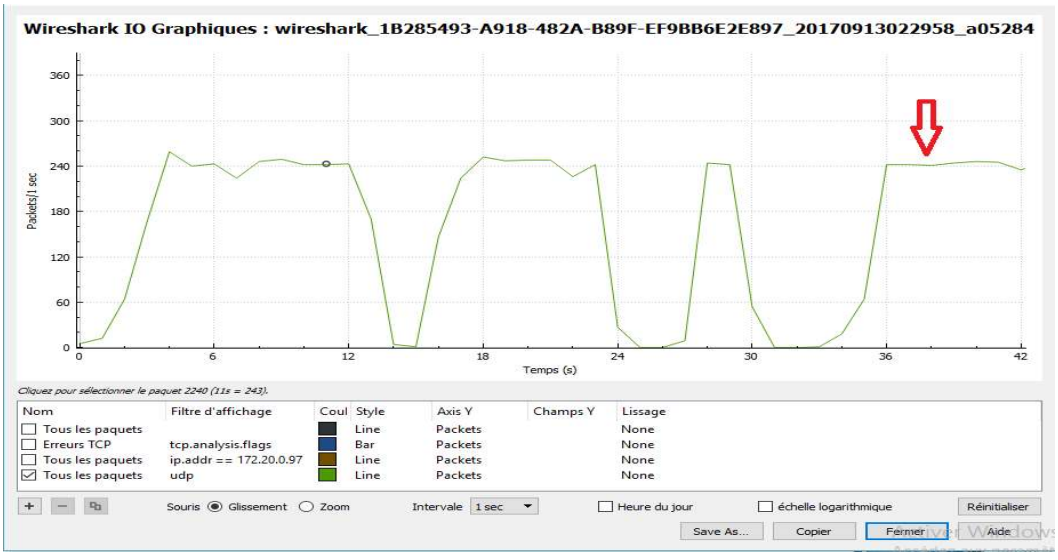

Figure 3.14 : Flux de donnée d'une vidéo de youtube

On remarqué que le nombre de la quantité des paquets ne dépasse pas 250 paquets par seconde durant toute la vidéo.

b Visualisation des paquets d'attaque UDP

Dans ce type d'attaque DOS, un serveur est inondé de paquets UDP. Contrairement à TCP, il n'y a pas de processus de communication entre client et hôte, cela rend plus difficile pour les mécanismes défensifs d'identifier une attaque d'inondation UDP.

Cette méthode est similaire à l'attaque TCP. On Sélectionne le type d'attaque comme UDP.

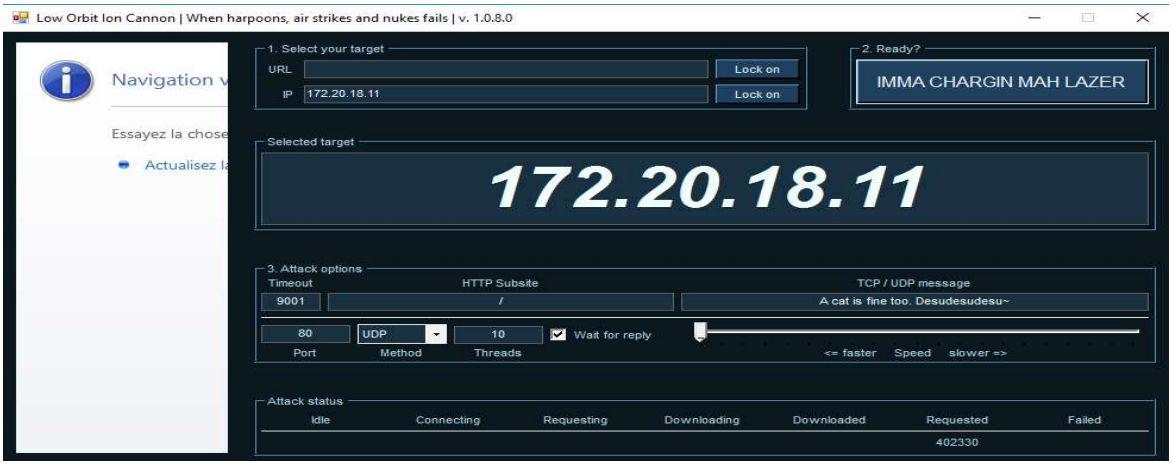

Figure 3.15 : L'attaque UDP

On exécute le Wireshark pour capturer les paquets d'attaque UDP, illustré dans la figure suivante :

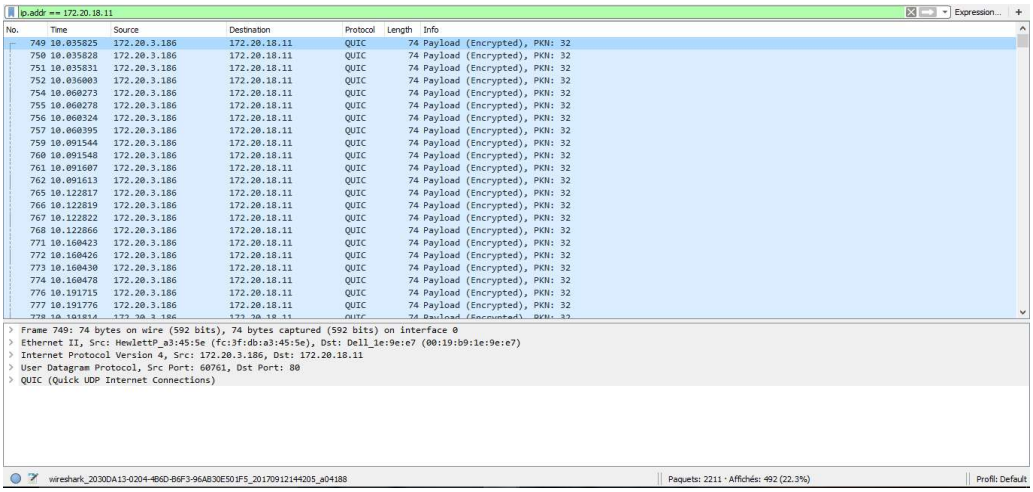

Figure 3.16 : Capture d'un paquet d'attaque UDP

Notre pc IP=172.20.3.186, Pc victime IP=172.20.18.11

Cette capture nous montre qu'il y'a plusieurs messagesUDP envoyés ce qui peut provoquer un déni de service.

Le graphe suivant montre le flux de données de l'attaque UDP :

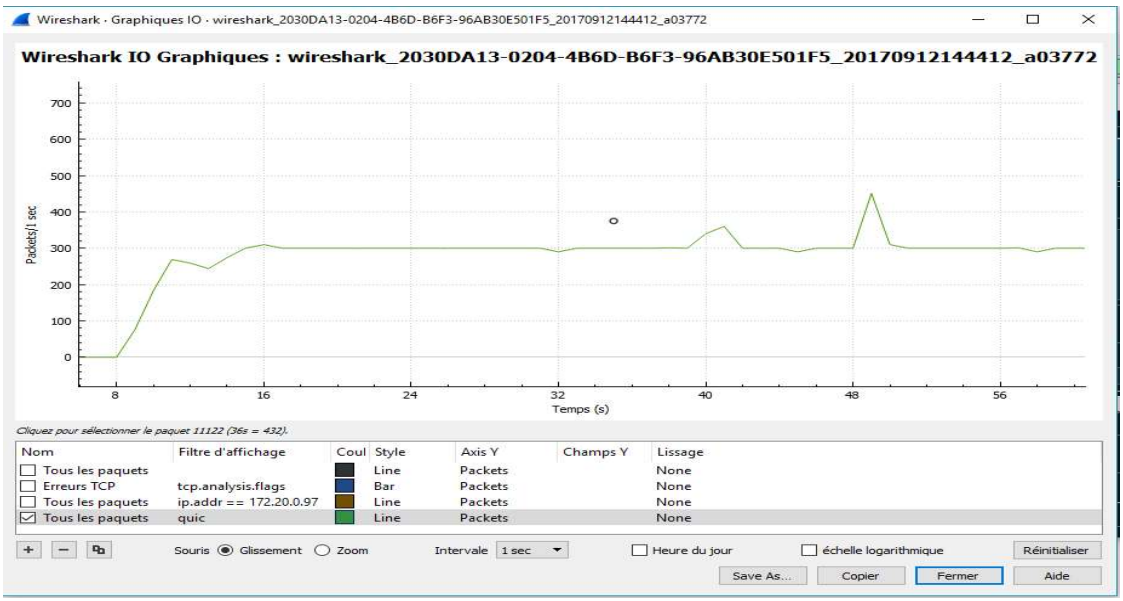

Figure 3.17 : Le flux de données de l'attaque UDP

La chose qui attire notre attention durant l'attaque UDPest le nombre des paquets par seconde qui est fixé à environ 300 paquets.

D'après notre visualisation des paquets de protocole UDP normal et celles des attaques on a pu définir la signature de cette attaqueUDP, illustré dans le tableau suivant :

	Cas d'un paquet UDP normal	Cas d'un paquet d'attaque UDP	La différence entre les deux cas
Le nombre des paquets par seconde	Ne dépasse pas 250 paquets.	Elle se stabilise à 300 paquets.	L'attaque UDP dépasse le nombre du paquet normal

Tableau 3.4 : La différence entre un paquet UDP normale et d'attaque

c L'impact de l'attaque UDP

Après avoir appliqué l'attaque UDP, on a remarqué sur le gestionnaire des taches de pc de la cible des augmentations dans les performances de réseau et le processeur, voir figures suivantes :

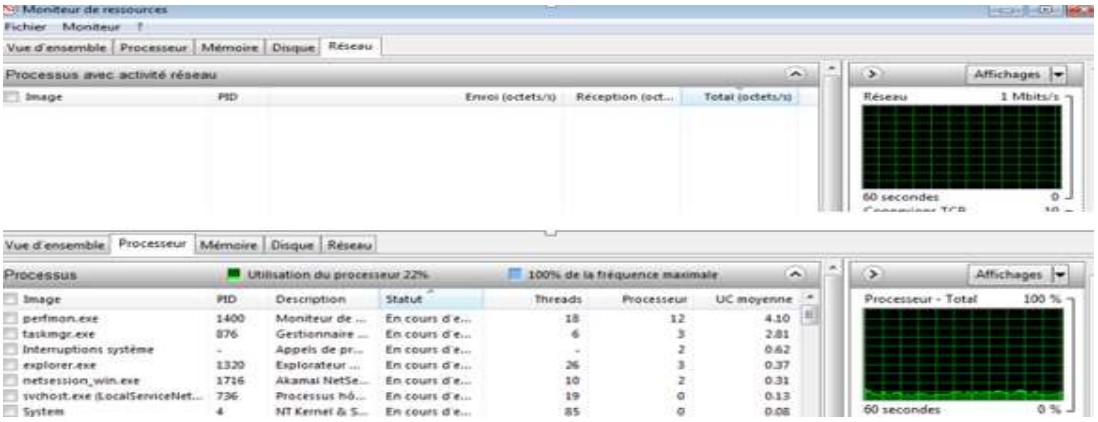

Figure 3.18 : Capture de l'utilisation du processeur et réseau avant l'attaque

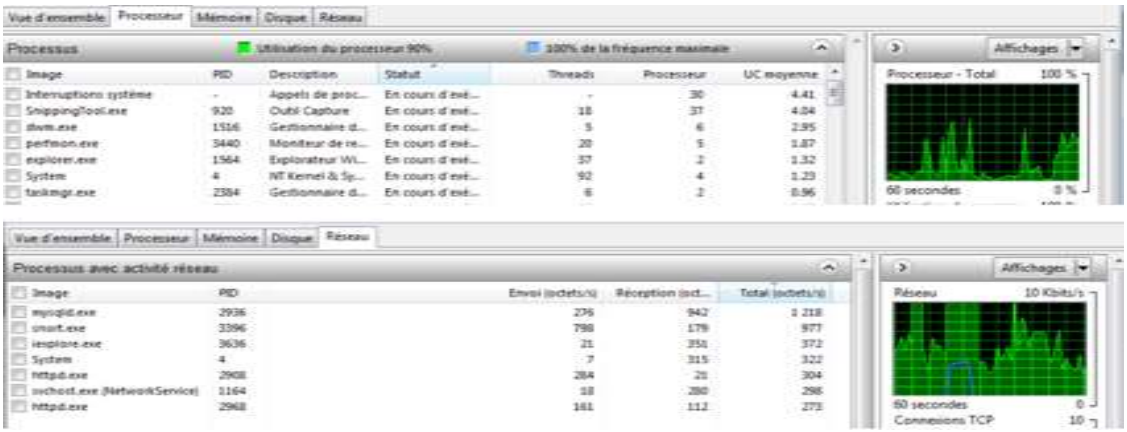

Figure 3.19 : La capture de l'utilisation du processeur et réseau après l'attaque UDP

Cette action engendre un grand changement au niveau de moniteur de ressources qui se manifeste par une énorme augmentation sur l'échelle de processeur et réseaux.

3.2.4 Visualisation des paquets HTTP

a Visualisation le paquet HTTP normale

Prenant comme exemple les requêtes de Google, présenté sur la figure suivante :

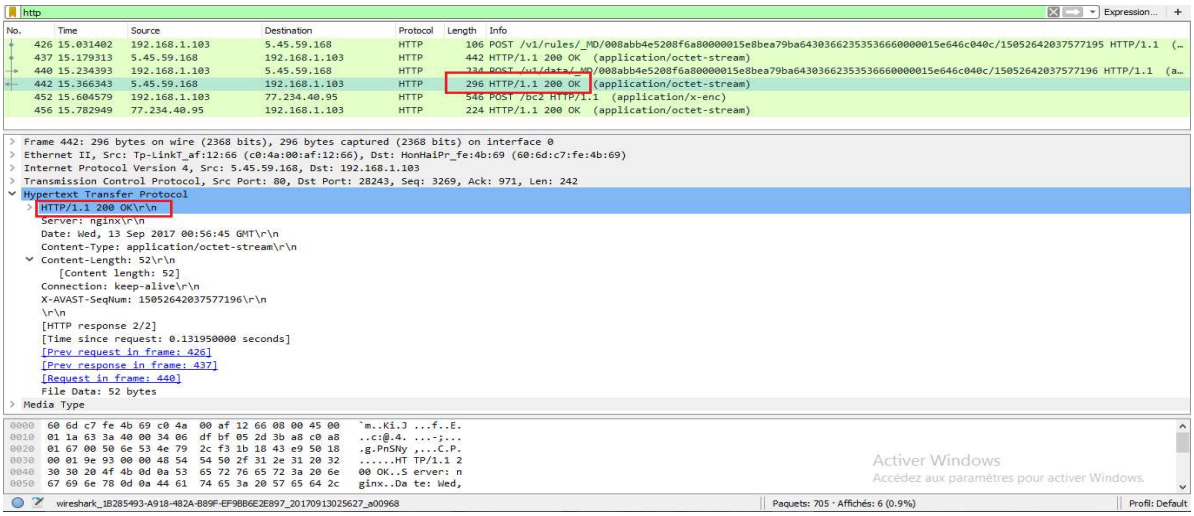

Figure 3.20 : La requête de HTTP

On note que la requête HTTP est sous la forme : HTTP/1.1 200 ok \r\n.

b Visualisationde paquet d'attaque HTTP

Dans cette attaque, l'outil **LOIC**envoie des requêtes HTTP au serveur cible. Dans l'attaque HTTPla chaîne elle est incluse dans le contenu d'un message HTTP GET.

On utilise la même procédure entamée dans les attaques précédente TCP et UDP, on sélectionne le type d'attaque HTTP et l'adresse de la cible vous pouvez également entrer l'adresse IP du système.

L'url de la cible :HTTP://www.univ-blida.dz

Adresse IP de la cible :193.194.83.181

Le port d'attaque est 80 car on utilise la méthode d'attaque HTTP via internat

Illustré dans les figures suivantes :

Figure 3.21 : L'attaque HTTP

On utilise Wireshark pour analyser notre réseau et la communication entre la cible et la victime :

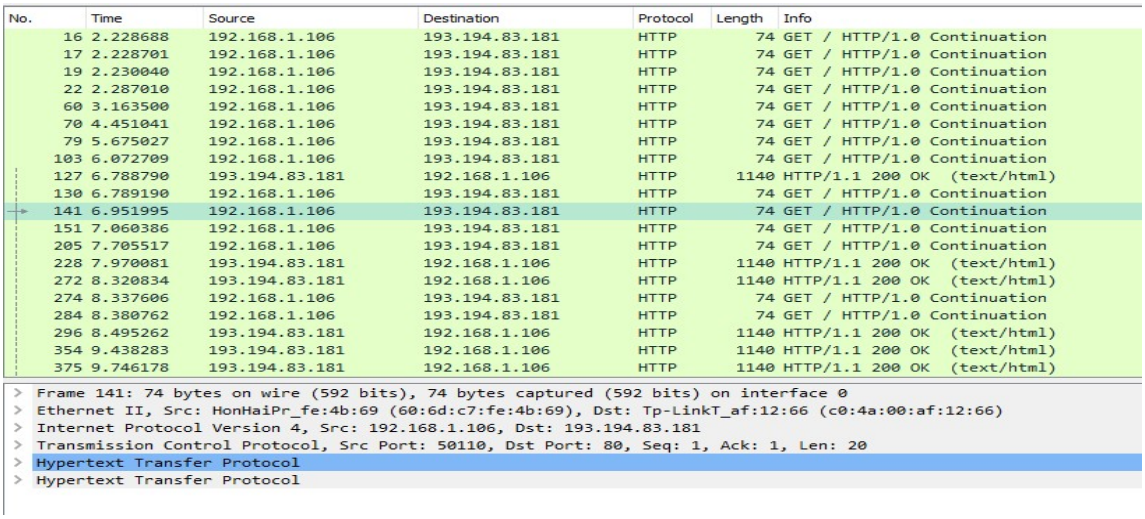

Figure 3.22 : La requête de l'attaque HTTP

On remarque que le serveur de la cible est inondé de requête HTTP

On a pris une autre capture d'information au niveau d'encapsulation :

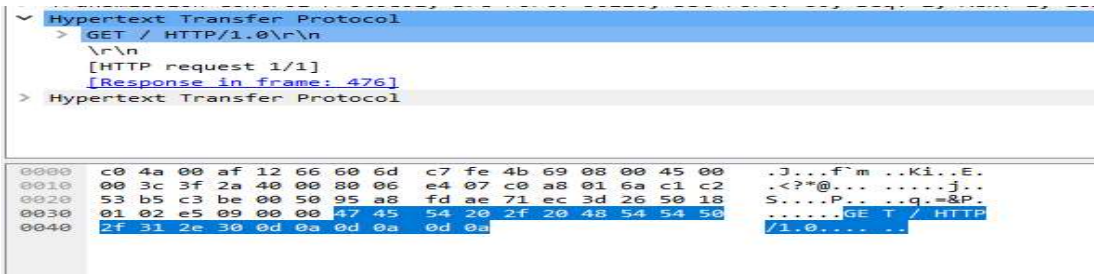

Figure 3.23 : La capture de requête HTTP détaillé

D'après la capture de Wireshark on note que la requête HTTP est sous la forme : HTTP/1.0.

D'après notre visualisation des paquets de protocole HTTP normal et celles des attaques on a pu définir la signature de cette attaque :

	Cas d'un paquet HTTP normal	Cas d'un paquet d'attaque HTTP	La différence entre les deux cas
La forme de la requête HTTP	http/1.1 200 ok \r\n	http/1.0	Mal formé

Tableau 3.5 : La différence entre le cas normale de HTTP et de l'attaque

Donc la signature de cette attaque : le format de la requête HTTP (HTTP/1.0\ r\n).

c L'impact de l'attaque HTTP

Une inondation HTTP est une méthode d'attaque utilisée par les pirates pour attaquer les serveurs Web et les applications. Il consiste à envoyer des ensembles de requêtes HTTP GET ou POST apparemment légitimes, à un serveur Web cible. Ces demandes sont spécifiquement conçues pour : consommer une quantité importante de ressources du serveur et peuvent donc entraîner une condition de déni de service.

3.2.5 visualisation des paquets ICMP

a visualisation des paquets ICMP (Ping)

Le Ping utilise le protocole ICMP, ce dernier est utilisé pour tester la connexion entre les pc. Les figures suivantes montrent le Ping et ces paquets envoyées d'hôte source vers l'hôtedestinateur :

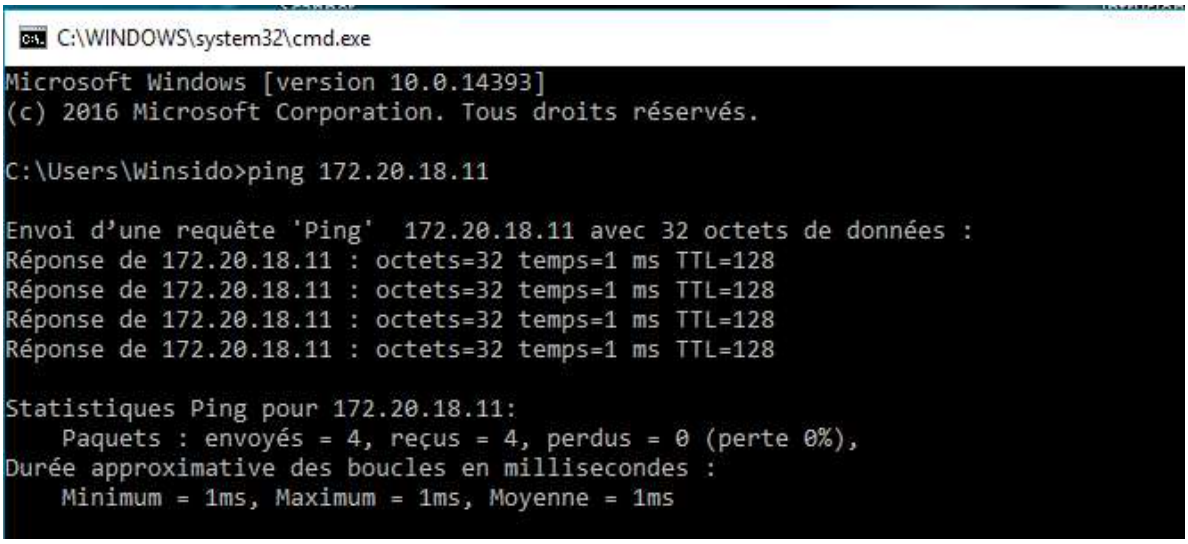

Figure 3.24 : Le Ping

Adresse source : 172.20.3.106, adresse destination : 172.20.18.11

Pour analyser les données envoyées par le Ping on utilise Wireshark, illustré dans la figure suivante :

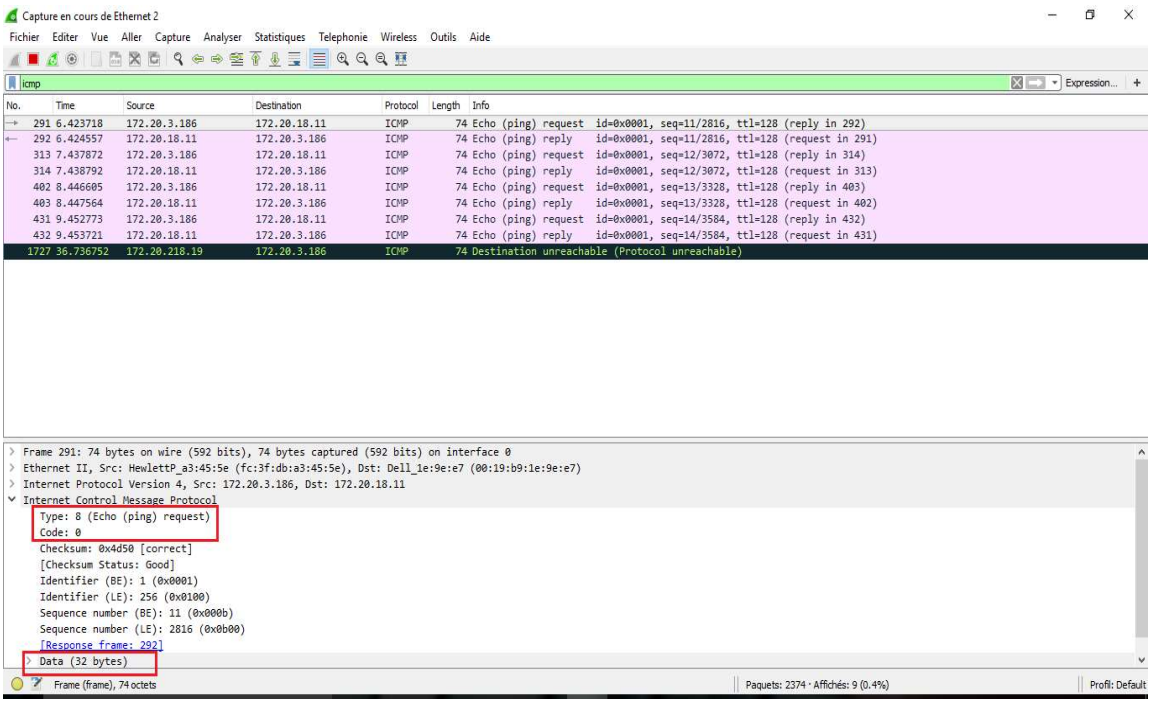

Figure 3.25 : La capture des paquets ICMP

On remarque que le Ping utilise le protocole ICMP qui envoie des messages de type 8, le code 0, ainsi que sa longueur est de 32 octets.

b Visualisation des paquets de ping of death

Pour réaliser cette attaque, la première étape consiste à ouvrir l'invite de commande (cmd).

Notre victime sera la machine : 172.20.18.11

Notre machine pirate sera : 172.20.3.186

On tape sur l'invite de commande cette commande : Ping –t –l 65500 10.0.0.3 pour pinge la cible.

-t : une boucle qui permet de répéter le Ping.

-l : détermine la taille de paquet envoyé.

Le résultat de cette attaque est illustré sur la figure suivante :

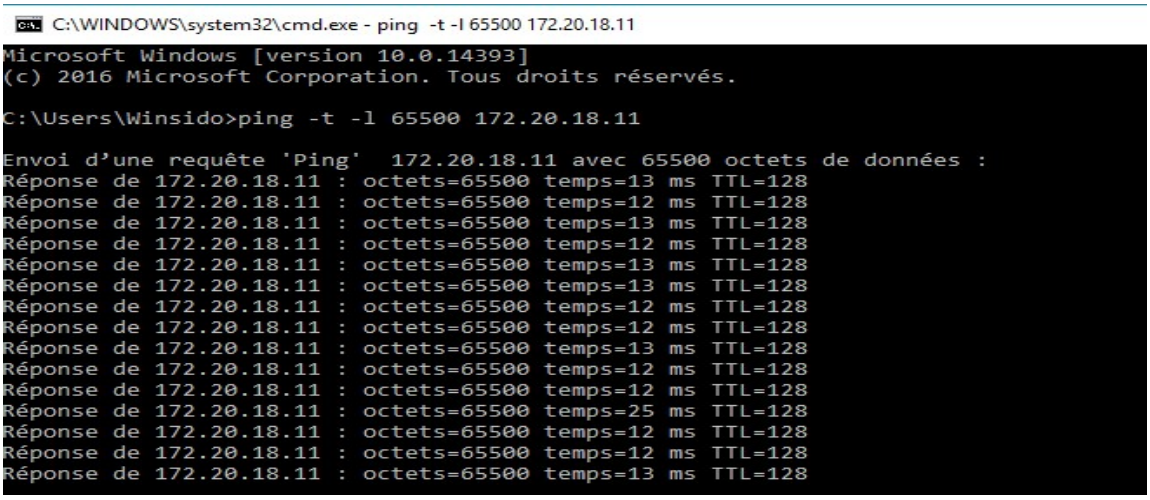

Figure 3.26 : Le Ping of death

Cette commande va envoyer des paquets ICMP vers la cible, c'est à dire des demandes de connexions et de teste réseau.

On utilise l'analyseur de trafic pour analyser le trafic envoyé depuis le pc pirate vers la cible, illustré dans la figure suivante :

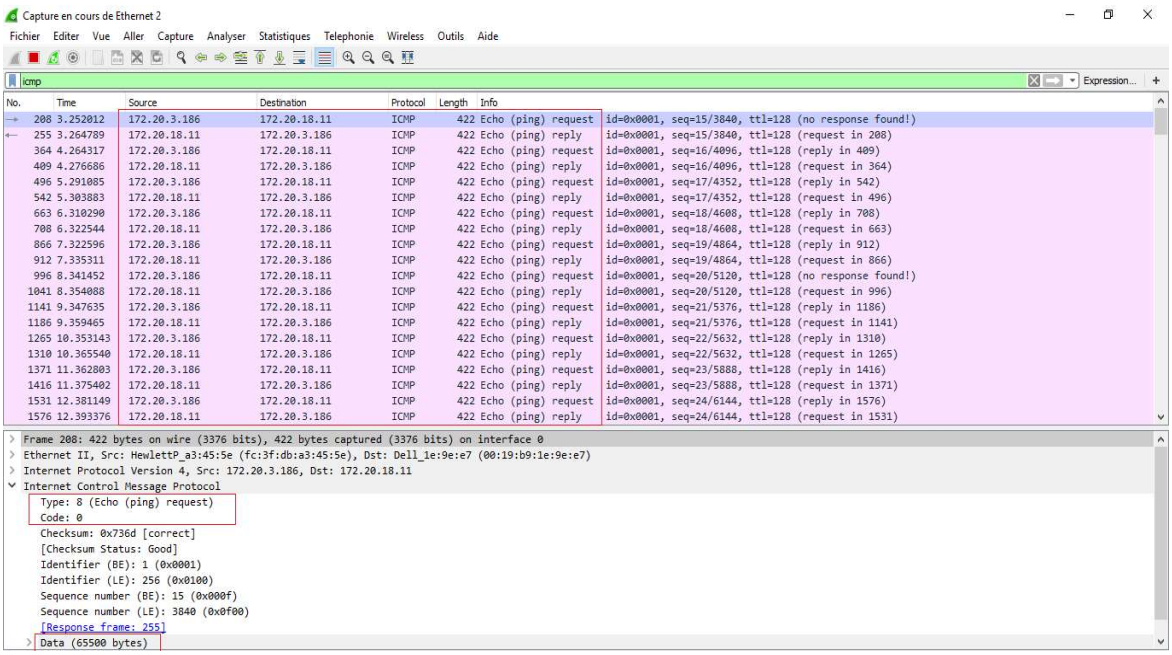

Figure 3.27 : La capture détaillée de ping of death

On a remarqué sur notre analyseur que nous avons effectué un ping of death (Ping mais mal forme) figure3.26, si on le compare avec le Ping normale (figure 3.24) on trouve qu'on a dépassé la taille autorisée (32bytes), plusieurs requêtes de longueurs de 65500 octets.

Et on note aussi que dans le cas de Ping normale il y'a au total 8 messages (4 messages echo reply et request) ICMP contrairement au Ping of death qui dépend de la taille des donnée envoyées (mais >8 messages).

D'après notre visualisation des paquets Ping normale celles des attaques on a pu définir la signature de cette attaque, illustrée dans le tableau suivant :

	Cas de Ping normal	Cas de l'attaque ping of death	Difference
La taille de paquet	32 octets	65500 octet	Ping of death superieur à celle de ping normal

Tableau 3.6 : La différence entre le ping normale et ping of death

3.3 Conclusion

Les trois méthodes implémentent le même mécanisme d'attaque. L'outil ouvre plusieurs connexions au serveur cible et envoie une suite continue de messages qui peuvent être définis à partir de l'option de paramètre de message TCP / UDP disponible sur l'outil. Dans les attaques TCP et UDP, la chaîne est envoyée en texte brut mais dans l'attaque HTTP, elle est incluse dans le contenu d'un message HTTP GET.

Cet outil continue d'envoyer des requêtes au serveur cible ; après un certain temps, le serveur cible devient surchargé. De cette façon, le serveur cible ne pourra plus répondre aux demandes des utilisateurs légitimes, ce qui le fermera efficacement.

Pour détecter ces attaques et mettre en place une bonne politique de sécurité, on propose dans le chapitre(4) nos propres règles de détection implémentéesau niveau du système de détection d'intrusion « Snort ».

Chapitre 4 Détection des attaques

4.1 Introduction

Nous avons vu précédemment qu'il était très simple de mettre en œuvre une attaque du type Déni de Service ou encore sans avoir de réelles connaissances techniques sur les protocoles réseau et en programmation puisque des outils prêts à l'emploi existent.Pour luttercontre ces types d'attaques il faut en premier les détecter, c'est pour cela on a consacré ce chapitre à la mise en œuvre de la détection d'attaques à base de l'IDSSnort.

4.2 L'utilisation de Snort

Le déroulement de Snort fait appel à la commande **cmd**, la figure suivant démontre l'opération :

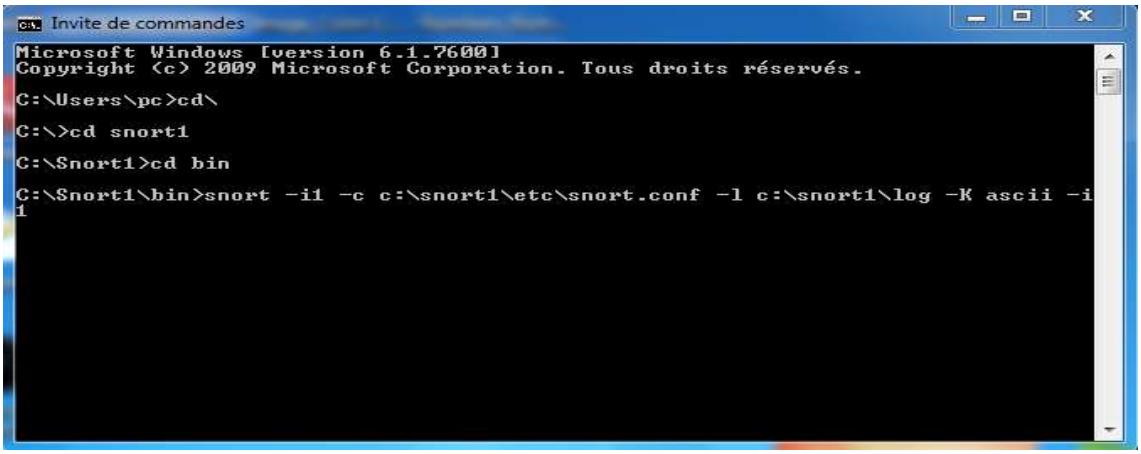

Figure 4.1 : Mise en marche de Snort

La commande « cd Snort1 » : fait appel à Snort qui est installé dans le disque local (c).

-i1 : c'est la carte réseau qui est activée.

C : \Snort1\log : permet d'enregistrer les alertes dans le fichier log.

4.3 Les règles de détection des attaques

On a créé des règles spécifiques pour lutter et détecter les différentes attaques (TCP, UDP, HTTP, Ping of death) à partir des signatures.

Nous avons remarqué qu'ils y a des paramètresqui influent sur l'attaque, à partir de ces paramètres en va créer ces règles.

4.3.1 Détection de l'attaque TCP

A l'aide de la signature de l'attaque TCP qu'on a réalisé dans le chapitre(3) on va créer notre règle de détection de l'attaque TCP :

```
Alert tcp any any -> any any (msg:"LOIC Dos attaque TCP mod"; flag:ap; threshold:
type threshold, track bay_src, count:100, second:3, sid:100000004; rev:1)
```

Alert : Une alerte « LOIC Dos attackTCP mod »sera générée si les critères sont réunis.

any any -> any any : de n'importe quel adresse IP source et port à n'importe quel adresse, port destination.

Msg : affiche le msg (LOIC Dos attack TCP) dans les alertes.

Flags : le paquet est caractérisé par PSH et ACK.

Threshold : indiquez les alertes de seuil (100 paquet/s).

Track bay_src : le taux est suivi soit par IP source, soit par l'adresse IP de destination. Les ports ou toute autre chose ne sont pas suivis.

Count :d'après les résultats de chapitre 3 on a choisi leseuil= 100 paquet/s,le count c'est le nombre de paquet qu'il ne faut pas dépasser.

Second :3 secondes c'est périodes sur laquelle le compte est accumulé.

Sid : 100000004 c'est l'identifiant de notre signature dans la base de signature.

Rev : 1 est utilisé pour identifier de manière unique les révisions des règles Snort. Permet remplacer les signatures par des mises à jour.

Après la création de la règle de TCP attaque dans le fichier local.ruls on lance l'attaque pour tester la règle :

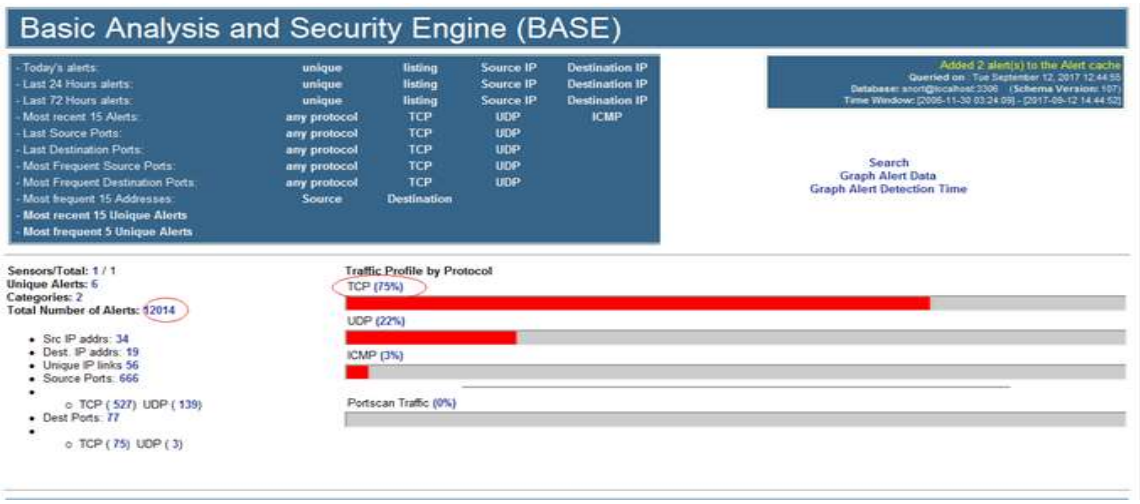

Figure 4.2 : Détection de l'attaque TCP

On remarque bien que Snort a détecté l'attaque TCP à partir d'augmentation du pourcentage de nombre des paquets TCP et des alertes.

Les informations plus détaillées sur l'attaque sont présentées sur la figure suivante :

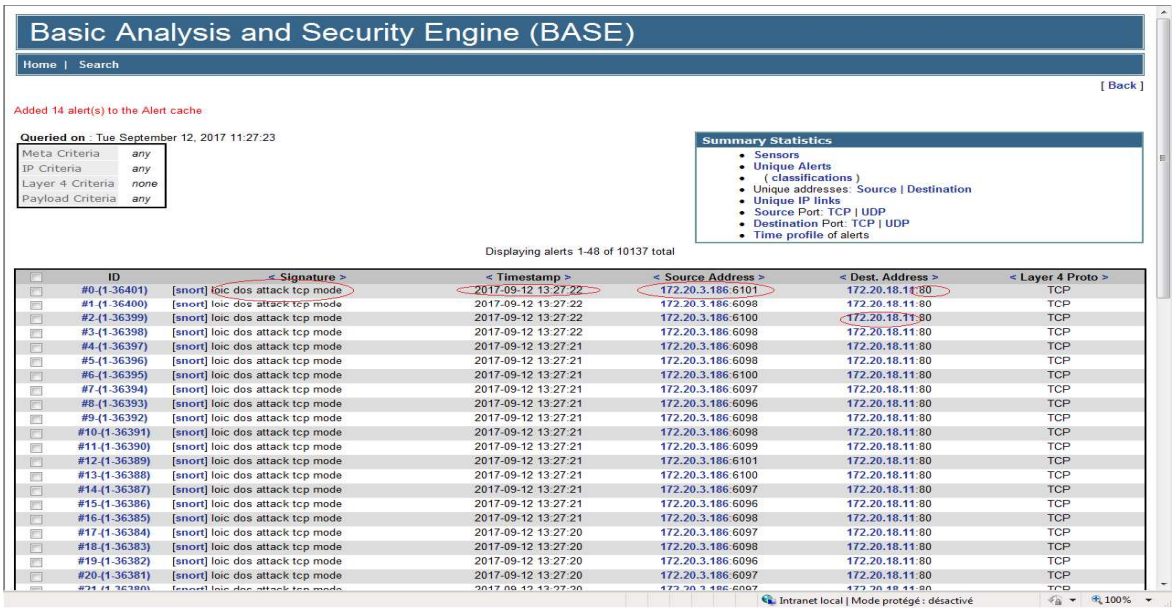

Figure 4.3 : Les informations détaillées de l'attaque TCP

Snort nous avise qu'il y'a une attaque de type TCP, l'outil utilisé par cette attaque (LOIC). Cette dernière utilise plusieurs ports vers une destination unique port 80, Snort nous informe aussi de la date d'attaque.

4.3.2 Détection de l'attaque UDP

C'est la même procédure qu'on a suivi pour créer la règle de l'attaque TCP c'est juste qu'ona changé la signature, on met celle d'attaqueUDP.

La règle qui définit l'attaque UDP est la suivante :

```
Alert udp any any -> any any (msg:"LOIC Dos attaque UDP mod"; threshold: type threshold, track
bay_src, count:300, second:3, sid:100000003; rev:1)
```

Alert : Une alerte « LOIC Dos attack UDP mod » sera générée si les critères sont réunis.

Msg : affiche le msg (LOIC Dos attack UDP mod) dans les alertes.

Threshold : indiquez les alertes de seuil (300 paquet/s).

Track bay_src : le taux est suivi soit par IP source, soit par l'adresse IP de destination. Les ports ou toute autre chose ne sont pas suivis.

Count : d'après les résultats de chapitre 3 on a choisi le seuil de limite= 300 paquet/s, le count c'est le nombre de paquet qu'il ne faut pas dépasser.

Second :3 secondes c'est périodes sur laquelle le compte est accumulé.

Sid : 100000003 c'est l'identifiant de notre signature dans la base de signature.

Après la création de la règle de UDP attaque dans le fichier local.ruls on lance l'attaque pour tester son efficacité :

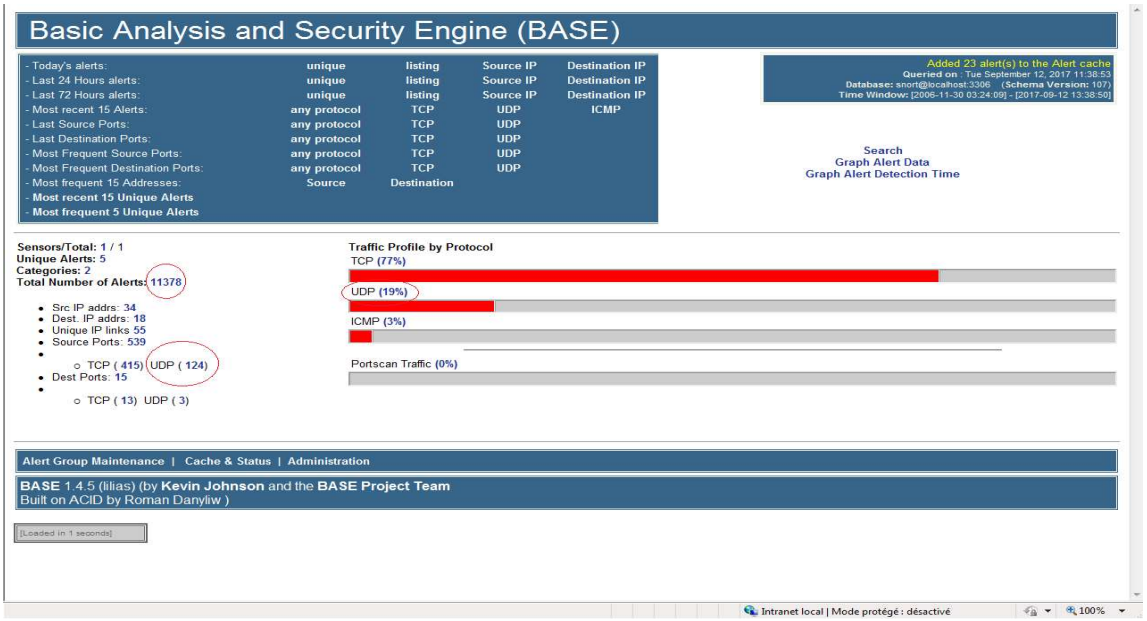

Figure 4.4 : La détection de l'attaque UDP

On remarque bien sur la figure 4.4 qu'il y a de l'argumentation au niveau des paquets UDP et les alertes, ce qui confirme que Snort a pu détecter l'attaque.

Les informations plus détaillées sur l'attaque UDP sont présentées sur la fig. suivante :

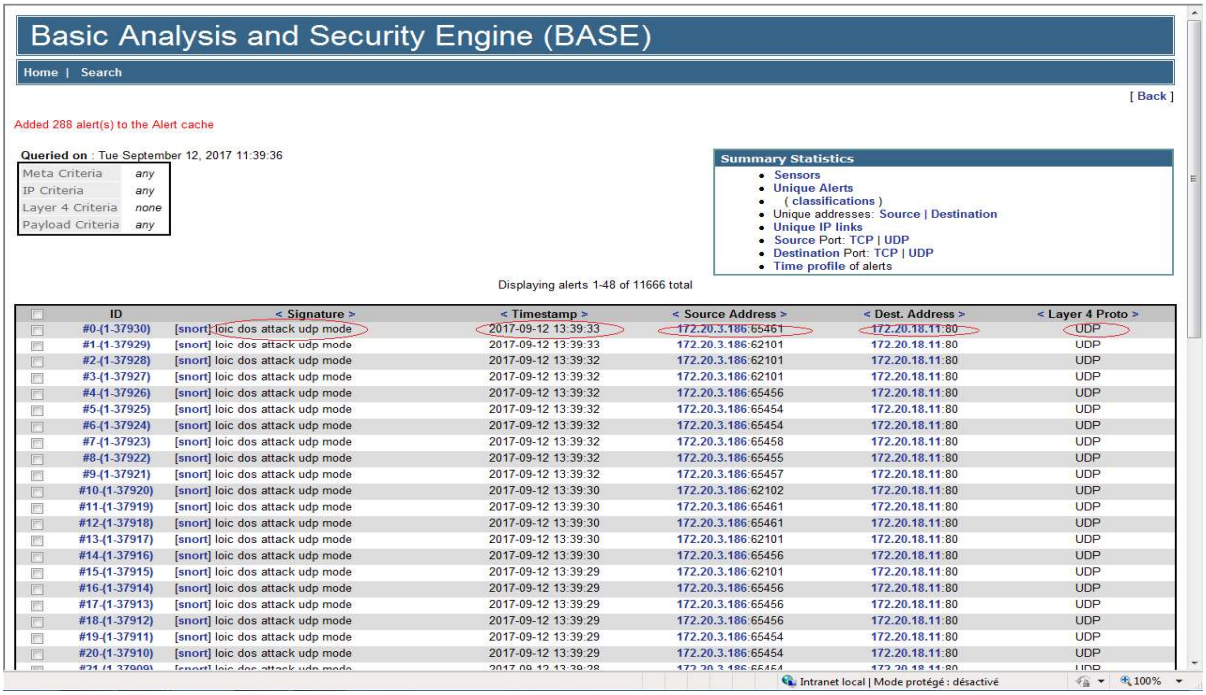

Figure 4.5 : La capture de la détection de l'attaque UDP

La signature nous signale qu'il y'a une attaque de type UDP, l'outil utilisé par cette attaque estLOIC. Cette dernière utilise plusieurs ports vers une destination unique port 80.

4.3.3 Détection de l'attaque HTTP

La règle qui définit l'attaque HTTP est la suivant :

```
Alert tcp any any -> any any (msg :"LOIC  Dos attack HTTP mod" flags :ap ; content |47 45 5452
2f 20 48 54 54 50 2f 31 2e 30 0d 0a 0d 0a 0d 0a| ; sid :100000006 ; rev :1 ;)
```

|47 45 54 52 2f 20 48 54 54 50 2f 31 2^{e} 30 0d 0a 0d 0a 0d 0a|: c'est le message HTTP (HTTP/1.0\ r\n) en hexadécimale.

Alert : Une alerte « LOIC Dos attack HTTP mod » sera générée si les critères sont réunis.

any any -> any any : de n'importe quel adresse IP source et port à n'importe quel adresse, port destination.

Msg : affiche le msg (LOIC Dos attack HTTP mod) dans les alertes.

Flags : le paquet est caractérisé par PSH et ACK (ap).

Threshold : indiquez les alertes de seuil (100 paquet/s).

Track bay_src : le taux est suivi soit par IP source, soit par l'adresse IP de destination. Les ports ou toute autre chose ne sont pas suivis.

Content : d'après les résultats de chapitre 3 on a choisi le content=(HTTP/1.0\ r\n), c'est la requête http get mal formé.

Second :3 secondes c'est périodes sur laquelle le compte est accumulé.

Sid : 100000006 c'est l'identifiant de notre signature dans la base de signature.

La figure suivante montre la détection de l'attaque HTTP :

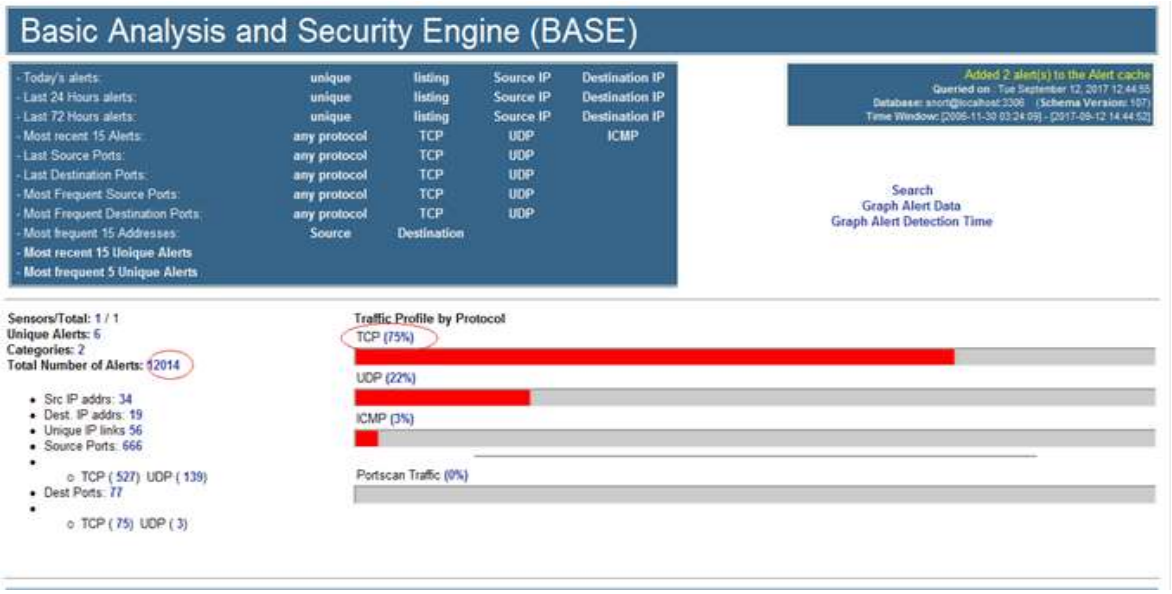

Figure 4.6 : La détection de l'attaque HTTP

On remarque l'augmentation de flux TCP et les alertes car les attaque HTTP utilisent le protocole TCP.

La figure suivante illustre les données détaillées de HTTP :

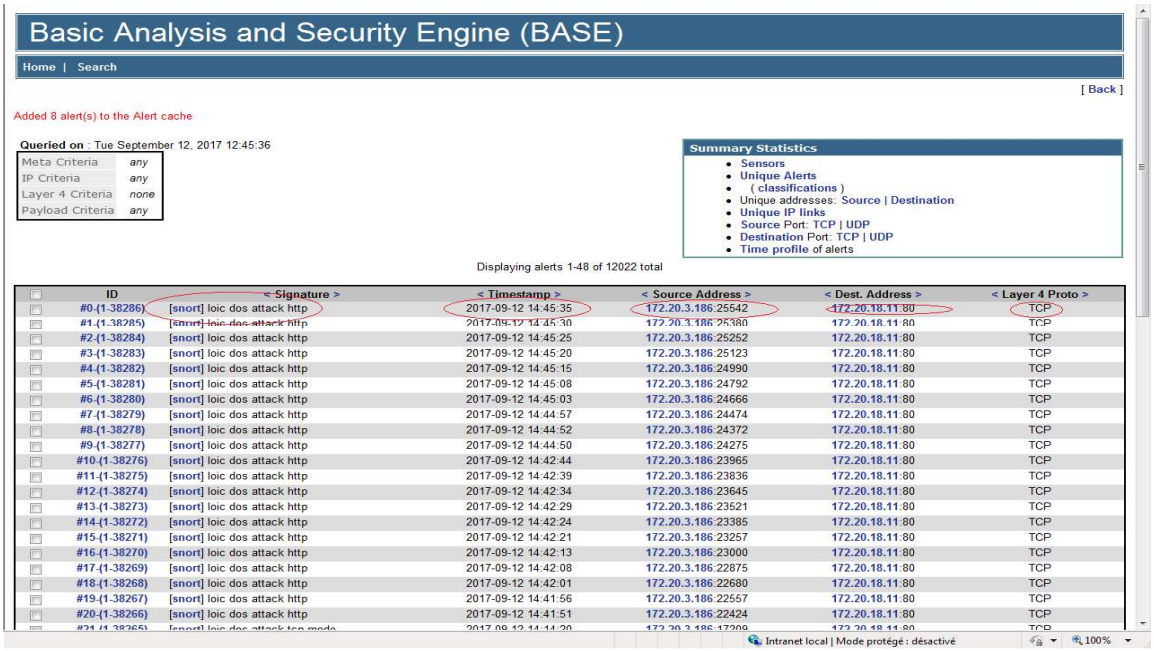

Figure 4.7 : Les informations détaillé de l'alerte d'attaque HTTP

On a obtenu les mêmes résultats que l'attaque TCP sauf qu'il y a un changement au niveau de signature.

4.3.4 Détection de l'attaque ping of death

La règle qui définit l'attaque UDP est la suivante :

```
Alert icmp any any ->any any (msg :"attack Ping of death"; disize>1000 ;itype :8 ; icode : 0 ;
sid :100000007 ; rev :1;)
```

Alert : Une alerte «attack Ping of death » sera générée si les critères sont réunis.

any any -> any any : de n'importe quel adresse IP source et port à n'importe quel adresse, port destination.

Type 8 et code 0 :demande de renvoi d'informations (Ping).

Itype : teste le champ type ICMP contre la valeur de Ping (max 1000 octets).

Icode : teste le champ code ICMP contre la valeur de Ping.

Disize :d'après les résultats de 3 chapitre, on prend la valeur maximum=1000 octets car le Ping (32 octets) ne dépasse pas cette valeur.

Msg : affiche le msg (attack Ping of death) dans les alertes.

Sid : 100000007 c'est l'identifiant de notre signature dans la base de signature.

La figure suivante montre la détection de l'attaque ping of death :

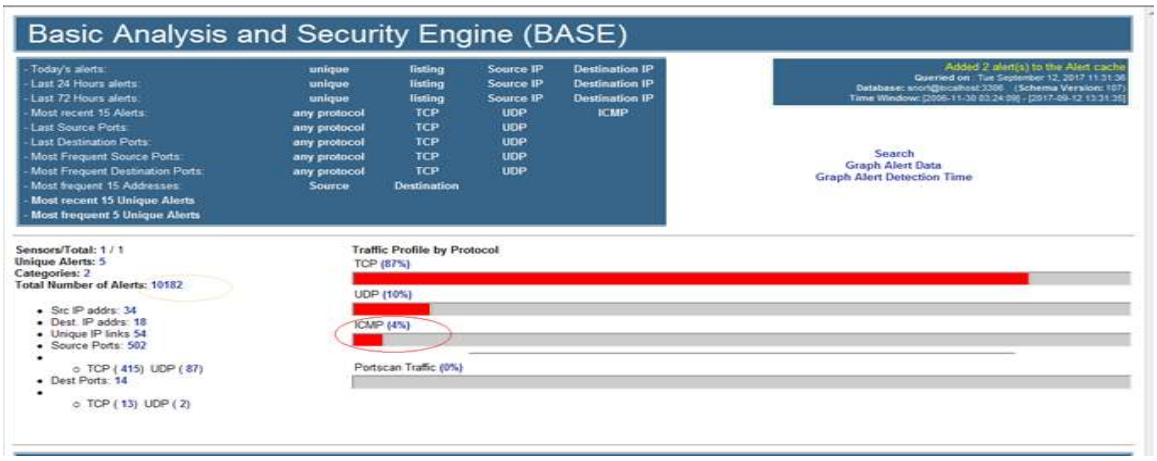

Figure 4.8 : Détection de l'attaque Ping of death

On remarque que le pourcentage ICMP est augmenté

La figure suivante illustre les données détaillées de Ping of death :

ID	< Signature >	< Timestamp >	< Source Address >	< Dest. Address >	< Layer 4 Proto >
#0-(1-36464)	[snort] ping of death detected	2017-09-12 13:31:53	172.20.3.186	172.20.18.11	ICMP
#1-(1-36463)	[snort] ping of death detected	2017-09-12 13:31:52	172.20.3.186	172.20.18.11	ICMP
#2-(1-36462)	[snort] ping of death detected	2017-09-12 13:31:51	172.20.3.186	172.20.18.11	ICMP
#3-(1-36461)	[snort] ping of death detected	2017-09-12 13:31:50	172.20.3.186	172.20.18.11	ICMP
#4-(1-36460)	[snort] ping of death detected	2017-09-12 13:31:49	172.20.3.186	172.20.18.11	ICMP
#5-(1-36459)	[snort] ping of death detected	2017-09-12 13:31:48	172.20.3.186	172.20.18.11	ICMP
#6-(1-36458)	[snort] ping of death detected	2017-09-12 13:31:47	172.20.3.186	172.20.18.11	ICMP
#7-(1-36457)	[snort] ping of death detected	2017-09-12 13:31:46	172.20.3.186	172.20.18.11	ICMP
#8-(1-36456)	[snort] ping of death detected	2017-09-12 13:31:45	172.20.3.186	172.20.18.11	ICMP
#9-(1-36455)	[snort] ping of death detected	2017-09-12 13:31:44	172.20.3.186	172.20.18.11	ICMP
#10-(1-36454)	[snort] ping of death detected	2017-09-12 13:31:43	172.20.3.186	172.20.18.11	ICMP
#11-(1-36453)	[snort] ping of death detected	2017-09-12 13:31:42	172.20.3.186	172.20.18.11	ICMP
#12-(1-36452)	[snort] ping of death detected	2017-09-12 13:31:41	172.20.3.186	172.20.18.11	ICMP
#13-(1-36451)	[snort] ping of death detected	2017-09-12 13:31:40	172.20.3.186	172.20.18.11	ICMP
#14-(1-36450)	[snort] ping of death detected	2017-09-12 13:31:39	172.20.3.186	172.20.18.11	ICMP
#15-(1-36449)	[snort] ping of death detected	2017-09-12 13:31:38	172.20.3.186	172.20.18.11	ICMP
#16-(1-36448)	[snort] ping of death detected	2017-09-12 13:31:37	172.20.3.186	172.20.18.11	ICMP
#17-(1-36447)	[snort] ping of death detected	2017-09-12 13:31:36	172.20.3.186	172.20.18.11	ICMP
#18-(1-36446)	[snort] ping of death detected	2017-09-12 13:31:35	172.20.3.186	172.20.18.11	ICMP
#19-(1-36445)	[snort] ping of death detected	2017-09-12 13:31:34	172.20.3.186	172.20.18.11	ICMP
#20-(1-36444)	[snort] ping of death detected	2017-09-12 13:31:33	172.20.3.186	172.20.18.11	ICMP

Figure 4.9 : La capture détaillée de la détection de ping of death

Le message d'attaque Ping of death montre clairement qu'il y'a une attaque de type ICMP c'est le Ping of death, l'outil utilisé par cette attaque est LOIC. Snort nous informe aussi de la date d'application d'attaque.

4.4 Conclusion

On utilisant nos différentes règles de détection qu'on a créée nous avons pu obtenir des détections des quatre attaques (TCP, UDP, HTTP, Ping of death) mentionnée dans Snort grâce à son système BASE. Et à l'aide des différentes signatures tirer durant notre simulation d'attaques.

Conclusion générale

Les attaques de déni de service (attaque TCP, UDP, HTTP, Ping of death) ne sont pas quelque chose dont une entreprise veut être victime, cela pourrait coûter cher. Toutes les attaques ci-dessus peuvent être mises en œuvre par une personne qui n'a pas beaucoup de compétence et peut causer beaucoup de dégâts.

Le système de détection d'intrusion (IDS) est l'un des éléments mise en place dans le cadre de la politique globale de sécurité définie, En effet il est très utile et très efficace.

L'objectif de notre projet consisté a créé des règles de détection des attaquesà base de Snort :

- ✓ Simulation des attaques.
- ✓ Reconnaissances des signatures à l'aide d'analyseur de paquet Wireshak.
- ✓ Création des règles de détection au niveau de Snort.

Afin de valider notre application et vérifier son bon fonctionnement, nous avons effectué une série de tests tel que les attaques de type déni de service (Ping of death, TCP, UDP, HTTP). Nous avons réussi à rendre ces attaques visibles et détectableà l'aide de notre analyseur de paquets Wireshark, ce denier nous as aidé a tiré les différents signatures d'attaques, qu'on a développé pour créer nos règles de détection des attaques à base de Snort.

L'objectif que nous nous sommes fixé a été atteint et ce modeste travail nous a permet de :

- ✓ Découvrir la sécurité informatique.

- ✓ Améliorer nos connaissances en programmation.
- ✓ Utiliser de nouveaux outils.

Finalement, cette application pourra être améliorée en lui ajoutant d'autres fonctionnalités tel qu'un système de prévention d'intrusion, un antivirus,… etc.

Mais le risque zéro ne sera jamais atteint et plus les technologies évolues, plus les pirates auront accès a des ressources importantes. Les attaques seront alors très compliquées, la course contre les pirates ne s'arrêtera jamais.

Annexes : Installation de l'IDSSnort

Introduction

Ce document va tenter d'expliquer les différentes étapes pour mettre en place le détecteur d'intrusions SNORT à partir des sources. Un détecteur d'intrusions s'appelle aussi "IDS" pour Intrusion Detection System. SNORT est un système de détection d'intrusions réseau en Open Source, capable d'effectuer l'analyse du trafic en temps réel. On l'utilise en général pour détecter une variété d'attaques et de scans tels que des débordements de tampons, des scans de ports furtifs, des attaques CGI, des scans SMB, des tentatives d'identification d'OS, et bien plus.

Avant de commencer l'installation de SNORT, vous devez avoir installé :

PACKAGES REMARQUES

MySQL La base de données MySQL

MySQL−client La partie cliente de MySQL (connexion BD)

PHP MySQL le module PHP de MySQL

Apache Le serveur web Apache

mod_php Le module PHP pour Apache

libpcap/libpcap0−devel Librairie utilisée par SNORT pour capturer les paquets (rpm téléchargeable sur rpmfind.net)

gcc indispensable pour compiler les sources de SNORT

Si vous n'avez pas encore installé le trio Apache/PHP/MySQL, il y a un article sur Lea vous expliquant comment le faire.

Les étapes pour l'installation de SNORT sont les suivantes :

• Installation de l'outil SNORT

• Installation des règles SNORT

• Liaison MySQL et SNORT

• Mise en place d'ACID (Interface PHP pour visualiser les logs SNORT)

Installation de SNORT

Téléchargez le dernier release de SNORT à l'adresse suivante : HTTP://www.SNORT.org/dl. La compilation de ce programme reste traditionnelle :

COMMANDES REMARQUES

Cd /Usr/local/Snort ...

Tar −xvzf SNORT−1.9.*.tar.gz Décompacte l'application

. /configure −−with−mysql=/usr/lib/MySQL Retirez l'argument −−with−mysql si vous ne souhaitez pas rediriger les logs SNORT vers

Une base de données mysql *

make Compilation

make install Installation

Pour l'argument −−with−mysql, vous pouvez l'adapter si vous utilisez une base de données autre que MySQL :

• −−with−odbc=$PATH_ODBC : pour une base de données Microsoft SQL server

• −−with−postgresql=$PATH_POSTGRE : pour une base PostegreSQL

• −−with−oracle=$ORACLE_HOME : pour une base de données Oracle.

Installation des règles SNORT

Maintenant, il faut télécharger les règles de SNORT. En effet, SNORT utilise des règles pour détecter les intrusions. Il existe aujourd'hui environ 1200 règles différentes. Ces règles se caractérisent par un ensemble de fichiers (ftp.rules, p2p.rules,telnet.rules etc...). Vous devez téléchargez les sourcesde ces règles à l'adresse suivante :

HTTP://www.SNORT.org/dl/signatures

Créez le répertoire de configuration SNORT, et installez−y les règles :

COMMANDES REMARQUES mkdir /etc/Snort Création du répertoire contenant la configuration SNORTcp /usr/local/Snort*/etc/Snort.conf /etc/Snort Copie du fichier de config Snort dans /etc/Snortcp Snortrules.tar.gz /etc/Snort Mise en place des règles dans le répertoire de configuration SNORTcd /etc/Snort On se place dans le répertoire de configuration SNORTtar −xvzf Snortrules.tar.gz Décompactage des règles

Les règles SNORT sont alors placées dans le répertoire /etc/Snort/rules.

Installation de l'IDSSNORT

1

Maintenant, Il faut éditer le fichier de configuration Snort (/etc/Snort/Snort.conf) et spécifier le réseau sur lequel l'IDS travaille. Il faut pour cela

Modifier la variable HOME_NET :

Var HOME_NET [10.1.1.0/24] # SNORT travaille sur le réseau 10.1.1.0

Var HOME_NET (10.1.1.0/24,192.168.1.0/24] # Si votre carte réseau possède 2 alias

Dans le fichier de configuration de SNORT (/etc/Snort/Snort.conf), vous avez toute une série d'include. Il s'agit des règles utilisées par

SNORT pour détecter d'éventuelles intrusions. Il y a des règles de Telnet, ICMP, FTP, ... Bref, commentez celles que vous ne voulez pas et

Décommettez celles qui vous paraissent utile. Conseil : Décommettez les règles ICMP, car elles ne cessent pas de vous remonter des alarmes très

Souvent inutiles.

Pour des explications plus détaillées concernant les règles SNORT, allez voir ici.

Lancement de SNORTDeux possibilités s'offrent à nous. Soit vous lancez SNORT tout seul, et dans ce cas, il génèrera ces logs dans un fichier plat. Soit vous décidez

deL’interface avec une base de données. Suivant le cas, SNORT ne se lancera pas de la même façon.

Sans MySQL :

/Usr/local/Snort*/src/Snort −c /etc/Snort/Snort.conf −i eth0 −D

Avec MySQL :

/Usr/local/Snort*/src/Snort −c /etc/Snort/Snort.conf

Remarque : Si vous souhaitez interfacer SNORT avec une base de données, ne lancez pas SNORT avec l'argument −L qui spécifie l'emplacement des logs.

Lier les logs SNORT avec MySQL

Maintenant, nous allons éditer le fichier de configuration de SNORT afin de lui indiquer qu'il faut rediriger les logs dans une base de données (ici

MySQL). Avec vos yeux de lynx, retrouvez la ligne suivante dans le fichier de configuration SNORT /etc/Snort/Snort.conf :

#output database:log,mysql,user=root password=test dbname=SNORT host=localhost

Décommentez et modifiez cette ligne par :

output database:log,mysql,user=user_Snort password=Snort_pwd dbname=Snort host=localhost

Ici, l'utilisateur MySQL accédant à la base de données s'appelle "user_Snort", son password associé est "Snort_pwd", le nom de la base MySQL utilisée

ParSnort est "Snort" et la machine qui fait tourner la base Mysql est la même que celle où SNORT tourne.

Création de la base de données SNORT

Au préalable, assurez−vous d'avoir installé :

PACKAGES REMARQUES

MySQL−client−* partie cliente de MySQL

MySQL−devel−*

Astuce : La commande "rpm −qa | grep client" vous permet de vérifier que votre station Linux possède bien ces packages installés.

Suivez alors les instructions suivantes :

COMMANDES REMARQUES

cd /usr/local/Snort*/contrib on se place à l'endroit du fichier contenant les tables

SQL de SNORT

mysql −u root −p Connexion à la base de données en tant qu'administrateur (au passage, si ce n'est pas encore fait, définissez un password pour l'administrateur de la base par la commande 'set password for root@localhost=PASSWORD('totomdp'); create database SNORT; Création de la base de données SNORTuse mysql.On se place ici pour créer l'utilisateur MySQL qui gérera la base de données Snortinsert into user values('localhost', 'user_Snort', password('Snort_pwd'), 'Y', 'Y', 'Y', 'Y', 'Y', 'Y', 'Y', 'Y', 'Y',

'Y', 'Y', 'Y', 'Y', 'Y', 'Y', 'Y', 'Y', 'Y', 'Y', 'Y', 'Y', 'Y', '', '', '', '', 'Y', 'Y', 'Y') ;

Création utilisateur MySQL "user_Snort". Attention le nombre de 'Y' dépend de votre version de MySQL. (Faitesun select * from user ; pour voir combien il faut en mettre) Lancement de SNORT

2

grant ALL PRIVILEGES ON SNORT.* TO user_Snort@localhost IDENTIFIED

BY 'Snort_pwd' WITH GRANT OPTION;

Attribution des droits de la base "Snort" à l'utilisateur "user_Snort" flush privileges; Recharge les tables de droits pour prendre en compte les nouvelles modifications use Snort; on se place dans la base où l'on veut créer les tables pour SNORTSource create_mysql Création des tables pour SNORT

Vérifiez que les tables sont bien créées. Allez voir dans /var/lib/mysql/Snort et vous y verrez tout un tas de fichiers correspondant au nom des tables de la base de données SNORT (il doit y avoir 3 fichiers par tables).

Lancez SNORT. Désormais, SNORT envoie les informations dans la base de données (astuce : installez PhpMyAdmin, et vérifiez la taille de la base de données SNORT. Si tout fonctionne, vous la voyez augmenter si bien évidemment il y a du trafic !).

Installation/Configuration ACID

ACID est une interface PHP qui permet de visualiser les remontées d'alarmes générées par SNORT. Cette partie sous−entend que vous avez une base de données qui récupère les informations envoyées par SNORT. Avant de suivre l'installation de cette application, assurez−vous d'avoir

Téléchargé :

Adodb : Contient des scripts PHP génériques de gestion de bases de données. L'installer dans la racine d'apache (/var/www/html/adodb par exemple)

• PHPlot : librairie de scripts PHP utilisée par ACID pour présenter graphiquement certaines données statistiques (optionnel)

Le téléchargement d'ACID se fait ici. Imaginons que la racine de votre serveur web est /var/www/html. Installez ACID dans la racine d'apache :

COMMANDES REMARQUES

Cd /var/www/html Placez−vous dans la racine du serveur web

Tar −xvzf acid* Décompactage de ACID

Tar −xvzf adodb* Décompactage de AdoDB

Tar −xvzf phplot* Décompactage de PHPlot

vi /var/www/html/acid/acid_conf.php Renseignez les champs suivants :

♦ $DBlib_path="../adodb";

♦ $Chartlin_path="../phplot";

♦ alert_dbname="Snort"

♦ alert_host="localhost"

♦ alert_user="user_Snort"

♦ alert_password="Snort_pwd"

Voilà, maintenant vous pouvez vérifier que ACID est bien configuré (allez voir sur HTTP://localhost/acid). Si vous le souhaitez, L'accès peut se faire via

certificat SSL de manière à crypter l'échange entre vous et le détecteur d'intrusions.

Sachez que ce document a pour but de vous apporter quelques éléments de réponse concernant l'installation et la configuration de l'IDSSNORT. Il est

loin d'être parfait. Vos remarques sont les bienvenues. Je prévois de modifier le présent document suivant les remarques que vous y apporterez.

Bibliographie

[1] Maxime MORGE : ' Initiation aux réseaux informatiques ', Ecole nationale supérieur des mines, SAIN-ETIENNE, 2003.

[2] William PUECH : 'Classification des réseaux', Université de Nîmes ,2012.

[3] Mohammed OUMSIS : ' Réseaux informatiques ', Université Sidi Mohamed Ben Abdallah Faculté des sciences Dhar El Mahrez Fés, 2007/2008.

[4] Philippe Latu : ' Technologie Ethernet ', Inetdoc, 2000.

[5] Bernard COUSIN : ' Le réseau local : Token Ring ', Université Rennes I, 1998.

[6] Maxime MORGE : ' Les supports de transmission', Ecole nationale supérieur des mines SAIN-ETIENNE, 2003. 1998.

[7] Patrick LALLEMENT : ' Les grandes fonctions des réseaux ', Les filières technologiques des enseignements supérieurs, Edition ellipses, 2012.

[8] Pascal Nicolas : ' cour des réseaux maitrise d'informatique ', Université d'ANGERS, 1999.

[9] ACISSI :' sécurité informatique-Ethical hacking,Apprendre l'attaque pour mieux se défendre',Edition Eni,2011.

[10] Jean-Francois pillou : ' Tout sur la sécurité informatique', Edition DUNOD, 2005.

[11] : Mehdi Merouane, 'introduction aux attaques informatiques', université Blida1 département d'électronique, 2015/2016.

[12] : Roulot, ' le piratage de A à Z ', Edition Edigo, 2010.

[13] : Xavier Tannier, ‘ Se protéger sur internet, Conseil pour la vie en ligne ‘, Edition Eeyrolles, 2010.

[14] : Mohamed ROMDHANI, ‘Modules Sécurité de Linux et des Services WEB‘, Université de la Manouba, Ecole Nationale des Sciences de l’Informatique, Mastère Spécialisé en Sécurité Informatique 2004/2005.

[15] Cisco Systems, Inc : ' Commande Ping et traceroute ', Cisco Networking Academy 1992–2007.

[16] : Luca Allodi, ‘Denial of Service Attack and Defense ‘, UNIVERSITAS ATHESINA STUDIORUM, 2016.

[17] : Network Startup Resource Center, ‘Surveillance du réseau et de gestion‘, Définition et analyse desPerformances du réseau, 2010.

[18] :Danes Adrien, ‘La protection des réseaux contre les attaques DoS ‘, Université paris Descartes, 2009.