REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

Ministère de l'enseignement supérieur et de la recherche scientifique

Université 20 Août 1955 Skikda

Faculté des sciences-Département de l'informatique

Mémoire de fin d'études en vue l'obtention du diplôme Master : Réseaux et Systèmes Distribués (RSD/SI)

Thème :

Etude et mise en œuvre D'un honeypot

Réalisé par :

- ✓ Youcef Khodja Ahmed Iheb
- ✓ Saghir Abdeldjalil

Encadré par :

- Touil Ghassen

Année Universitaire : 2022 / 2023

Remerciement

Nous exprimons tout d'abord notre gratitude envers Dieu de nous avoir accordé la force morale et physique et de nous avoir permis de mener à bien ce travail.

Nous souhaitons adresser nos remerciements les plus sincères au professeur Touil Ghassen, notre superviseur, qui a guidé nos réflexions et a accepté de répondre à nos questions avec patience, disponibilité et précieux conseils, ainsi que pour sa critique constructive.

Nous sommes reconnaissants envers nos familles pour leur soutien et leurs encouragements tout au long de ce parcours.

Nous tenons également à remercier les membres du jury qui ont accepté de participer et d'évaluer ce travail.

Merci.

Résumé :

Aujourd'hui, assurer la sécurité des réseaux représente un grand défi car les menaces évoluent constamment. Les protections traditionnelles comme les pare-feu ne suffisent plus à détecter les nouvelles formes d'attaques. C'est pourquoi les honeypots ont été développés. Ce sont des systèmes qui attirent les attaquants en leur faisant perdre leur temps sur de faux systèmes. Ils permettent également d'étudier les techniques des intrus.

Ce mémoire présente la technologie des honeypots, qui simulent les services SSH pour interagir avec les pirates. L'objectif est d'épuiser les ressources de l'attaquant, d'en apprendre plus sur ses méthodes et de le distraire des vrais systèmes. Nous utilisons un serveur central, et le programme Virtual Box avec des machines virtuelles pour simuler des attaques sur des serveurs SSH (honeypots "Kippo"). Nous avons réalisé une attaque par force brute sur le protocole SSH à l'aide de l'outil Nmap dans le système Linux-Kali.

Cette approche offre une solution solide et proactive pour lutter contre les nouvelles menaces qui émergent constamment dans la sécurité des réseaux.

ملخص :

اليوم، ضمان أمان الشبكات يمثل تحدياً كبيراً نظراً لتطور التهديدات بشكل مستمر. الحمايات التقليدية مثل جدران الحماية لا تكفي لاكتشاف أشكال الهجمات الجديدة. ولهذا تم تطوير الهونيبوتس .(Honeypots) إنها أنظمة مصممة خصيصاً لجذب المهاجمين وإضاعة وقتهم على أنظمة وهمية. كما توفر بيئة مناسبة لدراسة تقنيات المتطفلين.

تقدم هذه المذكره تقنية الهونيبوتس، التي تحاكي خدمات SSH للتفاعل مع المخترقين. الهدف هو استنزاف موارد المهاجم، واكتساب معلومات أكثر عن أساليبه وتحويل انتباهه عن الأنظمة الحقيقية. نحن نستخدم برنامج Box Virtual بما فيه من أجهزة افتراضية لمحاكاة الهجمات على خوادم SSH هونيبوتس Kippo. قمنا بتنفيذ هجوم عن طريق القوة الغاشمة على بروتوكول SSH باستخدام أداة Nmap في نظام Linux-Kali.

تقدم هذه النهج حلاً قوياً واستباقياً لمكافحة التهديدات الجديدة التي تظهر بشكل مستمر في مجال أمان الشبكات.

Abstract:

Today, ensuring network security is a major challenge due to the constantly evolving threats. Traditional protections like firewalls are no longer sufficient to detect new forms of attacks. That's why honeypots have been developed. They are systems specifically designed to attract attackers and waste their time on fake systems. They also provide an environment for studying the techniques used by intruders.

This thesis presents honeypot technology, which simulates SSH services to interact with hackers. The objective is to deplete the attacker's resources, learn more about their methods, and distract them from real systems. We use Honey Modern Network, a central server, and the Box Virtual program with virtual machines to simulate attacks on SSH servers (Kippo honeypots). We conducted a brute-force attack on the SSH protocol using the Nmap tool in the Linux-Kali system.

This approach offers a robust and proactive solution to combat the constantly emerging threats in network security.

Table des matières

Chapitre 01 : La Sécurité Informatique

Chapitre 02 : Les honeypots

Chapitre 03 : Présentation du honeypot « Kippo »

Chapitre 04 : Simulation

Liste des figures

Liste des tableaux

Liste des Abréviations

ACL: Access Control List.

ARP: Address Resolution Protocol.

CD-ROM: Compact Disk – Read Only Memory.

DDOS: Distributed Denial of Service.

DMZ: Demilitarized Zone.

DNS: Domain Name System.

DTK: Deception Toolkit.

FTP: File Transfer Protocol.

HTTP: Hyper Text Transfer Protocol.

IP: Internet Protocol.

IPS: Intrusion Prevention System.

MHN: Modern Honey network.

MITM: Man In The Middle.

NMAP: Network Mapper.

OS: Operating System.

SNMP: Simple Network Management Protocol.

SQL: Structured Query Language.

SSH: Secure Shell.

TCP: Transmission Control Protocol.

TFN: Tribal Flood Network.

UDP: User Datagram Protocol.

UML: User Mode Linux.

Introduction générale

Introduction générale

Chaque jour, notre dépendance aux outils informatiques augmente et les entreprises ouvrent de plus en plus leurs systèmes informatiques à leurs partenaires et fournisseurs. Il est donc essentiel de protéger et de contrôler l'accès des utilisateurs et leurs droits pour prévenir les menaces. Avec la croissance continue de l'utilisation des réseaux et des systèmes informatiques, ainsi que la sensibilité des données manipulées, il est crucial de mettre en place des mécanismes de sécurité.

Dans ce contexte, nous avons choisi le sujet de notre thèse : "Étude et mise en œuvre d'un honeypot". Contrairement aux outils de sécurité traditionnels, les honeypots adoptent une approche non-défensive en attirant les attaquants à explorer un système informatique et en analysant leur comportement détaillé tout en les trompant.

Nous poserons les questions suivantes :

- Qu'est-ce que la sécurité informatique et quels sont ses critères ?

- Qu'est-ce qu'un honeypot ? Quelle est son importance et ses objectifs ?

- Quels sont les types de honeypots ?

Notre mémoire est organisé de la manière suivante :

- Le premier chapitre traite de la sécurité informatique, des attaques actuelles contre les systèmes d'information et des solutions de détection.

- Le deuxième chapitre présente les honeypots de manière générale, en abordant leurs objectifs, leur classification, leurs types et leur architecture.

- Le troisième chapitre se concentre sur le déploiement du honeypot Kippo dans notre réseau.

- Le dernier chapitre concerne la simulation d'une attaque sur le honeypot Kippo.

Chapitre 01

1. Introduction :

L'internet a radicalement changer le cours de la vie humaine, Malheureusement, ce vaste réseau et les technologies qui lui sont associées ont également entraîné dans leur sillage le nombre croissant de menaces pour la sécurité. Le moyen le plus efficace de se protéger contre ces menaces et ces attaques est de connaître les pratiques courantes en matière de cybersécurité.

Dans ce chapitre, nous allons présenter quelques notions reliées à la sécurité informatique et les différents types d'attaques.

2. La sécurité informatique :

La sécurité informatique est une discipline qui vise à protéger les systèmes informatiques, les réseaux, les données et les utilisateurs contre les menaces informatiques telles que les virus, les logiciels malveillants, les pirates informatiques et les attaques en ligne. La sécurité informatique est devenue de plus en plus importante à mesure que la technologie a évolué et que les ordinateurs et les réseaux sont devenus omniprésents dans notre vie quotidienne.

Les organisations et les entreprises sont particulièrement vulnérables aux attaques informatiques, car elles stockent souvent des informations confidentielles et sensibles sur leurs clients et leurs activités. La sécurité informatique comprend des pratiques telles que l'identification et la prévention des vulnérabilités, la gestion des accès, la gestion des identités et la surveillance des activités suspectes. Les professionnels de la sécurité informatique sont chargés de protéger les systèmes et les données contre les menaces, et de prévenir les pertes et les dommages causés par les attaques informatiques. [01]

3. Les critères de sécurité :

Les critères de sécurité sont des exigences ou des mesures visant à protéger les systèmes, les données, les personnes et les biens contre les menaces et les risques de sécurité. Les critères de sécurité sont utilisés dans de nombreux domaines, tels que la sécurité informatique, la sécurité physique, la sécurité des produits et des services, la sécurité financière…etc.

Voici quelques exemples de critères de sécurité :

1) **Confidentialité :**

La confidentialité est l'un des critères de sécurité les plus importants. Elle garantit que les informations ne sont accessibles qu'aux personnes autorisées.

2) **Intégrité :**

l'intégrité garantit que les informations sont exactes, complètes et fiables. Les données doivent être protégées contre toute modification ou altération non autorisée.

3) **Disponibilité :**

la disponibilité garantit que les systèmes, les données et les services sont accessibles aux personnes autorisées en tout temps. Les interruptions de service doivent être minimisées et les temps de récupération doivent être rapides. [03]

4) **Authenticité :**

l'authenticité garantit que les informations sont véridiques et fiables. Les personnes et les processus doivent être vérifiés et autorisés avant d'accéder aux informations.

5) **Traçabilité :**

la traçabilité garantit que toutes les actions effectuées sur les systèmes et les données sont enregistrées et suivies. Cela permet de détecter les activités suspectes et de suivre les responsabilités.

6) **Non-répudiation :**

la non-répudiation garantit que les personnes ne peuvent pas nier leur participation à une action ou une transaction. Les preuves doivent être fiables et incontestables.

7) **Résilience :**

la résilience garantit que les systèmes et les services peuvent récupérer rapidement des perturbations et des pannes. Les plans de continuité doivent être mis en place pour minimiser les perturbations.

8) **Sécurité physique :**

La sécurité physique garantit que les bâtiments, les locaux, les équipements et les personnes sont protégés contre les menaces physiques, telles que les intrusions, les vols et les actes de violence.

Ces critères de sécurité sont souvent combinés pour créer des politiques de sécurité complètes qui protègent les systèmes, les données et les personnes contre les menaces et les risques de sécurité [02].

4. Les menaces et les attaques informatiques :

4.1. Qu'est qu'un Hacker ?

Un hacker est un individu qui utilise ses compétences en informatique pour trouver des vulnérabilités dans les systèmes informatiques et les réseaux afin de les exploiter ou de les protéger. Le terme « hacker » est souvent utilisé pour décrire les personnes qui cherchent à pénétrer dans des systèmes informatiques sans autorisation, mais il peut également être utilisé pour décrire les professionnels de la sécurité informatique et les experts en cybersécurité. [03]

Les hackers peuvent être classés en différentes catégories en fonction de leurs intentions et de leurs méthodes d'attaque. Les hackers éthiques, également connus sous le nom de « chapeaux blancs », sont des experts en sécurité informatique qui cherchent à protéger les systèmes informatiques en trouvant et en signalant les vulnérabilités aux propriétaires des systèmes. Les hackers criminels, également appelés « chapeaux noirs », sont des individus qui cherchent à exploiter les vulnérabilités pour commettre des actes de piratage malveillants, voler des données, extorquer de l'argent ou causer des dommages. [04]

Il existe également des hackers « gris », qui se situent entre les hackers éthiques et les hackers criminels. Ils peuvent utiliser leurs compétences pour des activités légales, mais aussi pour des activités illégales, selon les circonstances et les opportunités. [04]

Il est important de noter que tous les hackers ne sont pas des criminels et que les compétences en informatique peuvent être utilisées à des fins positives pour améliorer la sécurité et la protection des systèmes informatiques. [04]

4.1.2. Les différents types de pirates :

Il existe différents types de pirates informatiques, classés en fonction de leur motivation, de leurs compétences, de leurs méthodes d'attaque et de leurs objectifs. Voici une classification générale des types de pirates informatiques :

1) Les pirates informatiques criminels :

Également appelés « chapeaux noirs », ils sont des individus ou des groupes qui utilisent des compétences en informatique pour mener des activités criminelles, telles que le vol de données, le chantage, la fraude, l'extorsion ou le sabotage. Leurs motivations sont souvent financières.

2) Les hackers éthiques :

Également connus sous le nom de « chapeaux blancs », ils sont des professionnels de la sécurité informatique qui cherchent à identifier les vulnérabilités dans les systèmes informatiques pour aider à améliorer la sécurité des entreprises et des organisations.

3) Les activistes :

Également appelés « hacktivistes », ils utilisent leurs compétences en informatique pour promouvoir une cause ou une idéologie politique. Ils peuvent mener des attaques DdoS (Distributed Denial of Service) ou des opérations de fuite de données.

4) Les pirates informatiques internes :

Ils travaillent à l'intérieur d'une entreprise ou d'une organisation et ont accès aux systèmes et aux données sensibles. Ils peuvent être motivés par le gain financier, la vengeance ou d'autres raisons personnelles.

5) Les script kiddies :

Ils sont des pirates informatiques inexpérimentés qui utilisent des outils et des programmes préexistants pour mener des attaques. Ils n'ont souvent pas de motivations claires et peuvent agir simplement pour le plaisir ou la curiosité.

Il est important de comprendre les différents types de pirates informatiques pour mettre en place des mesures de sécurité adéquates et répondre efficacement aux attaques. [03]

4.1.3. Typologie de pirate :

La typologie des pirates informatiques peut varier en fonction de plusieurs critères tels que leur motivation, leurs compétences, leurs méthodes d'attaque et leurs objectifs. Cependant, voici une classification générale des types de pirates informatiques :

1) Les hackers éthiques :

Également connus sous le nom de « chapeaux blancs », ils sont des professionnels de la sécurité informatique qui cherchent à identifier les vulnérabilités dans les systèmes informatiques pour aider à améliorer la sécurité des entreprises et des organisations.

2) Les pirates informatiques criminels :

Également appelés « chapeaux noirs », ils ont l'intention de causer des dommages, de voler des données et de commettre des fraudes pour des gains financiers.

3) Les activistes :

Également appelés « hacktivistes », ils utilisent leurs compétences en informatique pour promouvoir une cause ou une idéologie politique. Ils peuvent mener des attaques DDoS (Distributed Denial of Service) ou des opérations de fuite de données.

4) Les pirates informatiques internes :

Ils travaillent à l'intérieur d'une entreprise ou d'une organisation et ont accès aux systèmes et aux données sensibles. Ils peuvent être motivés par le gain financier, la vengeance ou d'autres raisons personnelles.

5) Les script kiddies :

Ils sont des pirates informatiques inexpérimentés qui utilisent des outils et des programmes préexistants pour mener des attaques. Ils n'ont souvent pas de motivations claires et peuvent agir simplement pour le plaisir ou la curiosité.
Il est important de comprendre la typologie des pirates informatiques pour mettre en place des mesures de sécurité adéquates et répondre efficacement aux attaques. [03]

4.2. Les types d'attaque :

Il existe de nombreuses méthodes d'attaque utilisées par les pirates informatiques pour accéder à des systèmes ou des données sensibles. Voici une liste des types d'attaques les plus courants :[05]

1) L'attaque par phishing :

Cette méthode consiste à envoyer un e-mail ou un message contrefait qui semble provenir d'une source légitime, dans le but de tromper l'utilisateur pour qu'il divulgue des informations sensibles, telles que des identifiants de connexion ou des données financières.

2) L'attaque par injection SQL :

Cette méthode consiste à insérer des codes malveillants dans des formulaires de saisie de données d'un site web, dans le but d'obtenir un accès non autorisé à la base de données sous-jacente.

3) L'attaque par déni de service distribué (DdoS) :

Cette méthode consiste à inonder un serveur avec un grand nombre de requêtes simultanées provenant de plusieurs sources différentes, dans le but de le faire tomber ou de le rendre inutilisable.

4) L'attaque par force brute :

Cette méthode consiste à essayer de deviner des identifiants de connexion en testant toutes les combinaisons possibles de noms d'utilisateur et de mots de passe jusqu'à ce que l'accès soit autorisé.

5) L'attaque par exploitation de vulnérabilités :

Cette méthode consiste à rechercher des failles de sécurité dans les systèmes ou les applications, puis à les exploiter pour accéder à des informations sensibles ou prendre le contrôle du système.

6) L'attaque par ransomware :

Cette méthode consiste à installer un logiciel malveillant sur le système de la victime, qui chiffre les fichiers et demande une rançon pour les déchiffrer.
Il est important de comprendre les différents types d'attaques afin de pouvoir les prévenir et les contrer efficacement. Les mesures de sécurité telles que les mises à jour régulières du logiciel, les mots de passe forts et l'utilisation d'outils de sécurité peuvent aider à réduire les risques d'attaques [05].

4.3. Les techniques d'attaques :

1) Logiciel malveillant :

Les logiciels malveillants sont des programmes informatiques conçus pour nuire aux systèmes ou aux données. Il existe de nombreuses techniques d'attaque utilisées par les logiciels malveillants. [06]

2) Virus informatique :

Les virus informatiques sont l'une des formes les plus anciennes et les plus courantes de logiciels malveillants. Les virus informatiques sont des programmes qui s'insèrent dans des fichiers ou des programmes existants sur un ordinateur, et se propagent ensuite à d'autres fichiers ou programmes [07].

3) Ver informatique :

Un ver informatique est un type de logiciel malveillant qui se propage lui-même à travers les réseaux informatiques. Contrairement aux virus, les vers n'ont pas besoin de fichiers hôtes pour se propager. Au lieu de cela, ils exploitent les vulnérabilités des logiciels ou des systèmes pour se répliquer et se propager [07].

4) Logiciel espion ou Cheval de Troie :

Les logiciels espions et les chevaux de Troie sont des types de logiciels malveillants qui ont pour objectif de voler des informations personnelles, telles que des identifiants de connexion, des données financières ou des informations sensibles. [08]

5) Vol d'appareil portatifs ou mobiles :

Le vol d'appareils portatifs ou mobiles est une technique courante utilisée par les pirates informatiques pour accéder aux données personnelles et aux informations sensibles des utilisateurs [09].

6) Phishing (Hameçonnage) :

Le phishing ou hameçonnage en français est une méthode d'escroquerie sur internet qui utilise des techniques de courrier électronique frauduleux pour tromper les utilisateurs et les inciter à divulguer des informations personnelles. Les attaques de phishing se présentent souvent sous la forme d'e-mails, de sites web ou de messages instantanés qui imitent les marques et les logos de sociétés ou organisations légitimes, afin de convaincre les utilisateurs de fournir des informations personnelles telles que des informations de compte, des numéros de carte de crédit, des identifiants de connexion, et d'autres informations confidentielles ou sensibles [10].

7) Accès non autorisé à l'information :

Accès non autorisé à l'information (ou accès non autorisé aux données) désigne tout accès ou tentative d'accès à des données, à des informations ou à des systèmes informatiques sans autorisation ou consentement de la personne ou de l'organisation propriétaire de ces éléments. Cela peut inclure des actions telles que la violation de l'accès par mot de passe, le détournement d'authentification ou l'utilisation non autorisée d'un ordinateur ou d'un réseau [11].

8) Attaque par déni de service :

Il s'agit d'une attaque informatique visant à saturer un serveur, un réseau ou une application en envoyant un grand nombre de demandes simultanées. Cela peut entraîner une interruption de service pour les utilisateurs légitimes [12].

9) Attaque par mot de passe :

Il s'agit d'une attaque visant à découvrir un mot de passe (ou un identifiant) pour accéder à des informations ou des systèmes protégés. Cela peut se faire en utilisant des techniques telles que le cracking (bruteforce) ou l'ingénierie sociale (phishing) [13].

10) Attaque Man in the middle:

Il s'agit d'une attaque qui consiste à intercepter et à modifier les communications entre deux parties, sans que ces parties ne soient conscientes de la présence de l'attaquant [14].

5. Les failles informatiques :

1) Les failles informatiques :

Les failles informatiques sont des vulnérabilités ou des défauts dans les systèmes informatiques qui peuvent être exploitées pour accéder à des informations confidentielles ou pour perturber le fonctionnement normal du système. Les failles peuvent être causées par des bugs de logiciel, des erreurs de configuration, ou encore des problèmes de conception. Les pirates informatiques peuvent exploiter ces failles pour infecter des ordinateurs avec des logiciels malveillants, accéder à des données personnelles, voler de l'argent, ou commettre d'autres activités illicites [15].

2) Les failles physiques :

Les failles physiques font référence aux vulnérabilités de sécurité physique qui peuvent être exploitées pour accéder de manière non autorisée à du matériel informatique ou à des données. Cela peut inclure des faiblesses dans les serrures, les contrôles d'accès et les barrières physiques qui protègent l'équipement informatique. Les attaquants peuvent exploiter ces vulnérabilités physiques pour voler ou accéder à des données sensibles, installer des logiciels malveillants sur des systèmes informatiques ou causer d'autres types de dommages ou de perturbations [16].

3) Les failles réseaux :

Les failles réseaux (ou vulnérabilités réseau) désignent des vulnérabilités dans les réseaux informatiques, telles que des pares-feux, des routeurs, des commutateurs et d'autres périphériques réseau. Ces vulnérabilités peuvent être exploitées par des attaquants pour compromettre la sécurité du réseau, voler des informations confidentielles ou causer une perturbation du service [17].

4) Les failles Web :

Les failles web sont des vulnérabilités dans les applications web, y compris les sites web, les services web et les applications mobiles. Ces vulnérabilités peuvent être exploitées pour accéder à des informations sensibles, voler des données confidentielles, ou causer des dommages au système cible [18].

5) Les failles systèmes :

Les failles systèmes sont des vulnérabilités dans les systèmes d'exploitation, les services, les couches basses du système et d'autres éléments du système informatique. Ces vulnérabilités peuvent être exploitées pour accéder à des informations sensibles, installer des logiciels malveillants, ou causer des dommages aux systèmes cibles [19].

6) Les failles applicatives :

Les failles applicatives sont des vulnérabilités spécifiques aux applications, qui peuvent être exploitées pour accéder à des données confidentielles, modifier les données ou causer des dommages à l'application. Les failles applicatives incluent notamment les injections SQL, les attaques de script intersites (XSS), les problèmes de gestion des identités et des accès, et les failles de sécurité dans les API [20].

6. La sécurité informatique :

6.1. Objectifs :

Les objectifs de la sécurité informatique sont les buts que l'on vise à atteindre pour protéger les systèmes informatiques et leurs données.[21]

Ces objectifs peuvent inclure la confidentialité, l'intégrité et la disponibilité des données, la prévention et la détection d'attaques informatiques, la protection contre les logiciels malveillants, la sécurité des réseaux et la gestion des identités et des accès. [22]

6.2. Démarche générale :

La démarche générale de la sécurité informatique fait référence à l'ensemble des étapes et des activités à mettre en place pour garantir la sécurité des systèmes informatiques et des données stockées. Cette démarche peut inclure la gestion des risques, la définition de politiques de sécurité, la mise en œuvre de mesures techniques et organisationnelles, la sensibilisation et la formation des utilisateurs, la surveillance des activités et l'analyse des incidents de sécurité.[22]

6.3. Planification de la démarche de sécurisation :

La planification de la démarche de sécurisation informatique fait référence à l'étape initiale de toute démarche de sécurité informatique. Cette étape vise à définir les objectifs de sécurité, les contraintes, les risques et les mesures appropriées pour sécuriser les systèmes informatiques et les données associées.

Cette étape est cruciale car elle permet de définir une stratégie globale de sécurité et de s'assurer que les mesures de sécurité sont adaptées aux besoins spécifiques de l'organisation. [22]

6.4. Mécanisme de défense :

Le mécanisme de défense est un concept de la psychanalyse qui désigne des stratégies psychiques inconscientes mises en place par une personne pour faire face à des situations anxiogènes ou menaçantes. Ces mécanismes sont souvent involontaires et ont pour but de protéger l'individu contre des sentiments difficiles à gérer, tels que la peur, la colère, la culpabilité ou la honte. Les types courants de mécanismes de défense comprennent la projection, la rationalisation, la sublimation et le déni. [23]

7. La détection d’attaque :

La détection d'attaque, en sécurité informatique, fait référence à la capacité de détecter et d'identifier les attaques ou les tentatives d'attaques contre les systèmes informatiques, les réseaux ou les données. La détection d'attaque peut être réalisée grâce à des techniques de surveillance et de collecte de données, telles que les journaux et les événements système, ou à l'aide de systèmes de détection d'intrusion (IDS) et de systèmes de prévention d'intrusion (IPS). [24]

7.1.Les anti-virus :

Les antivirus sont des programmes informatiques conçus pour détecter, prévenir et supprimer les logiciels malveillants tels que les virus, les chevaux de Troie et les vers. Les antivirus fonctionnent en analysant les fichiers et les programmes sur un ordinateur, en cherchant des signatures de logiciels malveillants connus et en utilisant des techniques heuristiques pour identifier les comportements douteux.

Les antivirus peuvent être installés sur des ordinateurs personnels ou sur des serveurs pour protéger un réseau. Ils sont souvent intégrés à des logiciels de sécurité plus complets, tels que les pare-feu, pour offrir une protection complète contre les cyberattaques. [24]

7.2. Pare-feux :

Un pare-feu (firewall en anglais) est un dispositif de sécurité informatique conçu pour protéger les ordinateurs et les réseaux en empêchant les accès non autorisés de l'extérieur. Les pare-feux peuvent être des logiciels ou des matériels et agissent comme des barrières entre un réseau privé et Internet ou un réseau public. Ils contrôlent les connexions entrantes et sortantes en appliquant des règles de sécurité prédéfinies, telles que l'interdiction d'accès à certains ports ou l'autorisation de certaines adresses IP.[24]

7.3. La cryptographie :

La cryptographie est l’art de sécuriser les informations en les transformant de manière à les rendre inintelligibles pour ceux qui n’ont pas la clé de déchiffrement appropriée. Elle utilise des techniques de chiffrement pour protéger la confidentialité, l’intégrité et l’authenticité des données.

Le chiffrement consiste à prendre une information en clair, la transformer à l’aide d’un algorithme de chiffrement en une information chiffrée qui ne peut être lue que par les personnes disposant de la clé de déchiffrement correspondante. Cela permet de protéger les données en cas d’interception ou de vol.

Il existe plusieurs types de chiffrement, tels que le chiffrement symétrique, où une même clé est utilisée pour le chiffrement et le déchiffrement, et le chiffrement asymétrique, où deux clés

différentes sont utilisées, une pour le chiffrement et une autre pour le déchiffrement. Les algorithmes de chiffrement sont également classés en fonction de leur complexité, de leur efficacité et de leur résistance aux attaques.

La cryptographie est utilisée dans de nombreux domaines de la sécurité informatique, tels que les protocoles de communication sécurisés, les systèmes de stockage de données chiffrées, les signatures numériques pour garantir l'authenticité des documents électroniques, et les systèmes de paiement électronique sécurisés.

Cependant, la cryptographie ne garantit pas une sécurité absolue et peut être vulnérable à certaines attaques, telles que les attaques par force brute ou les attaques de type « man in the middle ». Elle doit être utilisée en conjonction avec d'autres mesures de sécurité pour fournir une protection optimale des données.[24]

7.4. Les systèmes de détection d'intrusion :

Les systèmes de détection d'intrusion (IDS) sont des outils de sécurité informatique conçus pour détecter une activité suspecte sur un réseau ou un système informatique [25]. Les IDS surveillent le trafic réseau pour détecter les comportements malveillants, tels que la tentative d'accès non autorisé, l'injection de code malveillant ou les attaques par déni de service. Les systèmes de détection d'intrusion peuvent être basés sur les heuristiques, l'analyse de signature ou les deux. Les techniques d'heuristiques s'intéressent aux modèles de trafic anormal qui pourraient indiquer une tentative d'intrusion. Les analyses de signatures, quant à elles, comparent les caractéristiques de chaque paquet de données à une base de données de signatures de menaces connues pour identifier les paquets malveillants.[24]

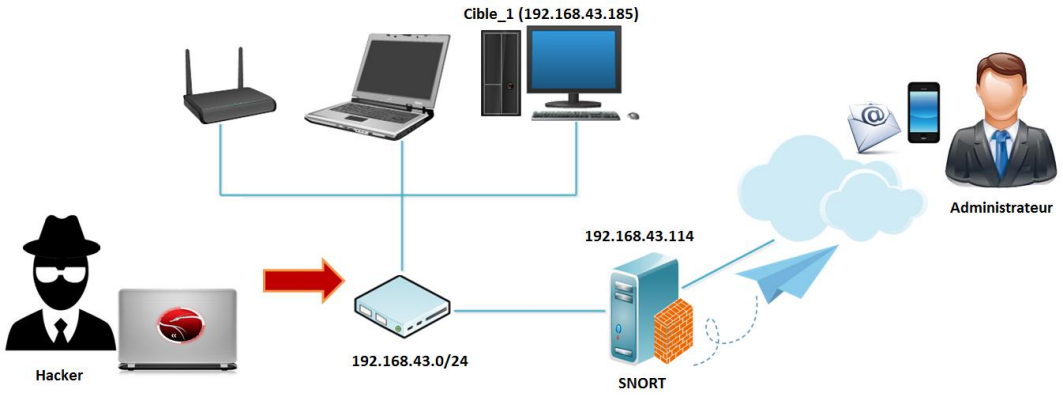

Figure 1.1 : Présentation d'un système de détection

8. Conclusion :

La sécurité informatique est primordiale pour protéger les systèmes contre diverses menaces telles que les malwares, les attaques réseaux, les attaques applicatives et les dénis de service.

Il existe plusieurs méthodes de protection telles que les anti-virus, les pare-feux, les systèmes de détection d'intrusions et la cryptographie. Dans le prochain chapitre, nous présenterons une solution relativement récente de sécurité informatique appelée "HoneyPot".

Nous commencerons par définir les objectifs de cette technologie, puis nous explorerons les différents types de honeypots, leur utilité et les interactions qu'ils permettent.

Chapitre 02

1. Introduction :

De nos jours, la fréquence des attaques informatiques ne cesse d'augmenter, soulignant ainsi l'importance cruciale de développer des outils de protection efficaces pour sécuriser les réseaux. L'un de ces systèmes de protection est le Honeypot, qui consiste à créer délibérément un environnement vulnérable pour attirer les attaquants, observer leurs méthodes et récupérer leurs outils. Ce dispositif permet aux experts en sécurité informatique d'analyser les techniques et les tendances des attaquants, afin de mieux comprendre leurs motivations et de renforcer la sécurité des réseaux.

2. Définition :

Le Honeypot est un outil de sécurité informatique qui est conçu pour détecter, attaquer ou compromettre des menaces potentielles [26]. Sa valeur réside dans sa capacité à attirer les attaquants en simulant une vulnérabilité et à enregistrer leurs actions. En analysant ces données, les experts en sécurité peuvent identifier les failles du système et prendre les mesures nécessaires pour renforcer la sécurité des réseaux. [27]

3. Objectifs :

L'objectif principal du Honeypot est de détecter les attaquants en utilisant le suivi de connexion ou la technique de détection de flux de modèles. Cela aidera par la suite l'organisation à renforcer la défense de son infrastructure informatique interne dans sa lutte contre les hackers. Les organisations doivent considérer tout trafic vers le Honeypot comme une activité suspecte, car il ne devrait pas y avoir de trafic illégitime tel que FTP et TELNET vers et depuis cette partie du réseau [28].

Il est essentiel de mesurer l'activité anormale ou malveillante pour évaluer le niveau de protection nécessaire pour garantir la sécurité du réseau de l'entreprise. Cette mesure permet de détecter les faiblesses potentielles du système et d'ajuster les niveaux de sécurité en conséquence, afin de protéger efficacement les données et les informations sensibles de l'entreprise contre les menaces et les attaques extérieures. [29]

4. Types de honeypots :

Il est possible de classer les Honeypots en deux types principaux en fonction de leur objectif : les Honeypots de production et les Honeypots de recherche. Les premiers sont utilisés pour détecter les attaques en temps réel dans un environnement de production et renforcer la sécurité de celui-ci, tandis que les seconds sont utilisés pour collecter des données et analyser les tendances en matière de sécurité dans un environnement isolé. [30,31]

4.1. Honeypots de production :

L'honeypot de production est un outil de sécurité utilisé pour protéger un réseau opérationnel en déroutant les attaques ciblant les différents services du système. En attirant ces attaques vers lui, l'Honeypot permet de réduire les risques et de renforcer la sécurité du système, tout en travaillant de concert avec d'autres mécanismes de sécurité tels que les firewalls et les systèmes de détection d'intrusions (IDS). L'Honeypot de production est également capable de détecter les attaques grâce à ses fichiers d'audit, qui peuvent être utilisés pour corriger les vulnérabilités.

4.2. Honeypots de recherche :

Les honeypots de recherche ont pour objectif d'étudier et de comprendre les comportements et techniques utilisés par les hackers, ainsi que leur évolution dans le temps. Contrairement aux honeypots de production qui ont pour but de sécuriser un système en particulier, les honeypots de recherche sont conçus pour simuler des environnements vulnérables afin d'attirer les attaquants et d'analyser leurs actions. Ils sont plus complets que les honeypots de production car ils permettent d'analyser le système dans son ensemble plutôt que de se concentrer sur un service spécifique. Cependant, leur gestion et leur analyse peuvent être complexes en raison de la sensibilité du système et des multiples résultats obtenus. Les honeypots de recherche sont donc principalement utilisés à des fins de recherche et d'analyse dans le domaine de la sécurité informatique.

5. Classification des honeypots :

La mise en place d'un honeypot dépend en grande partie du degré d'interaction du honeypot utilisé. Par conséquent, on peut classer les honeypots en trois catégories distinctes en fonction de leur niveau d’implication. [30 , 31].

Niveau D'interaction	Installation & Configuration	Déploiement & Maintenance	Informations Collectées	Niveau de Risque
Bas	Simple	Simple	Limitée	Bas
Moyen	impliqué	impliqué	Variables	Moyen
Haut	Difficile	Difficile	Riche	Haut

Tableau 2.01 : Différents niveaux des pots de miel

5.1. Honeypots" à faible interaction :

Les honeypots à faible interaction sont des simulations de services spécifiques tels que FTP ou HTTP qui ne sont pas réellement en cours d'exécution. Ils sont faciles à déployer et à maintenir car ils fonctionnent au-dessus de la couche du système d'exploitation et n'ont pas besoin de ressources système importantes. Cependant, ces honeypots ne fournissent que des informations limitées sur les activités de piratage. Leur principal avantage est qu'ils protègent le système réel contre les attaquants en limitant les dégâts qu'ils peuvent causer. Ils sont utiles pour identifier les adresses IP des attaquants. Les honeypots à faible interaction les plus connus sont Honeyd et Specter.[32]

Le fonctionnement général de ce type de Honeypot est illustré par la figure2.01

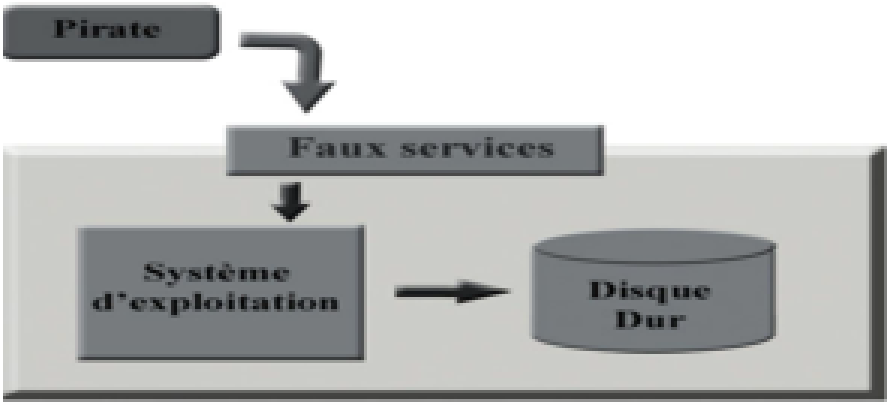

Figure 2.01: Schéma d'une interaction faible

5.1.1 Avantages

- Facilité de déploiement et de maintenance

- Protection du système ils protègent le système contre les attaquants. Les attaquants ne peuvent causer que des dommages limités à la simulation de honeypot.

- Collecte d'informations

5.1.2 Inconvénients

- Sont faciles à détecter par les attaquants à cause d'absence de réponses attendues dues à l'implémentation incomplète des services.
- il ne fournissent qu'une interaction limitée avec les pirates informatiques, ce qui signifie qu'ils ne peuvent pas fournir une image complète des tactiques et des techniques utilisées par les attaquants

5.2. Honeypots à moyenne interaction

Un honeypot à moyenne interaction est un honeypot semi-virtuel qui offre une simulation améliorée des services d'un système. Il permet de renvoyer des réponses aux attaquants, généralement fausses, pour les guider ou les dérouter sans éveiller de suspicion excessive. En plus des services simulés, il offre aussi quelques services réels, mais sans donner la possibilité au pirate de prendre un contrôle total du système

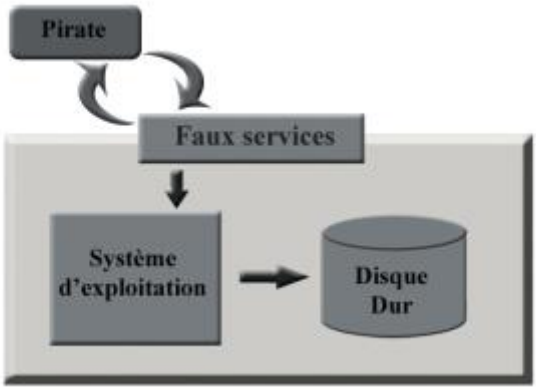

Figure. 2.02 – Schéma d'une interaction moyenne

5.2.1. Avantages :

- Simulation plus réaliste des services d'un système
- Réponses personnalisées pour guider ou dérouter les attaquants
- Collecte d'informations approfondie sur les tactiques et les motivations des attaquants
- Protection des systèmes réels en attirant les attaques vers le honeypot

5.2.2. Inconvénients

- Complexité de déploiement et de configuration.
- Risque de détection par des attaquants expérimentés.
- Coût supplémentaire pour l'acquisition et la maintenance.
- Complexité de l'analyse des données collectées.

5.3. Honeypots" à haute interaction

Les honeypots à haute interaction fournissent des informations approfondies sur les attaques, mais nécessitent des efforts plus importants lors de l'installation, de la configuration et de la maintenance. Ils présentent également un niveau de risque élevé car ils offrent aux attaquants un contrôle étendu sur le système d'exploitation. De plus, ces honeypots sont souvent déployés dans des segments de réseau non contrôlés. [33]

5.3.1. Avantages :

- Simulation très réaliste du système d'exploitation.
- Collecte d'informations détaillées sur les attaques.
- Possibilité de capturer des preuves exploitables.
- Utilisation pour tester la sécurité des systèmes.

5.3.2. Inconvénients :

- Installation, configuration et maintenance complexes.
- Coûts élevés en termes de temps et de ressources.
- Risque élevé pour les systèmes réels, donnant un accès potentiellement destructeur aux attaquants.
- Potentiellement illégaux dans certaines juridictions en tant qu'attaque active sur des systèmes informatiques.

6. Architecture d'un Honeypots

Les honeynets sont actuellement organisés en trois générations d'architectures, se distinguant principalement par la configuration du réseau mise en place pour assurer les trois principales fonctionnalités du honeynet.

6.1. Architecture de la 1ère génération

Cette génération d'architecture se caractérise par un modèle en couches, où le contrôle et la capture des données sont réalisés par différents dispositifs situés à chaque couche (routeurs, pare-feu, IDS). Cela permet une protection et une surveillance efficaces du honeynet, assurant ainsi une sécurité renforcée.[34] Autrement dit, le routeur, le pare-feu et l'IDSde cette génération sont des éléments indépendants comme le montre la figure Fig.2.3suivante.

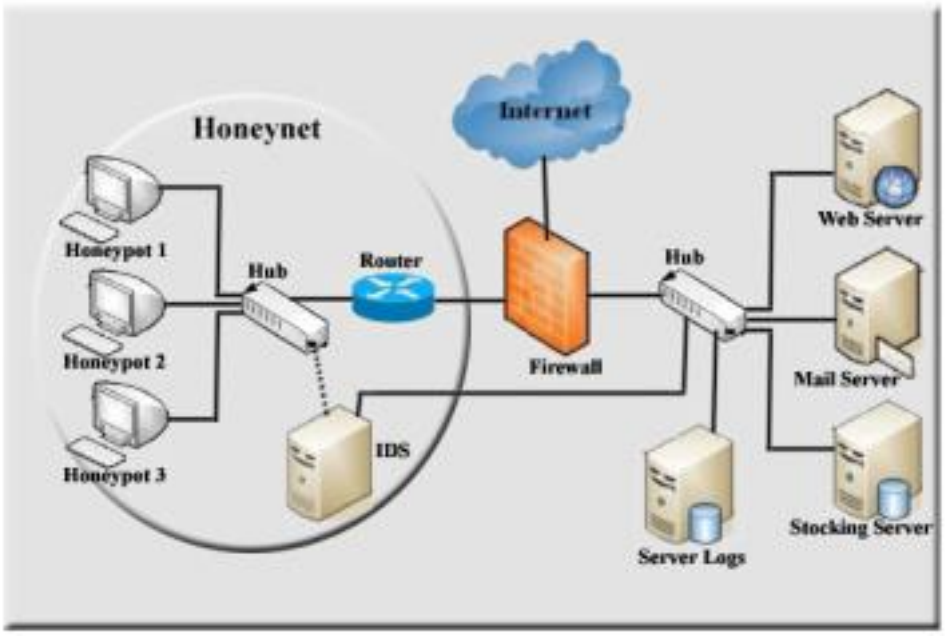

Figure. 2.03 – Architecture réseau d'un honeynet de la 1ère génération

Dans cette architecture le contrôle de données est assuré par : Cette architecture en trois couches comprend un pare-feu (première couche) qui limite les connexions entrantes, un routeur (deuxième couche) qui restreint les connexions vers Internet et cache le pare-feu du réseau interne. La capture et la collecte des données sont assurées par le pare-feu (première couche) qui enregistre toutes les connexions provenant d'Internet et du honeynet, l'IDS (deuxième couche) qui enregistre également toutes les activités du réseau du

honeynet (accessible depuis le réseau de production mais pas depuis le honeynet), et les honeypots (troisième couche) qui exploitent les journaux système.

Avec cette architecture, si l'un des dispositifs tombe en panne, la surveillance et la capture de données sont toujours assurées. Cependant, l'administration de cette architecture peut être relativement complexe en raison de sa complexité réseau. [34]

6.2. Architecture de la 2ème génération

Dans l'architecture de la deuxième génération (voir la figure Fig.2.04), la capture, le contrôle et l'analyse des données sont réalisés par un dispositif unique appelé "Honeywall". Ce dispositif est configuré en mode bridge (pont), ce qui lui permet de fonctionner au niveau des deux couches inférieures du modèle OSI (Open System Interconnection). Le Honeywall est responsable de la surveillance et de l'enregistrement des activités du réseau, ainsi que de l'analyse des données collectées pour détecter les attaques et les comportements suspects. Sa configuration en mode bridge lui permet d'intercepter et d'analyser le trafic réseau sans perturber la connectivité ou la performance du réseau.[35]

Le Honeywall, étant configuré en mode bridge, est difficilement détectable par les pirates car il n'a pas d'adresse IP propre et ne décrémente pas le champ TTL (Time To Live). Son rôle principal est de router le trafic malveillant destiné au réseau de production vers le réseau de honeypots, tout en assurant le contrôle et la capture des données relatives à ce trafic. Cette génération d'architecture de honeynets a permis d'améliorer leurs capacités, mais elle reste complexe à utiliser et à maintenir en raison du grand nombre de configurations requises, telles que la configuration des modules Snort_Inline, Sebek, IPTables, etc. Dans cette architecture, les modules IP Table et Snort_Inline du Honeywall sont responsables du contrôle des données.[36]

Dans cette architecture, le module IPTables est utilisé comme première couche pour limiter le nombre de trafics sortants du honeynet. Quant au module Snort_Inline, il est utilisé comme deuxième couche pour prévenir les attaques connues en bloquant le trafic sortant lorsqu'une attaque est détectée. En ce qui concerne la capture de données, elle est assurée à trois niveaux du Honeywall : le pare-feu IPTables, le Sniffer et le Sebek. Le Sebek est un outil essentiel pour la capture de données chiffrées, qui sont capturées au niveau des honeypots par les clients Sebeks, puis envoyées au serveur Sebek du Honeywall via un canal UDP.[30]

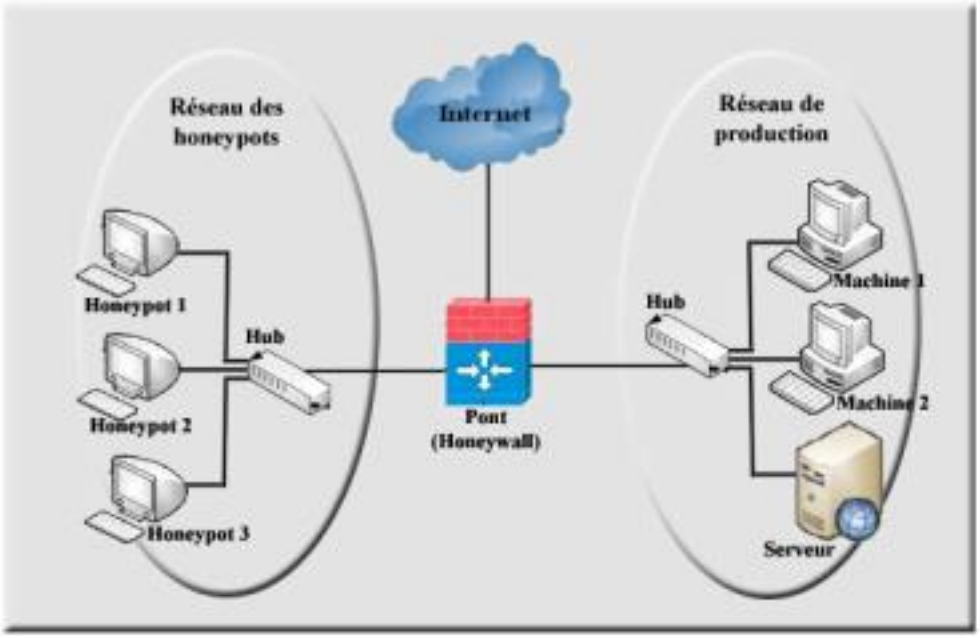

Fig. 2.04 – Architecture réseau d'un honeynet de la 2ème génération

6.3. Architecture de la 3ème génération

L'architecture de la troisième génération a été développée pour résoudre les problèmes de déploiement, de gestion et de maintenance rencontrés avec les honeynets de la deuxième génération. Elle intègre de nouveaux outils essentiels pour la collecte et l'analyse des données, regroupant tous les outils utilisés par le honeywall de la deuxième génération dans un seul CD bootable, facile à installer et à configurer. La première version de ce CD était appelée Eeyore, puis elle a été améliorée pour devenir "Honeynet ROO CDROM" en 2005.

Les outils de contrôle de données présents dans ce CD sont les mêmes que ceux de la deuxième génération (IPTables, Snort_Inline), tandis que de nouveaux outils tels que p0f et Argus ont été ajoutés pour renforcer la capture et la collecte de données. p0f est utilisé pour découvrir les systèmes d'exploitation des machines du pirate et de la cible, tandis qu'Argus permet de récupérer des informations détaillées sur les échanges de données réseau, incluant l'heure de début et de fin d'une connexion, le nombre d'octets et de paquets transmis dans chaque direction (dans le cas d'une connexion TCP bidirectionnelle).

Le CD-ROM honeywall comprend également un outil de script Perl appelé Hflow, qui fusionne toutes les données collectées par les outils de collecte dans une base de données centrale. L'analyse des données de cette génération est renforcée par une interface web graphique appelée "walleye", qui peut être lancée à distance via une connexion web sécurisée

(HTTPS). Cette nouvelle génération simplifie ainsi la gestion et l'analyse des données des honeynets.[36]

7. Analyses des attaques avec l'honeypot :

L'analyse des attaques à l'aide de honeypots est une pratique essentielle pour comprendre les techniques et les motivations des attaquants, ainsi que pour renforcer la sécurité d'un système. Voici les points clés à considérer lors de cette analyse :

1. **Collecte de données :** Les honeypots enregistrent les activités des attaquants, telles que les tentatives d'exploitation et les scans de ports. Ces données doivent être collectées et stockées de manière sécurisée pour une analyse ultérieure.
2. **Analyse du comportement des attaquants :** L'examen des journaux et des enregistrements permet de comprendre les méthodes et les outils utilisés par les attaquants. Cela aide à identifier les vulnérabilités exploitées et les schémas de comportement.
3. **Détection d'attaques avancées :** Les honeypots peuvent simuler des environnements vulnérables, attirant ainsi des attaques plus sophistiquées. L'analyse de ces attaques permet de repérer de nouvelles techniques et de développer des contre-mesures appropriées.
4. **Collecte de renseignements sur les attaquants :** En surveillant les activités des attaquants, des informations sur leurs adresses IP, leurs motivations et leurs méthodes peuvent être recueillies. Ces renseignements peuvent être partagés pour une meilleure collaboration avec d'autres organisations de sécurité.
5. **Amélioration de la défense :** L'analyse des attaques permet d'identifier les failles de sécurité et les vulnérabilités du système. Ces informations sont précieuses pour renforcer les mesures de sécurité existantes.
6. **Étude des tendances et des menaces émergentes :** L'analyse des attaques récurrentes et émergentes permet de comprendre les tendances en matière de cyberattaques et les nouvelles vulnérabilités exploitées. Cela permet d'anticiper les prochaines menaces et de prendre des mesures préventives.

8. Avantages des honeypot

Il y a de nombreux avantages des honeypots, mais nous concentrerons sur certains d'entre eux [37]

8.1. Valeur de données

Dans le domaine de la sécurité, l'un des défis majeurs réside dans l'exploitation efficace des données collectées. Les organisations génèrent quotidiennement de grandes quantités de données provenant de diverses sources telles que les journaux de pare-feu, les journaux système et les alertes de détection d'intrusion. Cependant, la quantité d'informations peut être écrasante, ce qui rend difficile la transformation de ces données en informations exploitables.

Les honeypots, en revanche, collectent une quantité relativement faible de données, mais celles-ci sont souvent très précieuses. Étant donné que les honeypots sont des systèmes isolés et ne permettent pas d'activités de production réelles, le niveau de bruit généré est considérablement réduit. Au lieu de collecter des gigaoctets de données chaque jour, les honeypots ne collectent généralement que quelques mégaoctets de données, mais chaque donnée enregistrée est potentiellement une analyse, une sonde ou une attaque, ce qui lui confère une grande valeur.

Les honeypots peuvent fournir des informations précises dans un format rapide et facile à comprendre. Cela simplifie considérablement l'analyse des données et réduit le temps de réponse. Par exemple, le projet Honeynet, un groupe de recherche spécialisé dans les honeypots, collecte en moyenne moins de 1 Mo de données par jour. Bien que cette quantité de données soit très petite, elle est principalement constituée d'activités malveillantes. Ces données peuvent être utilisées pour modéliser des statistiques, analyser des tendances, détecter des attaques et même étudier les attaquants. Il s'agit d'un effet de microscope : les données sont examinées en détail pour en extraire des informations pertinentes.

8.2. Simplicité:

Le principal avantage des honeypots réside dans leur simplicité et leur fiabilité. Contrairement à d'autres mesures de sécurité qui nécessitent des algorithmes sophistiqués et des bases de données de signatures complexes, les honeypots fonctionnent sur un principe simple. Il suffit de placer un honeypot quelque part dans l'infrastructure de l'organisation et d'attendre qu'un attaquant s'y connecte.

Aucun développement d'algorithme complexe n'est nécessaire, et il n'y a pas de risque d'erreurs ou de fausses alertes liées à la gestion de bases de données de signatures. Les

honeypots sont conçus pour être simples et efficaces. Ils permettent de détecter les activités suspectes en se concentrant sur les connexions entrantes vers des systèmes qui ne devraient pas être accessibles.

La simplicité de cette approche permet d'éviter les problèmes liés à la complexité, tels que les erreurs de configuration, les pannes ou les défaillances. En gardant la conception des honeypots simple, leur fonctionnement et leur maintenance sont plus fiables et moins susceptibles de rencontrer des problèmes.

8.3. Non besoin de signatures d'attaque connues:

Contrairement à l'IDS, les honeypots sont des outils de sécurité qui ne nécessitent pas de signatures d'attaque connues pour détecter les activités suspectes. Contrairement aux systèmes de détection d'intrusion basés sur des bases de signatures, les honeypots enregistrent toute activité comme étant suspecte, car ils sont conçus pour être des environnements factices inaccessibles aux utilisateurs légitimes. Cette approche permet de détecter les nouvelles techniques d'attaque et les variantes d'attaques existantes qui ne sont pas encore répertoriées dans les bases de données de signatures. Les honeypots sont donc utiles pour repérer les attaques zero-day et offrent une flexibilité dans la détection d'attaques inconnues ou émergentes.

9. Inconvénients des honeypots :

Il y a de nombreux désavantages des honeypots, mais nous nous concentrerons sur certains d'entre eux [38] :

9.1. Vision limitée :

Le principal inconvénient des honeypots est leur champ de vision étroit. Ils ne peuvent détecter que les activités dirigées spécifiquement contre eux. Si un attaquant cible d'autres systèmes de votre réseau, le honeypot ne sera pas conscient de ces activités, à moins d'être directement attaqué. Si l'attaquant reconnaît le honeypot comme tel, il peut l'éviter et infiltrer votre organisation sans que le honeypot en soit informé. Bien que les honeypots offrent une valeur précieuse dans l'analyse des données ciblées, leur champ de vision limité peut exclure les événements se produisant ailleurs dans le réseau. En résumé, bien que les honeypots soient utiles pour détecter les attaques ciblées contre eux, ils peuvent ne pas être

en mesure de repérer les attaques qui se déroulent ailleurs dans le réseau.

9.2. Prise d'empreinte :

Un inconvénient supplémentaire des honeypots, en particulier de nombreuses versions commerciales, est l'empreinte digitale. L'empreinte digitale se produit lorsque des caractéristiques ou des comportements spécifiques du honeypot permettent à un attaquant de l'identifier comme tel. Par exemple, un honeypot qui prétend être un serveur Web NT IIS mais présente des caractéristiques d'un serveur Unix, comme Solaris, peut être identifié par cette contradiction. Il existe différentes méthodes pour prendre des empreintes digitales d'un honeypot. Cela représente un risque encore plus important pour les honeypots de recherche, car un attaquant peut fournir de fausses informations pour tromper le honeypot et induire en erreur les responsables de la sécurité quant aux activités réelles des cybercriminels. En résumé, les honeypots peuvent être détectés par des attaquants grâce à leur empreinte digitale, ce qui peut compromettre leur efficacité et entraîner des conclusions erronées en matière de sécurité.

10. Exemples d'honeypots:

1) **Honeyd :**

Honeyd est un honeypot complet et convivial développé par Niels Provos de l'Université du Michigan. Il est conçu pour fonctionner sur les systèmes Unix et peut également être porté sur Windows. Honeyd simule des services et même de véritables systèmes d'exploitation sur des adresses IP inutilisées au sein d'un réseau. Grâce à ses fichiers de configuration, Honeyd peut émuler de nombreux services et personnaliser les réponses aux connexions, ainsi que simuler différentes piles IP pour tromper les attaquants sur la version du système d'exploitation. En résumé, Honeyd est un honeypot polyvalent et flexible qui permet de créer des environnements simulés pour attirer et surveiller les attaquants.

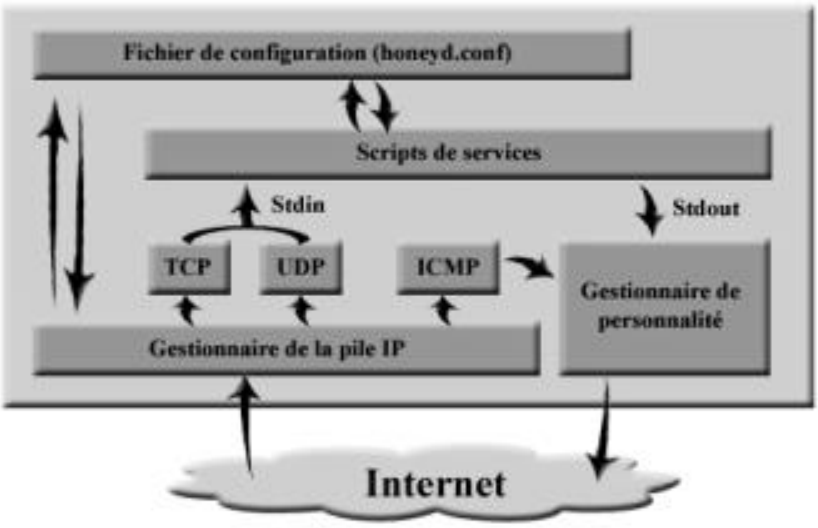

Fig. 2.05 – Architecture interne de Honeyd

2) **Kippo**

Kippo est un honeypot à interaction moyenne spécialement conçu pour étudier les attaques SSH. Il enregistre toutes les tentatives de connexion avec les noms d'utilisateur et les mots de passe utilisés lors d'attaques par force brute et par dictionnaire. Une fois qu'un attaquant se connecte avec succès au serveur SSH du honeypot, toutes les interactions effectuées via le shell sont enregistrées. Le processus d'authentification dans une session SSH classique reste le même, à la différence que le client est ici l'attaquant. Les noms d'utilisateur et les mots de passe utilisés par les attaquants sont comparés aux listes préconfigurées stockées dans le fichier userdb. Lorsque les attaquants devinent correctement les informations d'identification, ils sont autorisés à se connecter et à exécuter certaines commandes restreintes, telles que "ls" et "wget". Cependant, les attaquants peuvent souvent déterminer qu'ils sont dans un honeypot en raison de l'absence de certaines commandes Linux réelles. En résumé, Kippo est un honeypot SSH qui enregistre les tentatives d'authentification et les interactions shell des attaquants, mais qui peut être identifié par des attaquants avertis en raison de ses limitations. [39]

3) **Thug**

Thug un honeypot spécialisé dans l'analyse des attaques exploitant les navigateurs. Son objectif est d'émuler des navigateurs Web vulnérables afin d'attirer les attaquants et de surveiller leur activité. En utilisant cette approche, Thug est capable de collecter des informations sur les techniques d'exploitation et les outils utilisés par les attaquants lorsqu'ils ciblent des navigateurs. L'objectif principal de Thug est de fournir une meilleure

compréhension des vulnérabilités des navigateurs et d'aider à renforcer la sécurité contre de telles attaques.

4) **ManTrap**

Mantrap est un honeypot complet développé par Symantec. Il offre une interaction approfondie et implémente quatre systèmes d'exploitation distincts sur une seule machine hôte. Chacun de ces systèmes simule des applications réelles et fonctionne comme une entité autonome avec sa propre interface réseau. La gestion du système hôte se fait via une interface utilisateur graphique Java. Mantrap peut être utilisé à la fois comme honeypot de production, notamment pour la détection et la réaction aux attaques, et dans un contexte de recherche, bien que cela comporte des risques significatifs.

5) **Dionaea**

Dionaea est un honeypot open source conçu pour étudier et capturer les malwares et les attaques ciblant les services réseau. Il est principalement axé sur la simulation de services tels que FTP, HTTP, SMB et TFTP pour attirer les attaquants et collecter des échantillons de malwares. Dionaea est capable de capturer les flux réseau, les fichiers téléchargés, les commandes exécutées et les tentatives d'exploitation, ce qui permet une analyse détaillée des attaques.

Ce honeypot est conçu pour être extensible et modulaire, permettant aux utilisateurs de développer et d'ajouter de nouveaux modules pour détecter et capturer des types spécifiques d'attaques. Dionaea peut être configuré pour enregistrer les activités des attaquants dans des journaux et peut également être intégré à d'autres outils de sécurité et de surveillance du réseau.

11. La différence entre honeypot et IDS, IPS, Firewall

Un honeypot, un IDS (Intrusion Detection System), un IPS (Intrusion Prevention System) et un firewall sont tous des outils de sécurité utilisés pour protéger les systèmes et les réseaux, mais ils diffèrent dans leur fonctionnement et leurs objectifs.

Honeypot	IDS	IPS	Firewall
Le honeypot est un système de Prévention et Détection et Réponse.	L'IDS est un système de surveillance	L'IPS est un système de contrôle.	Un firewall est un outil informatique (matériel et/ou logiciel) conçu pour protéger les données d'un réseau.
Le honeypot est un système rendu volontairement vulnérable afin d'attirer les attaquants, observer leurs techniques et récupérer leurs outils.	Ils assurent principalement la détection des techniques de sondage, des tentatives de compromission de systèmes, d'activités suspectes internes ou des activités virales.	La fonction principale est d'empêcher toute activité suspecte détectée au sein d'un système.	Le firewall effectue des actions telles que le blocage et le filtrage du trafic.
Selon que le système leurre est destiné à surveiller les attaques extérieures ou bien internes au réseau de l'organisation, il existe trois positions possibles pour installer un honeypot : Devant le pare-feu, dans une zone démilitarisé (DMZ) ou derrière le pare-feu.	L'IDS est placé à la périphérie d'un réseau pour collecter tous les événements, enregistrer et détecter les violations.	L'IPS est placé derrière le pare-feu du réseau et communique en ligne avec le trafic entrant pour mieux prévenir les intrusions.	Il repose parfois sur un matériel réseau dédié, constituant un intermédiaire entre le réseau local (ou la machine locale) et un ou plusieurs réseaux externes.
Il existe deux types de classification : Une première qui consiste à les classer selon les interactions qu'ils permettent, et une seconde classification qui les catégorise selon leur intérêt d'utilisation.	L'IDS est divisé en fonction de l'endroit où se produit la détection et de la menace ou de la méthode de détection utilisée.	l'IPS sont divisés en différents types selon leur fonctionnalité comme Network based IPS, Host Based IPS.	Deux types de firewall existent : le pare-feu matériel et le pare-feu logiciel. En fonction de la situation, il est possible d'installer l'un ou l'autre, ou de cumuler les deux pour accroître la sécurité du réseau.

Tableau 2.02: La différence entre honeypot, IDS, IPS et Firewall

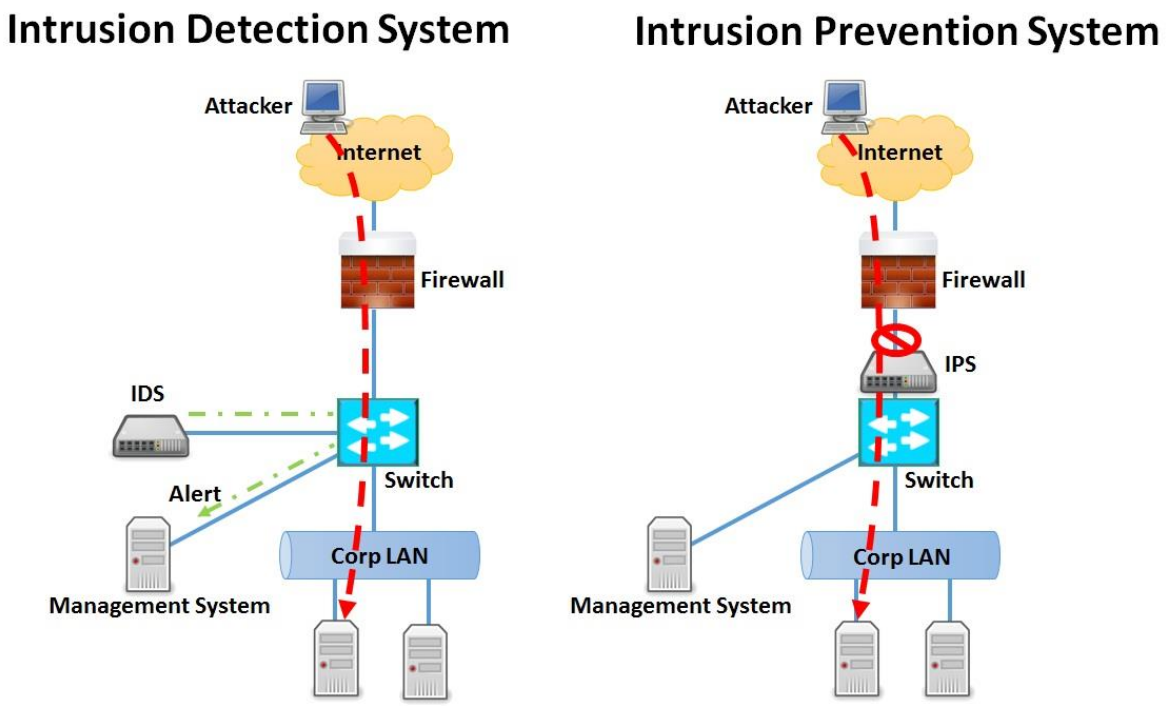

Figure 2.06 : ***Présentation de l'IDS/IPS***

12. Conclusion

Dans ce chapitre, nous avons abordé en détail les honeypots, en commençant par leur définition et leurs objectifs. Nous avons également examiné les différents types de honeypots, classés en fonction de leur niveau d'interaction et des informations qu'ils peuvent recueillir. Nous avons discuté de la mise en place des honeypots, en décrivant les étapes nécessaires pour les déployer efficacement. De plus, nous avons examiné les avantages et les limites des honeypots, en soulignant leurs capacités de collecte de renseignements précieux tout en notant leurs limites en termes de champ de vision et de détection d'attaques plus larges.

Chapitre 03

1.Introduction :

Dans ce chapitre, nous allons vous présenter Kippo, un outil de honeypot SSH écrit en Python. Kippo est spécialement conçu pour détecter les attaques par force brute et enregistrer les interactions effectuées par les attaquants.

L'installation et la configuration de Kippo seront abordées en détail, en mettant l'accent sur son exécution sur le système d'exploitation Ubuntu 14.04. Ce système d'exploitation populaire offre une base solide pour héberger Kippo et tirer parti de ses fonctionnalités de détection d'attaques SSH.

Vous apprendrez comment installer les dépendances nécessaires, configurer Kippo pour répondre aux attaques et surveiller les activités des attaquants. Nous expliquerons également comment personnaliser les options de configuration pour adapter Kippo à vos besoins spécifiques.

Grâce à ce chapitre, vous serez en mesure de mettre en place un honeypot SSH fonctionnel avec Kippo, vous permettant de collecter des informations sur les attaques par force brute et d'analyser les tactiques utilisées par les attaquants. Cette connaissance vous aidera à renforcer la sécurité de votre système en prenant des mesures préventives adaptées.

2. Présentation du projet :

Dans ce mémoire, le but principal est de détecter les attaques et identifier les ressources qui attirent les hackers afin de renforcer la sécurité des infrastructures et des données dans un réseau. Le honeypot sera utilisé pour récupérer des informations sur les hackers à l'origine de ces attaques. Pour mettre en place cette démarche, ce mémoire présentera l'implémentation d'un honeypot ainsi qu'une simulation des attaques fréquemment observées.

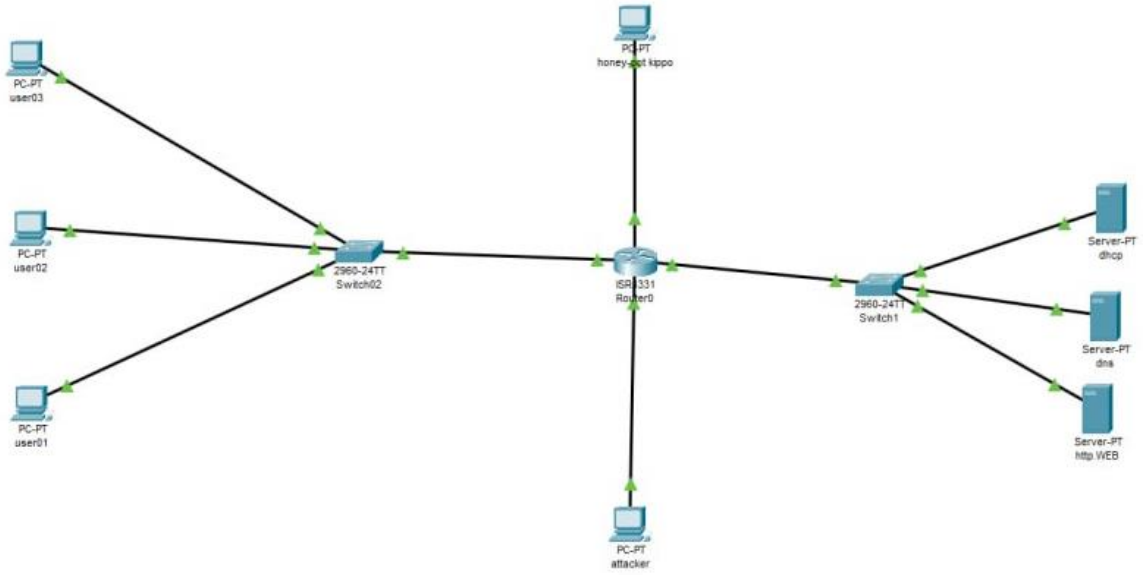

Figure 3.01: Architecture du réseau.

3.Les ressources utilisées :

3.1. Ressources matérielles :

La seule ressource matérielle utilisée dans ce projet est un ordinateur.
La configuration minimale requise est la suivante :

- RAM : 8 Go ou plus.
- Processeur : CORE i5 ou plus.
- 150 GB d’espace libre ou plus

3.2.Ressources logicielles :

Kippo peut être installé sur n'importe quelle machine qui répond aux exigences suivantes

- Un système d'exploitation (Ubuntu 14.04).
- Python 2.7.
- Twisted 8.0+.
- PyCrypto.
- Zope Interface.

4.Type de honeypot utilise dans le projet :

4.1. Présentation de « Kippo » :

C’est un honeypot SSH (Secure Shell) de niveau moyen, a été développé dans le but de recueillir des informations sur les attaques et les interactions complètes effectuées par les attaquants. Ce honeypot émule le service SSH en créant des sessions et en permettant, en cas d'authentification réussie, une interaction avec l'attaquant dans un environnement simulé. Son utilisation principale est de détecter les attaques brutales qui ciblent les mots de passe des utilisateurs pour le service SSH.

4.2. Avantages de Kippo :

1. Collecte de données détaillées.

2. Détection précoce des attaques.

3. Apprentissage des attaquants.

4. Protection des systèmes réels.

4.3. Inconvénients de Kippo :

1. Complexité de déploiement

2. Coût des ressources

3. Risque de fausses alertes

4.4. Fonctionnalités clés de Kippo :

1. Emulation SSH

2. Enregistrement des interactions

3. Simulation d'un shell

4. Journalisation détaillée

5. Gestion des fichiers

6. Surveillance des attaques

7. Configuration personnalisée

5. Installation et la configuration du honeypot « Kippo » :

Probablement tous les attaquants essaieront d'abord d'attaquer les serveurs via le port SSH par défaut 22. Changeons donc le port SSH. Utilisez n'importe quel nombre aléatoire, par exemple ici j'utilise 8925.

Pour ce faire, éditez le fichier /etc/ssh/sshd_config

 # nano /etc/ssh/sshd_config

➢ Vous devriez voir le fichier ci-dessous :

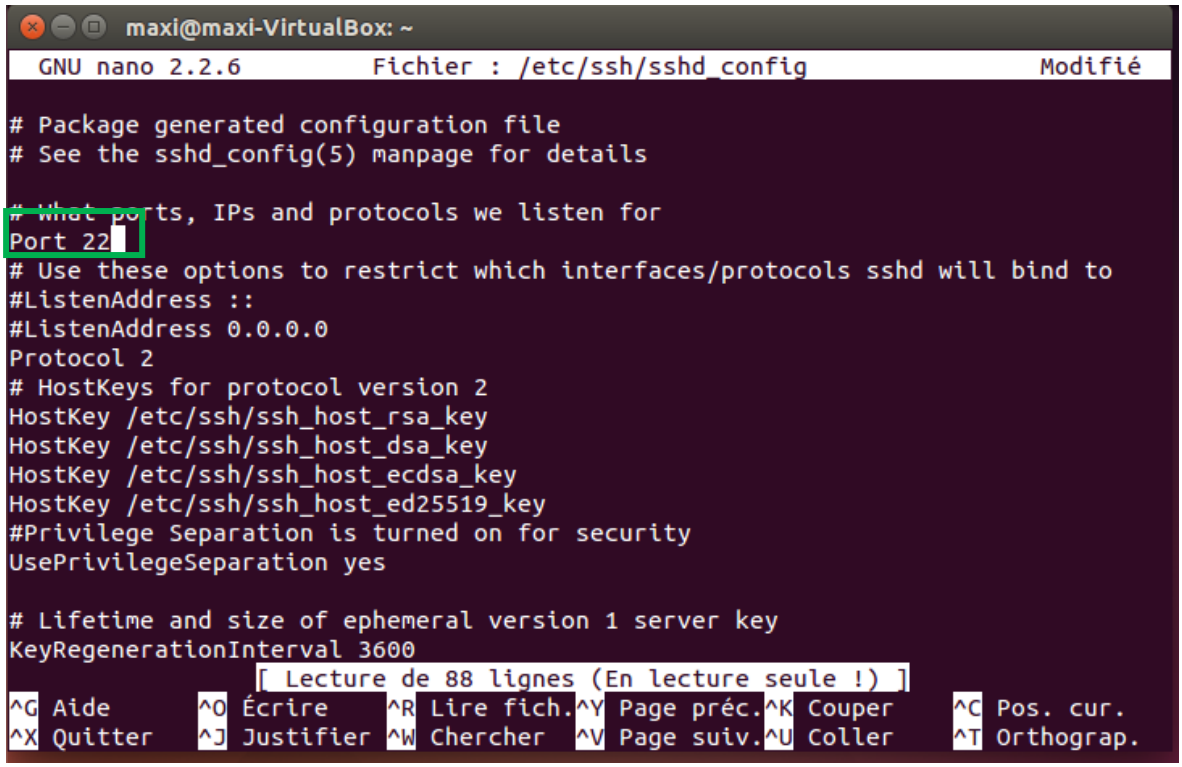

Figure 3.02: pour éditez le fichier

- Nous devons changer le numéro de port (22) en quelque chose d'autre. Dans cet exemple, ce sera 8925.

 # What ports, IPs and protocols we listen for Port 8925

- Enregistrez et quittez le fichier. Redémarrez le service ssh.

 # service ssh restart

- Cela devrait être le résultat attendu. Enregistrez le fichier et quittez dans votre terminal. Nous devons redémarrer SSH. Input:

#reload ssh

- Puisque Kippo est basé sur Python, installez certaines bibliothèques Python nécessaires à l'aide de la commande suivante. À partir de maintenant, toutes les commandes seront exécutées par l'utilisateur **root**. Vous pouvez passer a l'utilisateur **root** a l'aide de la commande **su (**utilisateur de remplacement**) .**
  Cela démarrera une nouvelle session shell en tant qu'utilisateur **root** , et toutes les commandes ultérieures que vous exécuterez auront des priviléges **root**.
  Pour passer a l'utilisateur **root** entrez :

Sudo su

  - puis nous installons les bibliothèques Python nécessaires dont nous avons besoin

 # apt-get install python-dev openssl python-openssl python-pyasn1 python-twisted

- La commande ci-dessus installera tous les packages nécessaires requis par Kippo.

  - Maintenant nous allons utiliser 'subversion' pour télécharger kippo (Subversion est un système de contrôle de version utilisé pour gérer les fichiers de projet et suivre les modifications. Il est nécessaire d'installer le honeypot Kippo car le code source de Kippo est stocké dans un référentiel Subversion) :

:-$ apt-get install subversion

- Créez un utilisateur non privilégié, ex.kippo et kippo sous cet utilisateur.

 # adduser -d /home/kippo -s /bin/bash -m kippo -g sudo

- Sur les systèmes Linux, l'utilisateur root est le seul utilisateur pouvant exécuter des ports inférieurs à 1024. Ce n'est pas non plus une très bonne idée d'exécuter Kippo en tant que root pour des raisons de sécurité. Nous utiliserons AuthBind pour sa facilité d'utilisation.

- Installez AuthBind :

#apt-get install authbind

- Créez un nouveau fichier :

#touch /etc/authbind/byport/22

- Changez la propriété en notre utilisateur Kippo :

#chown kippo /etc/authbind/byport/22

- Maintenant, déconnectez-vous et reconnectez-vous à l'utilisateur **Kippo**.

 # su kippo

- Assurez-vous que nous sommes dans notre répertoire d'accueil Kippo :

 # cd

- Maintenant vous devez télécharger git

 # sudo apt-get install git

- Pour installer le kippo, vous devez exécuter les trois commandes suivantes :

- Ouvrir le dossier opt.

 # cd /opt/

- Copiez le dossier kippo qui se trouve dans le github.

 # git clone https://github.com/desaster/kippo.git

Le répertoire Kippo contient le contenu suivant.

$ ls kippo

data dl doc fs.pickle honeyfs kippo kippo.cfg kippo.tclog start.sh txtcmds utils

➢ ou,

dl – les fichiers téléchargés avec wget sont stockés ici.

log/kippo.log – sortie de journalisation/débogage.

log/tty/ – journaux de session.

utils/playlog.py - utilitaire pour relire les journaux de session.

utils/createfs.py – utilisé pour créer fs.pickle.

fs.pickle - faux système de fichiers.

honeyfs/ - contenu du fichier pour le faux système de fichiers - n'hésitez pas à copier un vrai système ici.

➢ Remplacez le répertoire par le répertoire qui a été téléchargé :

$ cd kippo

➢ Nous voulons changer notre numéro de port, du 2222, que kippo écoute par défaut, au port 22. Déplacez et renommez le fichier de configuration fourni :

$ mv kippo.cfg.dist kippo.cfg

➢ Maintenant, nous éditons le fichier (rappelez-vous, j'utilise nano):

$ nano kippo.cfg

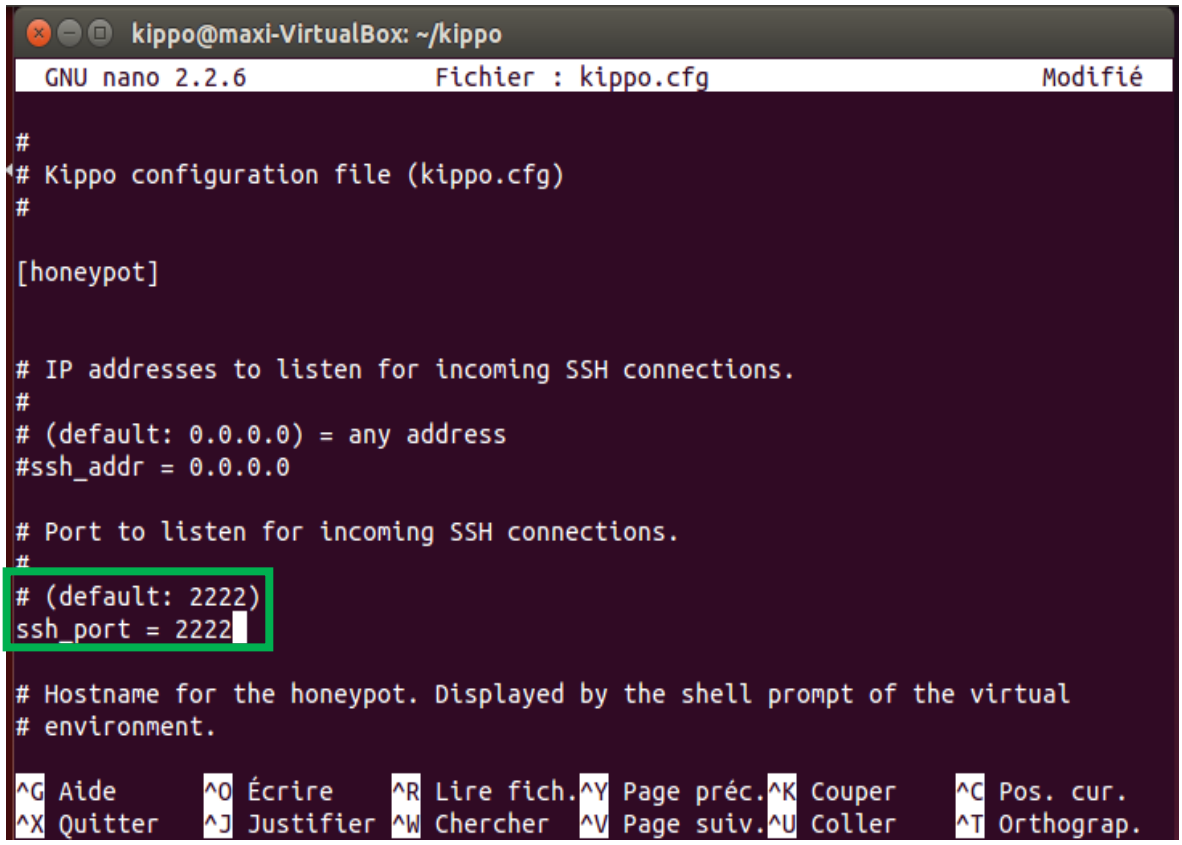

Figure 3.02: pour éditez le fichier I2

- À partir de là, nous éditons ceci :

 # Port pour écouter les connexions SSH entrantes.
#
 # (défaut : 2222)
ssh_port = 2222

- a :

 # Port pour écouter les connexions SSH entrantes.
#
 # (par défaut : 2222)
ssh_port = 22

- Sauvegarder et quitter.
- À ce stade, c'est une bonne idée de modifier le script de démarrage de kippo pour utiliser AuthBind pour notre solution SSH.
- Ouvrez le scénario :

 # nano start.sh

- Vous devriez voir ce qui suit :

```
#!/bin/sh
echo -n "Starting kippo in background..."
twistd -y kippo.tac -l log/kippo.log --pidfile kippo.pid
```

- Nous changeons:

twistd -y kippo.tac -l log/kippo.log --pidfile kippo.pid

- a :

authbind --deep twistd -y kippo.tac -l log/kippo.log --pidfile kippo.pid

- Sauvegarder et quitter.
- Maintenant, nous commençons notre script. Vous pouvez automatiser cela si jamais votre serveur redémarre :

./start.sh

- Ça devrait être ça, tout est maintenant connecté dans /home/kippo/kippo/log

Chapitre 04

1.Introduction au chapitre :

Dans ce chapitre, nous allons explorer le fonctionnement pratique du honeypot Kippo en simulant une attaque par force brute. Nous utiliserons le système d'exploitation Kali Linux, qui est couramment utilisé pour les tests de pénétration et les activités de sécurité.

Nous examinerons comment utiliser des outils populaires tels que Nmap et la commande Medusa pour lancer une attaque par force brute sur Kippo. Vous découvrirez comment Kippo détecte et enregistre ces tentatives d'attaque, ainsi que les interactions effectuées par les attaquants.

Ce scénario simulé vous permettra de comprendre comment Kippo peut fournir des informations précieuses sur les tactiques utilisées par les attaquants et comment renforcer la sécurité de votre système en conséquence**.**

2.Mise en œuvre de la simulation :

2.1.Identification des vulnérabilités :

La machine de l'attaquant, exécutant Kali Linux 1.0.6 (Debian), est un système d'exploitation spécialisé conçu pour les tests de pénétration du réseau. Il comprend divers outils pour analyser les vulnérabilités ainsi que pour les exploiter.

L'attaquant attaque le réseau en suivant les étapes suivantes :

- **Network Scanning** : en ligne ou hors ligne.
- **Port Scanning** : qu'est-ce que le service sur les ports ?
- **Service Scanning** : attaquez n'importe quel service.

✓ Maintenant, via le système root-kali, nous allons installer l'outil Nmap :

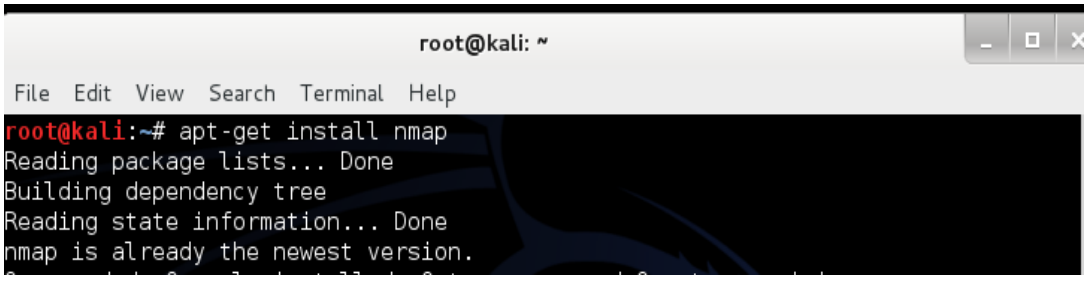

Figure 4.01 : l'outil Nmap .

✓ Scanner le réseau dans lequel se trouvent les serveurs **198.168.100.0/24**

La commande Nmap –sP : elle enverra un ensemble de requête ping pour déterminer quels appareils sont ouverts ou fermés.

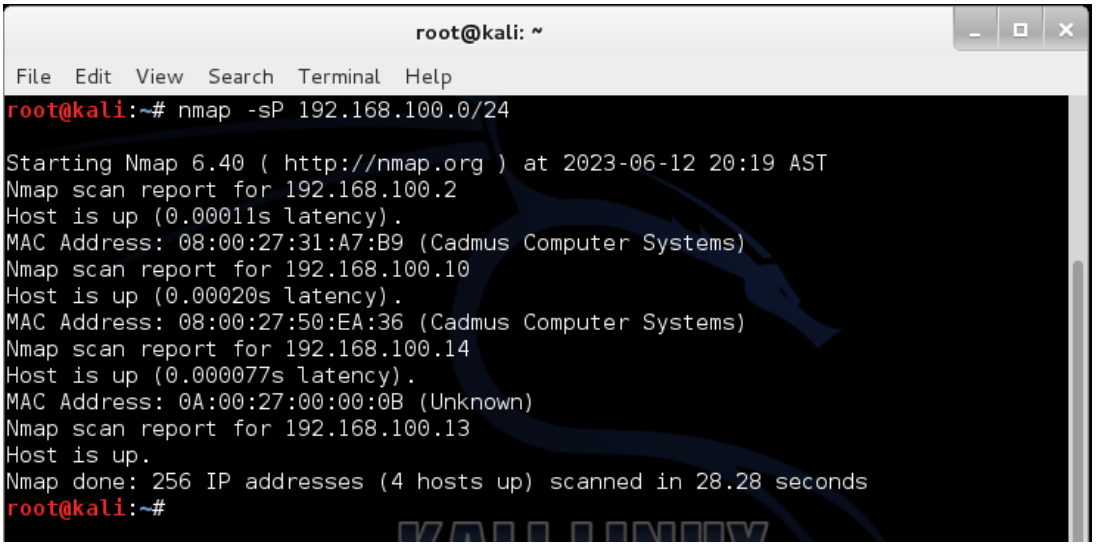

Figure 4.02: l'outil Nmap.

✓ Maintenant vérifiez les ports de tous les appareils avec lesquels la communication est disponible En ligne, nous utilisons cette commande Nmap –sS :

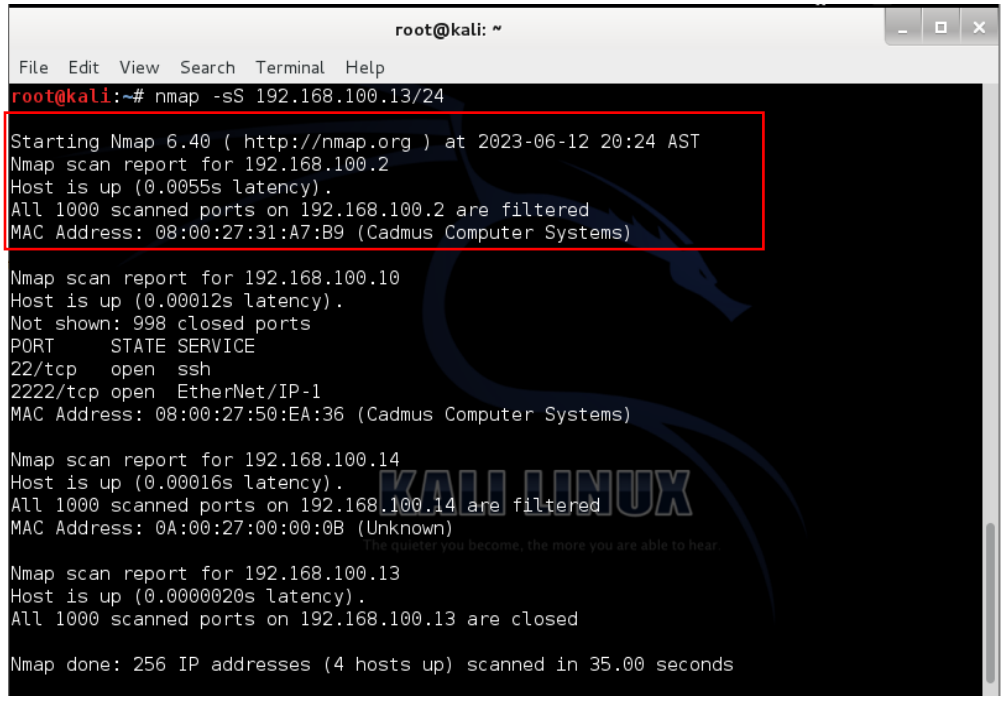

Figure 4.03 : l'outil Nmap.

✓ La première machine : Tous les 1000 ports scannés sur 192.168.100.2 sont Fermés(filtered).

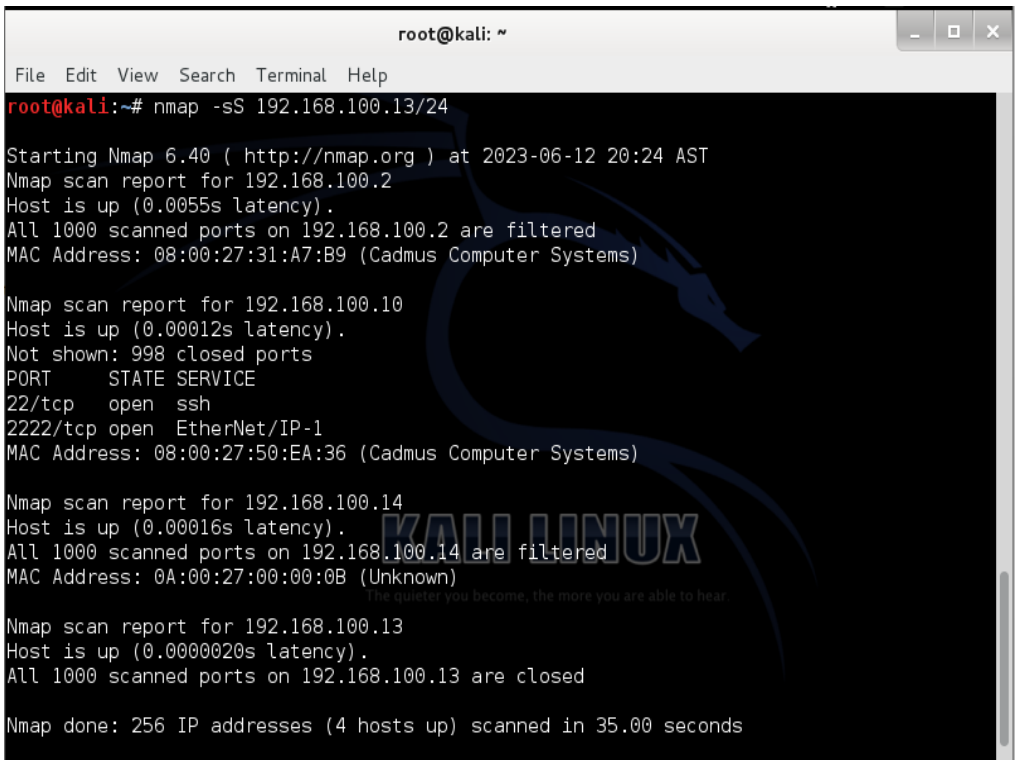

Figure 4.04 : l'outil Nmap

✓ La troisième machine : Tous les 1000 ports scannés sur 192.168.1.10 sont fermés(filtered).

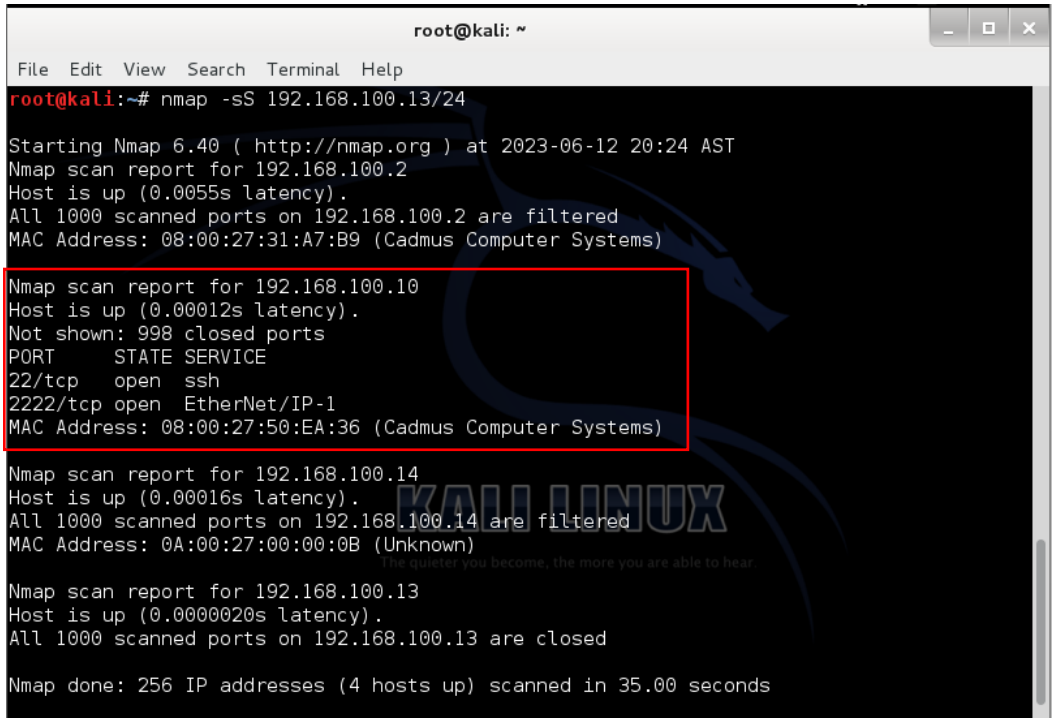

Figure 4.05: l'outil Nmap

- ✓ Le service SSH-Honeypot a répondu à l'attaquant pour tromper qu'il dispose d'un service SSH disponible pour se connecter, mais un faux qui attire l'attaquant pour l'éloigner des systèmes importants (open ssh).

2.2.Exploitation de la faille :

Maintenant l'attacker va essayer d'exploiter toutes les failles et les informations qu'il obtient pour pirater les serveurs. Premièrement : Brute-Force Attaque pour déchiffrer le nom d'utilisateur et le mot de passe du service Secur Shell.pour nous de savoir quel est le mot de passe (password) et l'utilisateur(user name) nous utilisons **'medusa'.**

Medusa : est un outil d'attaque par force brute utilisé pour le craquage de mots de passe. Il est conçu pour tester la sécurité des services réseau en tentant de deviner les identifiants de connexion via une méthode appelée brute-forcing.

mais avant d'utiliser medusa, nous avons créé une liste appelée **'passwords.txt'** elle contient des mots de passe aléatoires pour deviner le vrai mot de passe.

- ➢ Pour créé la liste qui contien les mots de passe, nous utilisons cette commande :

 # nano passwords.txt

- ➢ puis nous remplissons la liste que nous avons créée avec beaucoup de mots de passe en espérant que le bon mot de passe soit parmi eux :

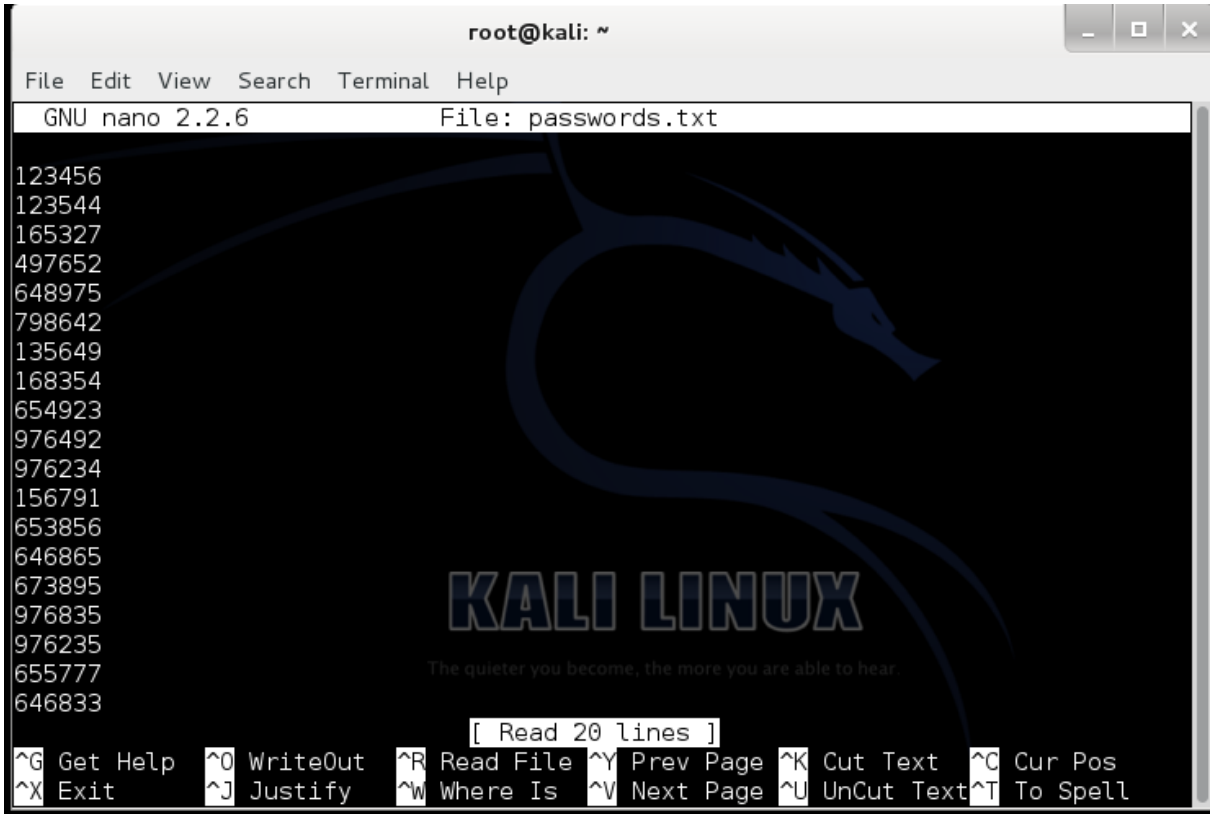

Figure 4.06: de saisir de mot de passe

- après la création de la liste de mots de passe maintenant nous l'utilisons , avec la commande medusa :

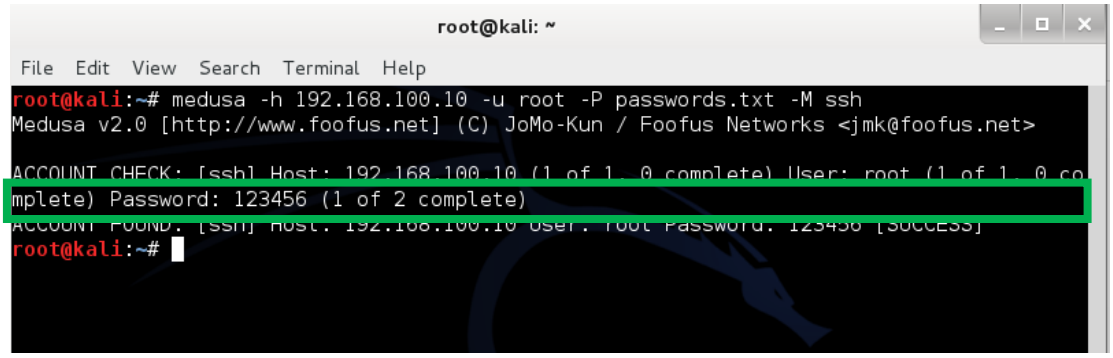

Figure 4.07: medusa command.

L'attaquant a pu obtenir le nom d'utilisateur et le mot de passe du serveur honeypot complètement isolé du vrai serveur, c'est-à-dire qu'il s'agit d'un faux système, il n'y a aucun danger sur le réseau car l'attaquant ne bénéficiera de rien, il ne fera que drainer son énergie, des ressources et du temps dans un lieu imaginaire qui ne contient pas d'information importantes, tout est faux pour l'ombrager et lui extraire plus d'information.

✓ Connexion au serveur ssh-honeypot avec l'utilisation du mot de passe et du nom d'utilisateur que nous venons d'obtenir :

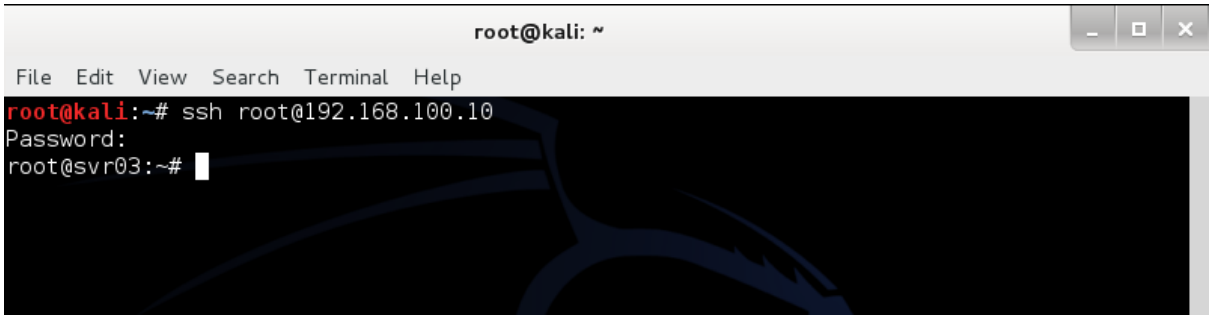

Figure 4.08 : Connexion au serveur ssh-honeypot.

✓ Et c'est parti, nous venons d'entrer dans le serveur honeypot nous utilisons la commande ls pour voir les fichiers:

→ **#ls**

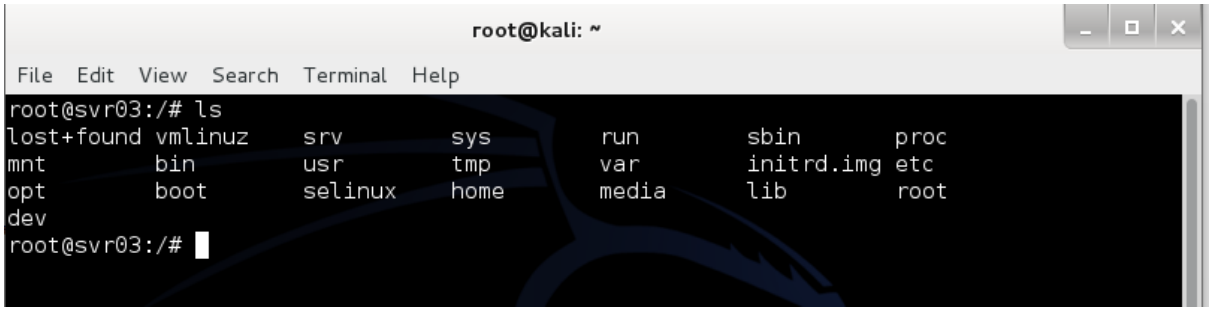

Figure 4.09 : faux fichiers que honeypot nous a donné.

✓ Ce sont tous de faux fichiers qui font croire à l'attaquant qu'il est connecté à un vrai serveur, ou il peut créer des fichiers ou supprimer des fichiers sur le serveur comme l'illustration suivante :

Ajoutez un fichier nommé 'just-test-folder 'en utilisant la commande **'mkdir'**:

→ **# mkdir just-test-folder**

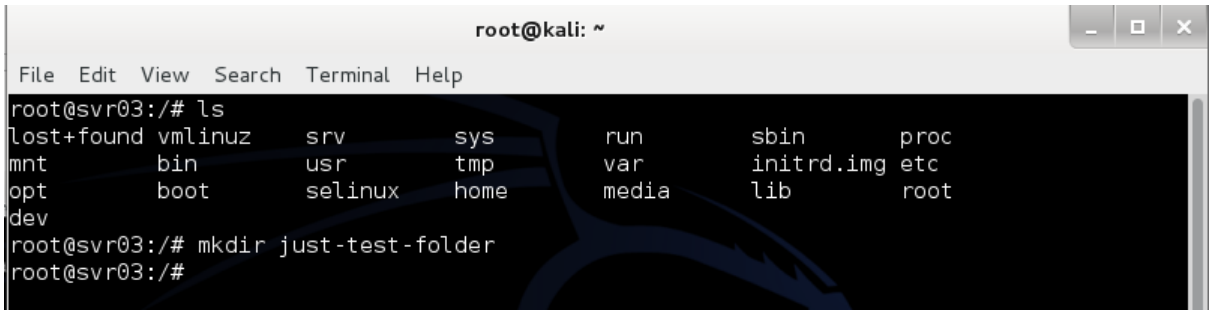

Figure 4.10 : créer un dossier nommé (just-test-folder).

✓ Maintenant, nous vérifions à nouveau les fichiers, et nous verrons notre dossier nouvellement créé avec l'utilisation de la commande **ls** :

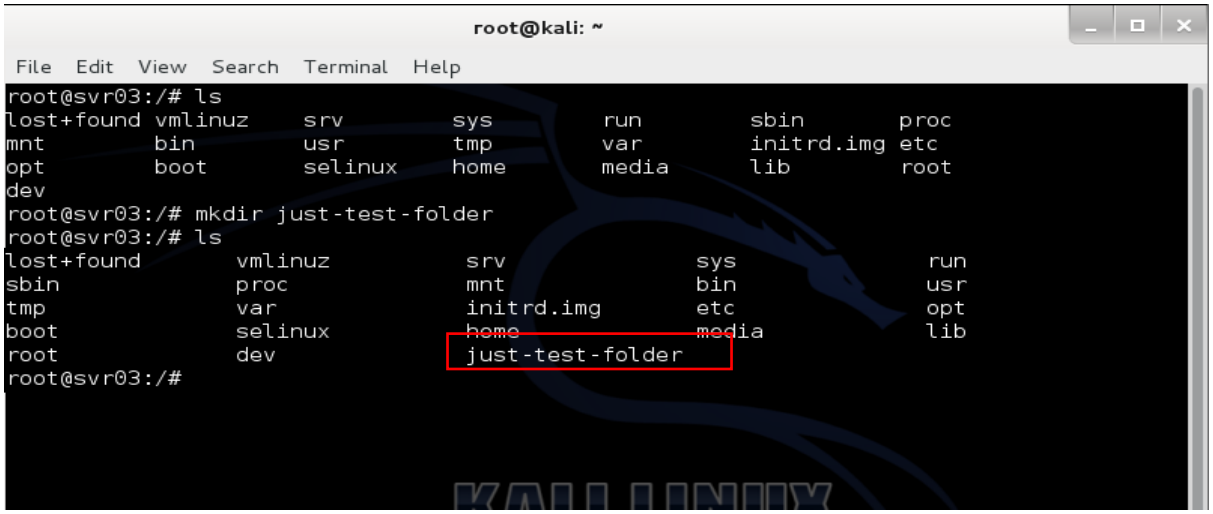

Figure 4.11: vérification du dossier just-test.

✓ Maintenant nous supprimons le dossier en utilisant la commande 'rm -r just-test-folder' et quitter le serveur :

Rm –r just-test-folfer

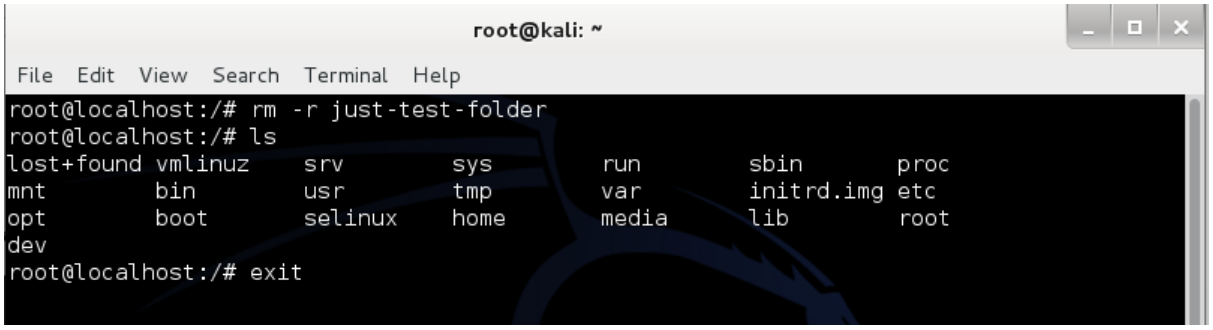

Figure 4.12 : suppression le dossier 'just-test-folder'.

- puis on sort en utilisant la commande exit :

2.3. Visualisation des données récoltées par le honeypot :

Après le processus de piratage, nous entrons dans le kippo honeypot pour voir les résultats.

✓ Nous allons dans 'Log/tty/' et nous voyons une nouvelle entrée :

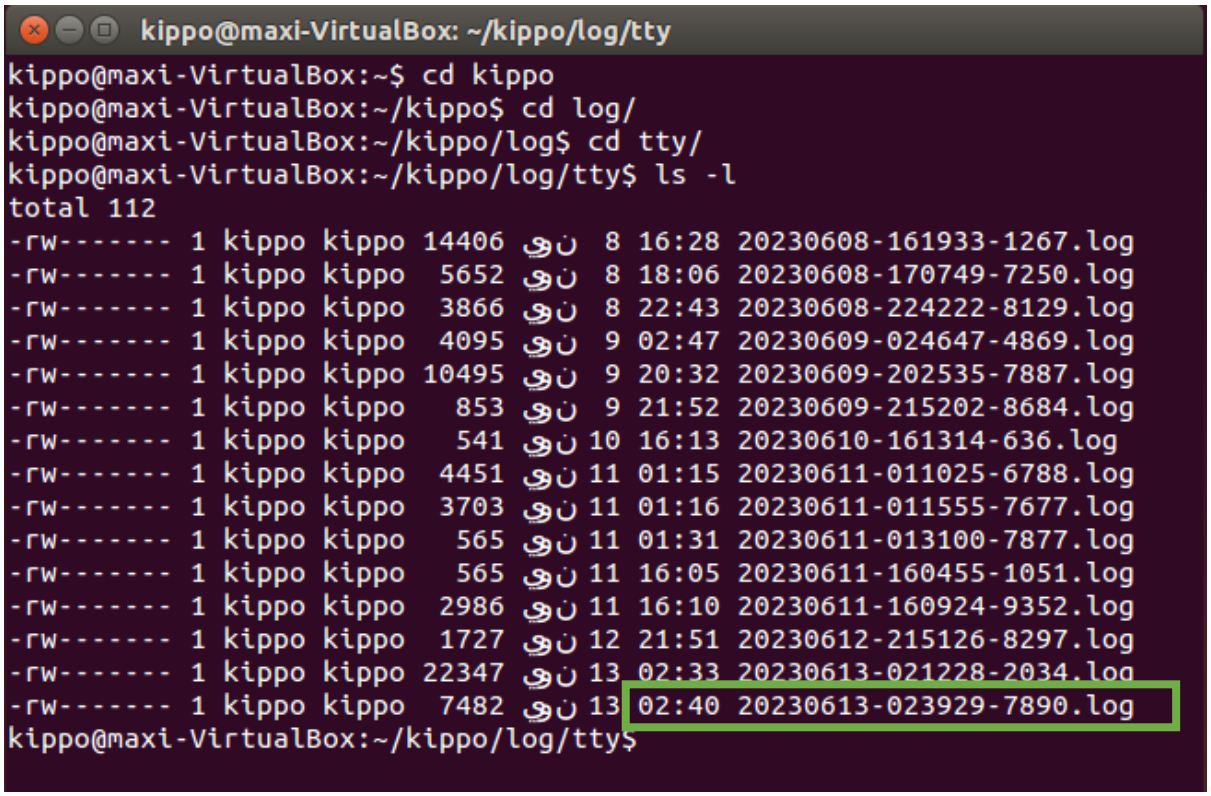

Figure 4.13 : nouvelle entrée

✓ et si nous voulons voir ce que l'attaquant a fait à nos fichiers, nous utilisons cette commande 'kippo.log' :

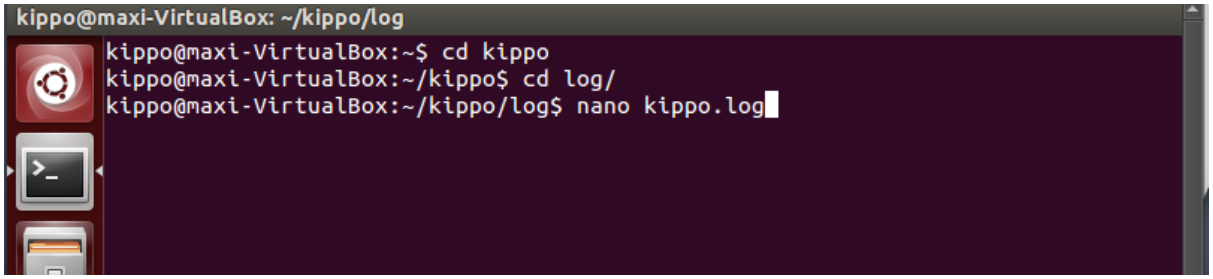

Figure 4.14 : entrer dans le kippo.log

✓ et nous pouvons lire ce Log pour voir ce que l'attaquant a fait sur le fichier kippo.log, nous voyons qu'il y avait une connexion qui a 192.168.100.13 sur le serveur et qui utilise ces commandes :

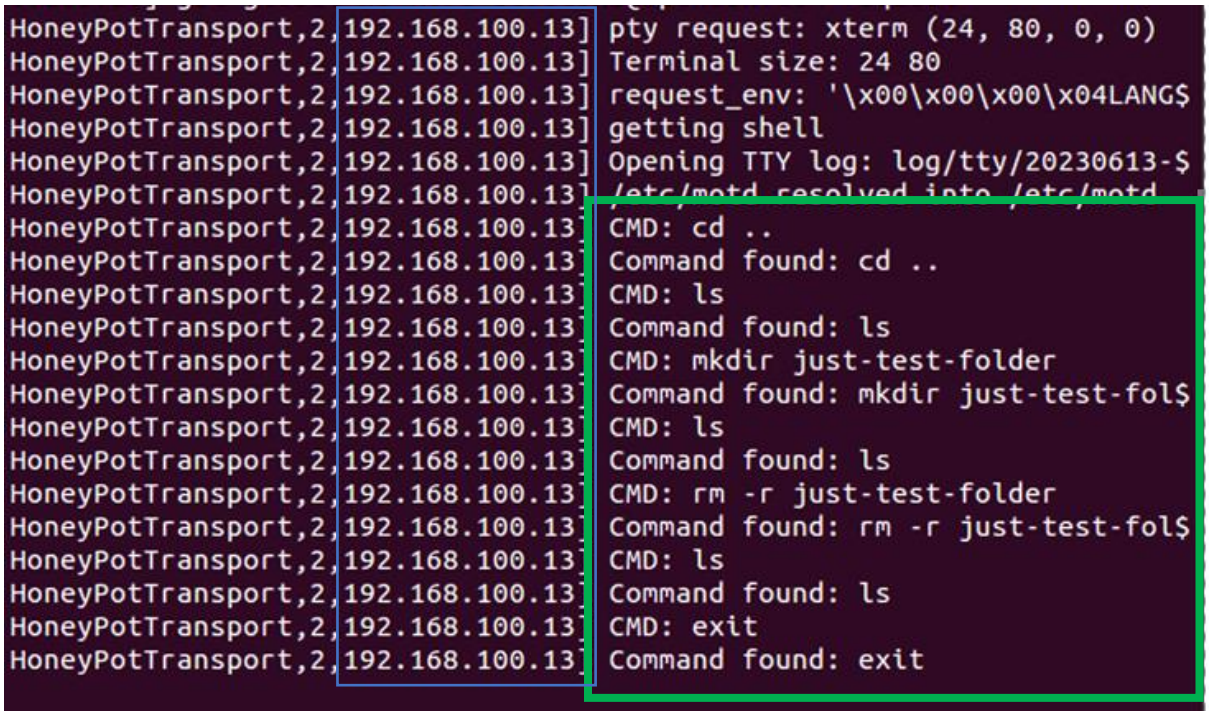

Figure 4.15 : L'affichage des résultats

3.Bilan de la simulation :

Cet examen portera sur les honeypots informatiques en général, mais pas exclusivement sur le honeypot Kippo.

En effet, il existe plusieurs types de honeypots, mais leur fonctionnement général demeure sensiblement similaire, c'est-à-dire qu'ils répondent de manière à satisfaire l'attaquant tout en recueillant des informations sur l'attaque.

Toutefois, une mauvaise administration du honeypot peut compromettre l'intégrité des données collectées. Les niveaux d'interaction dans un piège informatique offrent des avantages, mais peuvent également entraîner quelques problèmes dans certains cas.

4.Conclusion :

nous avons exploré le fonctionnement pratique du honeypot Kippo en simulant une

attaque par force brute. En utilisant Kali Linux, nous avons lancé une attaque avec des outils couramment utilisés tels que Nmap et Medusa, permettant ainsi de tester la robustesse de Kippo.

Grâce à Kippo, nous avons pu observer comment il détecte et enregistre les tentatives d'attaque, fournissant des informations précieuses sur les méthodes et les outils utilisés par les attaquants. Ces informations peuvent être utilisées pour renforcer la sécurité de votre système en prenant des mesures préventives appropriées.

En comprenant les tactiques d'attaque et en adaptant les stratégies de sécurité en conséquence, vous serez en mesure d'améliorer la résilience de votre système et de mieux vous protéger contre les attaques par force brute.

Conclusion générale

Conclusion générale

En conclusion, le honeypot se révèle être une solution adéquate pour faire face aux menaces informatiques ciblant les réseaux. Cependant, il est important de prendre en compte le fait que sa mise en place nécessite un investissement supplémentaire. De plus, il est essentiel de comprendre que le honeypot seul ne suffit pas à empêcher les attaques contre le réseau. Il doit être utilisé en conjonction avec d'autres éléments tels que les IDS/IPS et les pares-feux.

Le honeypot permet de révéler les intentions et les techniques utilisées par les pirates. Son objectif est de distraire le pirate afin de l'étudier. Il est donc d'une importance primordiale pour une entreprise de disposer d'un système de protection réseau performant. La protection des données est un enjeu crucial qui ne doit pas être pris à la légère. Les conséquences d'une attaque dans un environnement de production pourraient être considérables, notamment lorsque les pirates cherchent à corrompre ou à voler des données sensibles.

Le domaine du honeypot réunit une communauté qui travaille constamment à améliorer les capacités des produits développés, notamment à travers l'organisation "The Honeynet Project". Tout cela démontre que le honeypot est une solution de sécurité en évolution et qu'il sera probablement largement adopté par les entreprises pour contrer les tentatives d'attaques informatiques.

Enfin, cette étude a confirmé que les technologies de honeypot sont de bonnes solutions pour la sécurité des réseaux, avec leurs propres spécificités. Le honeypot est une technologie relativement jeune et en développement constant, ce qui explique sa utilisation moins répandue à l'heure actuelle. Cependant, dans quelques années, il est probable qu'il devienne l'un des composants essentiels de la sécurisation des systèmes informatiques. Cela permettrait de réduire considérablement le nombre d'attaques visant les réseaux et d'améliorer la protection des données sensibles.

Bibliographie
Et Webographies

BIBLIOGRAPHIE

[01] : Gunadiz, S. (2011). Algorithme d'intelligence artificielle pour la classification d'attaques réseaux à partir de données TCP (Thèse de magistère). Université M'Hamed Bougara, Boumerdes.

[02] : Cole, E., Krutz, R., & Conley, J. (2005). Network security bible. Wiley Publishing, INC.

[04] MUSSET, J. (2009). Sécurité informatique, Ethical Hacking: Apprendre l'attaque pour mieux se défendre. ENI.

[05] : Pierre-Louis Lussan Country Manager South-West Europe Mis à jour : 17 octobre 2022.

[07] :Bloch, L., Wolfhugel, C., Queinnec, C., Schauer, H., & Makarévitch, N. (2013). Sécurité informatique Principes et méthodes à l'usage des DSI, RSSI et administrateurs. Editions Eyrolles.

[08] :Thomas Hansjakob, Einsatz von GovWare – zulässig oder nicht?, in : Jusletter 5 décembre 2011

[09] : Sécurité D'appareils Mobiles- FOUDIL, NOUREDDINE-NOUIRI, FADI- Universite laarbi tebessi tebessa. http//localhost:8080/jspui/handle/123456789/1752

[11] : Organisation internationale de normalisation (ISO), "Gestion de la sécurité de l'information - Glossaire des termes - Partie 2: Termes liés à l'information", ISO/IEC 27000:2018.

[12] : CNIL, "Guide de la sécurité des données personnelles", 2018.

[13] : Organisation Internationale de Normalisation (ISO), "Sécurité - Manuel de l'utilisateur de la norme ISO/IEC 27001", 2013.

[14] : ANSSI, "Guide d'hygiène informatique", 2017.

[15] : Agence nationale de la sécurité des systèmes d'information (ANSSI), "Guide d'hygiène informatique", 2017.

[16] : ANSSI, "Guide d'hygiène informatique", 2017.

[17] : CNIL, "Guide de la sécurité des données personnelles", 2018.

[18] : OWASP, "The Ten Most Critical Web Application Security Risks", 2017.

[19] : ANSSI, "Guide d'hygiène informatique", 2017.

[20] : OWASP, "The Ten Most Critical Web Application Security Risks", 2017.

[22] : ANSSI, "Guide d'hygiène informatique", 2017.

[23] : Encyclopédie Universalis, "Mécanismes de défense", 2016.
[24] : "Guide d'hygiène informatique", agence nationale de la sécurité des systèmes d'information, 2017.
[25] : file:///C:/Users/AURES/Downloads/98766_NGANYEWOU_TIDJON_2020.pdf
[26] Spitzner, L. (2002). Honeypots: tracking hackers (Vol. 1). Reading: Addison-Wesley.
[27] Gerrit Göbel Jan, Dewald Andreas. (2011). Client-Honeypots: Exploring Malicious Websites. München: Oldenbourg
[28] C K Shyamala, N Harini, Dr T R Padomanabhan –
Cryptography and Security, May 2011
[30] L. Spitzner. "Honeypots : Tracking Hackers". Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, September 2002.
[32] Joshi, R. C., & Sardana, A. (Eds.). (2011). Honeypots: a new paradigm to information security. CRC Press
[33] G. Wicherski, Medium Interaction Honeypots, German Honeynet Project (avril 2006).
[34] L.Spitzner. "honeypots : Catching the insider threat". IEEE Proceedings of the 19th Annual Computer Security Applications Conference (ACSAC 2003) 1063-9527/03., 2003
[37] Mohammed, M., & Rehman, H. U. (2015). Honeypots and Routers: Collecting internet attacks. CRC Press
[38] Spitzner, L. (2002). Honeypots: tracking hackers (Vol. 1). Reading: Addison-Wesley.

Webographies :

[03] : https://www.linternaute.fr/dictionnaire/fr/definition/hacker/

[06] : https://www.mcafee.com/fr-fr/antivirus/malware.html

[10] : Gouvernement du Canada, Sécurité et protection des Canadiens, "Hameçonnage" : http://securitepublique.gc.ca/cnt/cntrng-crm/cbr-scrt/cybr-scrt/ntrnt-scrt/hmgnng-fr.aspx

[21] : https://www.chegg.com/flashcards/cybersecurite-4af78833-c42e-4b9f-bf17-31d4178630e6/deck

[29] The Honeynet Project, Papers-Know Your Enemy: The Social Dynamics of Hacking. [En ligne]. Disponible à: < http://www.honeynet.org/paper. > [Consulter le 26 mai 2022].

[31] L. Spitzner. "Honeypots : Definitions and Value of Honeypots". http ://www.csd.uoc.gr/ gvasil/stuff/honeypots/honeypots.html, 2003.

[35] Cisco-Systems-Documents. "Introduction to Internet". http ://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/introint.htm, 1999.

[36] Honeynet-Project. "Know Your Enemy : GenII Honeynets". http ://-www.honeynet.org/papers/gen2/index.html, Last access : March 2007.

[39] Kippo: An ssh honeypot. https://github.com/desaster/kippo.