École de technologie supérieure
Département de génie logiciel et des TI

Trimestre : **Hiver 2024**
Enseignant : **Oussama Boudar**

Chargé de lab. : **Hafidh Zouahi**

GTI-723 – Test d'intrusion

LABORATOIRE 2

Introduction

Lors d'un test d'intrusion, il est important de collecter le maximum d'informations sur l'environnement à tester. Le pentester doit donc passer par une première phase permettant d'identifier chaque serveur et service fonctionnant sur le réseau. Ainsi que l'extraction des certaines informations à partir de ces services et qui serviront par la suite dans la phase d'exploitation.

Le but de ce deuxième laboratoire est de :

- Familiariser l'étudiant avec les différents types de balayages réseau.
- Familiariser l'étudiant avec la phase d'énumération.
- Familiariser l'étudiant avec l'exploit des vulnérabilités applicatives.

Remarques

- Il est important de récupérer le script développé dans le premier laboratoire partie **1** question **5**, et de le transférer dans la machine Lab2m2, afin de pouvoir l'utiliser dans ce présent laboratoire.
- Pour chaque question vous devez inclure dans le rapport :

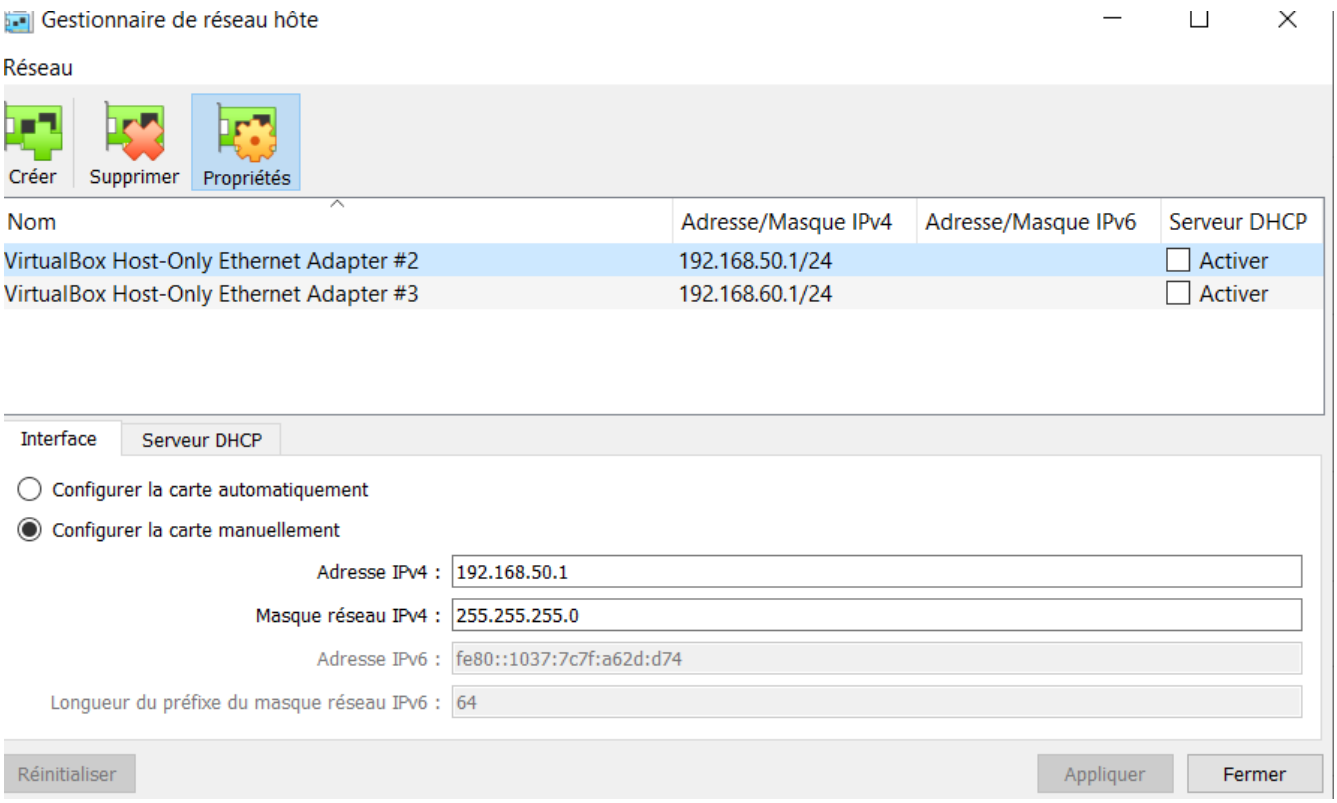
.

  - Le script (code développé)
  - Le résultat de l'exécution du script.

- Tous les scripts doivent être développés avec le langage de programmation python.
- Tous les scripts doivent être développés sur la machine Lab2m2.

Configuration du réseau

1. Installez virtualbox sur votre machine : https://www.virtualbox.org/wiki/Downloads
2. Téléchargez le fichier “Lab2.ova”.
3. Créez deux nouveaux segments réseau, en allant sur le menu « file ➔ Host network manager », et cliquez sur le bouton « create ». Voir la figure suivante :



4. Attribuez la plage d’adresses 192.168.50.1/24 au premier segment crée.
5. Attribuez la plage d’adresses 192.168.60.1/24 au deuxième segment crée.
6. Assurez-vous que le serveur DHCP est désactivé sur les deux segments.
7. Importez les machines virtuelles “Lab2.ova” sur virtaulbox, en allant sur le menu « file ➔ import appliance ».
8. Dans la section « Newtork » de la configuration des machines : Lab2M1, Lab2M2, Lab2M3, Lab2M4, Lab2M5, et Lab2M6, allez dans l’onglet « Adapter1 », choisissez « Attached to Host-Only Adapter » et attachez l’interface au premier segment crée « virualBox Host-Only Ethernet Adapter #2 ».
9. Dans la section « Newtork » de la configuration de la machine Lab2M3 allez dans l’onglet « Adapter2 », choisissez « Attached to Host-Only Adapter » et attachez l’interface au deuxième segment crée « virualBox Host-Only Ethernet Adapter #3 ».

10. Dans la section « Newtork » de la configuration de la machine Lab2M8 et Lab2M7 allez dans l'onglet « Adapter1 », choisissez « Attached to Host-Only Adapter » et attachez l'interface au deuxième segment crée « virualBox Host-Only Ethernet Adapter #3 ».
11. En allant dans : Settings ➔ Network ➔ Advanced; modifiez l'adresse MAC de chaque interface réseau sur l'ensemble des machines, comme suit :

    Lab2M1: 080027D07C89

    Lab2M2: 080027B15CEB

    Lab2M3: Interface 1: 080027A01599

    Interface 2: 08002733904C

    Lab2M4: 0800278A94BA

    Lab2M5: 0800273BEEE4

    Lab2M6: 080027A062D5

    Lab2M7: 08002713A8A4

    Lab2M8: 080027EAD5BA

12. En allant dans configuration de la machine labM7 : Settings ➔ System ➔ Processor, activer la fonctionnalité PAE/NX
13. Démarrez l'ensemble des machines virtuelles.
14. Connectez-vous sur la machine Lab2M2 :
    - Username : root
    - Mot de passe : GTI723!

I. Partie 1 : Balayage réseau

Afin de pouvoir identifier les vulnérabilités lors d'un test d'intrusion, il est important dans une première phase de repérer les machines connectées sur le réseau du client, les ports ouverts sur chaque machines, ainsi que les services hébergés derrière chaque port.

Dans cette première partie du laboratoire nous allons nous intéresser aux bases du balayage réseau. Vous devez donc sans l'utilisation des outils de scan automatiques (nmap etc.) identifier l'architecture réseau de la compagnie « **baldrinc** ».

1. Ecrire un script python permettant d'identifier les machines connectées sur un segment réseau en implémentant les deux méthodes de scan :
   a) Utilisation du protocole ICMP.
   b) Utilisation du protocole ARP.

2. Utiliser le script implémenté afin de scanner les deux plages réseaux **192.168.50.1/24** et **192.168.60.1/24** en utilisant le protocole ICMP.
3. Utiliser le script implémenté afin de scanner les deux plages réseaux **192.168.50.1/24** et **192.168.60.1/24** en utilisant le protocole ARP.
4. Comparer les résultats des deux scans, et expliquer pourquoi certaines machines n'apparaissent dans chacun des deux types de scan.
5. Implémenter un script permettant d'effectuer les types de scan suivants :
   a) Scan TCP complet.
   b) Scan silencieux
   c) Scan ACK
   d) Scan FIN
   e) Scan UDP

La machine **192.168.50.44** dispose d'une application permettant d'écrire dans les fichiers logs toutes les connexions **TCP établées** vers cette machine.
PS : pour l'ensemble des scans, utiliser la liste des ports présentés dans l'annexe.

6. Utiliser le script de scan implémenté dans la question **5** afin d'identifier les ports **TCP** ouvert sur cette machine sans laisser de traces.

La machine **192.168.50.2** dispose d'une application permettant d'écrire dans les fichiers logs tous les paquets contenant un segment de synchronisation (**SYN)**.

7. Utiliser le script de scan implémenté dans la question **5** afin d'identifier les ports **TCP** ouvert sur cette machine sans laisser de traces.
8. Identifier les ports **UDP** ouvert sur la machine en question.
9. Identifier les services hébergés dernière ces ports **UDP**, justifier votre réponse.

La passerelle **192.168.50.2** dispose d'un pare-feu permettant de filtrer certains ports entre les deux segments réseaux.

10. Le port 22 de la machine **192.168.60.77** est-il filtré par la passerelle ?

En utilisant un scan **TCP** complet pour l'ensemble des ports présentés dans l'annexe, ainsi que tous les ports entre **8070** et **8099** :

11. Identifier les ports ouverts sur l'ensemble des machines **192.168.50.5**, **192.168.50.20**, **192.168.60.49**, **192.168.60.77** :

II. Partie 2 : Enumération

Après avoir identifié les services hébergés sur l'ensemble du réseau, il est important de collecter des informations via le processus d'énumération, et qui seront utilisées par la suite afin de mieux cibler les vulnérabilités.

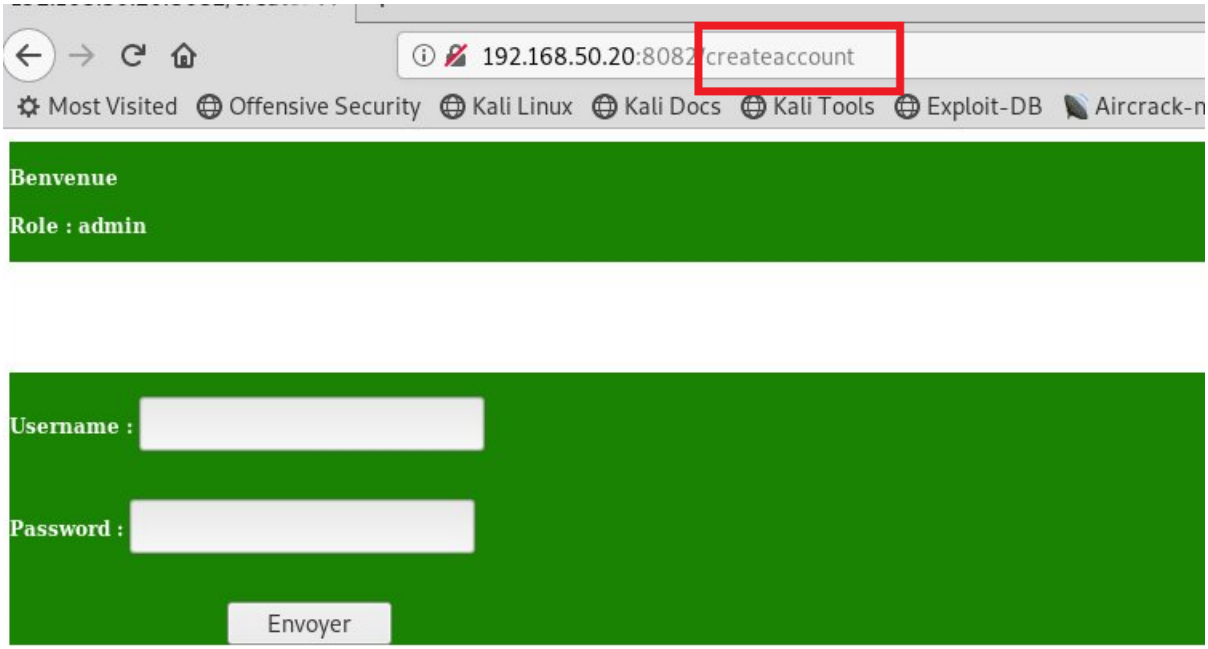
1. En se basant sur les scans effectués, quel est l'adresse IP du serveur active directory (AD).
2. Exploiter la faiblesse dans le serveur AD permettant la connexion RPC via une session nulle.
3. Quel est le nom de domaine utilisé par la compagnie « **baldrinc** »
4. Combien d'utilisateurs sont disponibles dans le domaine en question ?
5. Le serveur AD dispose-t-il d'une politique de mot de passe solide ?
6. Récupèrera la liste des utilisateurs appartenant au domaine.
7. Récupérer la liste des groupes appartenant au domaine.
8. Récupérer la liste des utilisateurs faisant parti du groupe de développeurs.
9. Récupérer la liste des utilisateurs faisant parti du groupe de Helpdesk.
10. En utilisant le nom de la compagnie « **baldrinc** » récupérer la liste des utilisateurs possédant des mots de passe faibles, Exemple : baldrinc, baldrinc2019 etc.

III. Partie 3 :

Les développeurs utilisent un serveur FTP (**192.168.60.77**) afin de partager les codes sources des applications en développement.

1. En utilisant le compte d'un développeur récupéré dans la partie précédente, accéder au serveur FTP en question et lister la liste des fichiers.

L'application Helpdesk hébergée sur la machine **192.168.50.20** est utilisée par les utilisateurs faisant parti du groupe Helpdesk, et administrée par le développeur « **jean** ». Le compte administrateur de l'application en question dispose de la page suivante permettant création des utilisateurs :



Sachant que la page en question souffre de la vulnérabilité **CSRF** et que le développeur **jean** connecté en tant que administrateur sur cette application, accède une fois chaque **6 min** au partage FTP afin de récupérer la page **contact.html** et l'exécuter afin de voir l'état d'avancement des développeurs.

2. Créer une page malicieuse dans le serveur FTP permettant l'exploit de la vulnérabilité **CSRF** afin de forcer l'utilisateur « jean » à créer un compte pour vous. PS : Les noms des deux formulaires présents dans la page sont : username et password.
3. Utiliser le compte crée afin de s'authentifier à l'application.

L'application permet l'envoi des requêtes aux agents Helpdesk. Sachant que les messages envoyés sont consultés une fois chaque **5 min** par l'utilisateur « **jean** », et que le mécanisme d'envoi de requêtes soufre de la vulnérabilité **XSS** :

4. Exploiter cette vulnérabilité afin de forcer l'application à vous envoyer les cookie de « **jean** ».
5. Intercepter les cookie sur votre machine en utilisant le script développé dans le premier laboratoire et permettant la création d'un serveur TCP.

Les cookie volées peuvent être utilisés afin de s'authentifier à l'application avec les droits administrateur.

6. En utilisant le script développé dans le premier laboratoire. Lancer un proxy **TCP** sur votre machine afin d'intercepter les requêtes HTTP échangées entre votre machine et le serveur de l'application, et d'afficher vos cookie utilisateur.
7. La page « **/adminpage** » de l'application en question nécessite une authentification avec un compte administrateur. En utilisant le script développé dans le premier laboratoire. Lancer un proxy **TCP** sur votre machine afin d'intercepter les requêtes

HTTP échangées entre votre machine et le serveur de l'application, et de remplacer les vos cookie par celles de l'administrateur.

L'accès administrateur dispose d'une fonctionnalité permettant d'afficher le contenu d'un ensemble de fichiers (Procédures d'utilisation).

8. Sachant que cette fonctionnalité soufre de la vulnérabilité **Injection de commandes**, en utilisant le script développé dans le premier laboratoire, exploiter cette vulnérabilité afin d'avoir un **reverse shell TCP**.

IV. Partie 4 :

Le serveur de fichiers **192.168.60.49** dispose d'une application permettant la gestion des images.

1. Sachant que cette application soufre de la vulnérabilité **SQL injection**, exploiter cette vulnérabilité afin de s'authentifier sur l'application en question.

Le mécanisme permettant de téléviser les images soufre de la vulnérabilité **FILE UPLOAD**.

1. En utilisant le script développé dans le premier laboratoire, exploiter cette vulnérabilité afin d'avoir un **reverse shell UDP**.

Annexe

Ports TCP/UDP

Port	Service
20-21	FTP
22	SSH
23	Telnet
25	SMTP
53	DNS
80	HTTP
88	Kerberos
110	POP3
139	Netbios
445	SMB
161,162	SNMP
389	LDAP
57	DHCP

Pondération

Introduction ---------- **3%**

Partie 1: Balayage réseau ---------- **30%**

1. Scan
    a. ICMP **2%**
    b. ARP **2%**
2. ICMP **2%**
3. ARP **2%**
4. Comparaison **2%**
5. Scan de port
    a. TCP complet **2%**
    b. Silencieux **2%**
    c. ACK **2%**
    d. FIN **2%**
    e. UDP **2%**
6. Scan TCP **2%**
7. Scan TCP **2%**
8. Scan UDP **2%**
9. Service UDP **1%**
10. Port filtré **1%**
11. Scan TCP complet **2%**

Partie 2 : Enumération ---------- **25%**

1. Adresse IP **2%**
2. Session nulle **2%**
3. Nom de domaine **2%**
4. Utilisateurs **2%**
5. Politique **2%**
6. Utilisateurs **2%**
7. Utilisateurs **2%**
8. Utilisateurs **3%**
9. Utilisateurs **3%**
10. Mots de passes **5%**

Partie 3 : ----------**25%**

1. Accès FTP **2%**
2. CSRF **5%**
3. Accès à l'application **2%**
4. XSS **5%**
5. Cookie **3%**
6. Interception cookie **2%**
7. Modification cookie **2%**
8. Reverse shell **4%**

Partie 4 : ----------**15%**

1. Injection SQL . **5%**
2. Reverse Shell . **15%**

Conclusions ---------- **2%**