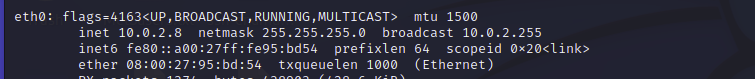
Belt Exam

The objective of this report is to show the process, both thinking and doing, in the enumeration and exploitation of the target machine. I am specifically looking for a Red Belt flag, Black Belt flag, and possibly three bonus flags.

The first thing I will start off with is using ifconfig to determine the network address of my host machine. My host machine in located at 10.0.2.9 so to locate my target machine I will use nmap 1.0.2.0/24 to locate it on the network.



The IP address of the host machine.

The nmap scan returned 4 active hosts with the target machine located at 10.0.2.20. This scan shows some promising information, but I am going to use some flags and scripts to see if I can get some more information out of it.

Graphical user interface, text

Description automatically generated

This is the IP address of the target machine.

The next nmap command I will use is nmap -sS -sV -S 10.0.2.20. This must be used in sudo mode as the some of the flags need root permission to run. The downside of this scan is that it is “noisy” and can be picked up by the target machine monitoring systems. I like this scan as is it gives state, service, and version information. The items of interest to me are the ftp anonymous log in, the ssh server and the http service on port 80.

Text

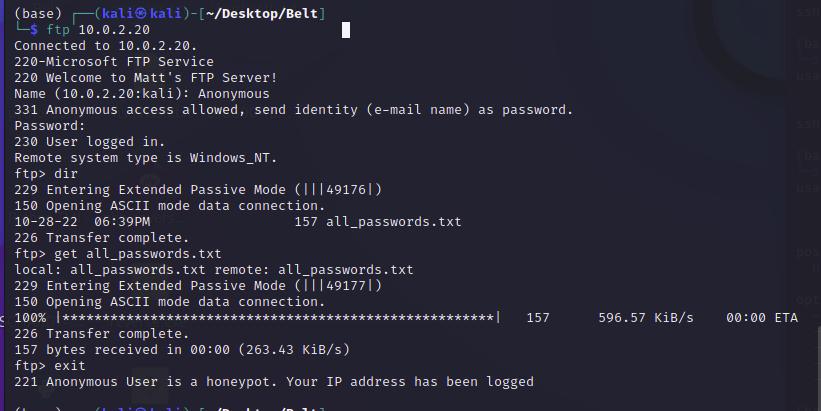
Description automatically generated

Text

Description automatically generated

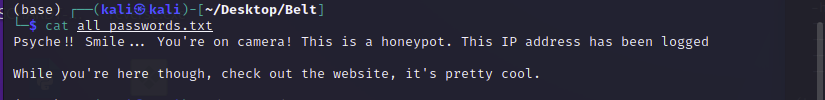
This shows the results of the second nmap scan. Sorry the blurriness of the second photo. I took several shots of it, but I think due the size this is the clearest shot I could get.

I’m going to start with the FTP server since it is there and has anonymous log in. It looks like I should have passed on it as it’s a honey pot and I got my had caught in the cookie jar. There’s a file there and since I downloaded it, I’m going to see what its information it doesn’t have in it.



Anonymous FTP Info.

The test file was in fact bogus and contains no passwords. It does encourage me to go the website. This is a mistake as it’s providing free information to the attacker, me in this case. To harden the system the ftp service needs to be disabled or have some information that doesn’t lead to the next target for the attacker. This will be my next destination.



Screenshot of the “all passwords” text document.

The website located at <http://10.0.2.20:80> contained a photo with the it “Looks more like a bridge than a flag”. I downloaded the photo to my host machine for further analysis. I also looked at the source code and found some interesting information. It says “alt=mathew”. This could be a username or password.

Graphical user interface, website

Description automatically generated

Screen shot of the webpage.

Text

Description automatically generated

Screen shot of the source code.

The first thing I did was to use steghide to find out if the photo had some hidden information in it. It needs a passcode, I could use matthew to see that is a password, but I am instead going to use another tool to brute force the password.

Text

Description automatically generated

Screen shot of steghide info.

I will stegcracker to see if the photo has a password. It does have a password and the password is “matthew”. Which I could have guessed from the information taken from the source code.

Text

Description automatically generated

Shows matthew is the password.

I used steghide to extract the file which turns out to be a text document.

Text

Description automatically generated

I used cat to read the file. Two things stand out to me here. The letters M A T & T are all capitalized. This is probably a username as that is not normal for a document have capitalized letters like that. The other thing that stands out is the use of “ROTten password like plorefrpebpxf” This is more than likely a play on the ROT-13 cipher so I will use an online decoder to see what I come up with.

Text

Description automatically generated

Contents of the thought.txt file.

The ROT-13 cipher show the phrase to be “cybersecrocks”. This is more than likely the password for the user MATT.

Graphical user interface, application

Description automatically generated

Screenshot of online decoder for ROT-13.

The first thing I thought of since this was a username and password was to try ssh. This was not a successful login . I then returned to the ftp server and used the information there to obtain a successful login. I found a ftp flag and a pcap file which I downloaded from the ftp server.

Text, website

Description automatically generated

Screenshot of successfully logging into the ftp server and downloading the files located there.

Text

Description automatically generated

Screenshot of the ftp flag.

Inspecting the pcap file I found three emails that contain some good information. There is an ssh login set up for Richmond. There is also a telnet login set up.

Text

Description automatically generated

Screen shot of the email about the ssh login being set up. The password can probably be found in the rockyou.txt file. The three headed mythical is either referring to Cerberus or Hydra. I am going to go with Hydra.

Text

Description automatically generated

Email mentioning the telnet server.

Text, timeline

Description automatically generated

Response from Richmond to Matthew.

Using hydra, I was able to find out that Richmond’s password is “password”. I will use this to login to the ssh server.

Text

Description automatically generated

Screenshot of hydra cracking the password.

The username and password successfully logged me into Richmond’s account.

Graphical user interface, text

Description automatically generated

Screenshot of initial access.

Graphical user interface, text

Description automatically generated

Screenshot of directory files.

Two files stand out to me. There is a .jpg file and a txt file. I will start with the text file, and it reveals how to obtain the red belt and how to continue onto the black belt. The image file was the file from the website, and it had the same hidden text in it.

Text

Description automatically generated

Screenshot of the text file.

Graphical user interface, text, application

Description automatically generated

Screenshot of the telnet login.

In the telnet server I find a text file contains a bonus flag.

Text

Description automatically generated

Screenshot of the telnet bonus flag.

From here I am going to create a payload in msfvenom to obtain a reverse shell in Metasploit for some addition enumeration and hopefully privilege escalation.

Text

Description automatically generated

I used msfvenom to create a reverse shell payload.

I will use scp to upload the created payload to Richmond’s system.

Graphical user interface, text

Description automatically generated

Screen shot of uploaded payload.

I will then go into MSF console and set up a listener using the multi handler. This will pick up the execution of the payload on Richmond’s system and provide me with a reverse meterpreter shell.

Text

Description automatically generated

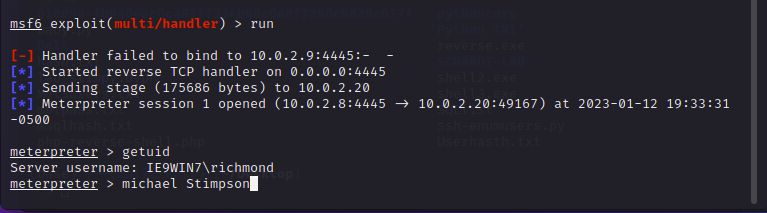
Screenshot of multi handler.

I will then return to Richmond’s system and execute the payload. This is accomplished by using .\shell.exe. Which will return the meterpreter shell in Metasploit.

Text

Description automatically generated

Screenshot of Richmond’s system and execution of the payload.



Screenshot shows the successful meterpreter shell in Metasploit of the user Richmond.

I believe Red Belt has been achieved! Woo-hoo!

Now onto the privilege escalation. The first thing I did was run WinPeas to see what information I could find. I was a little disappointed as it was lacking in information and didn’t have as much useful information as I had hoped for. The note to Richmond mentioned a xml file. I know this file can be located at C:\Windows\Panther\unattend.xml, so I will proceed to the file to see what I can locate.

Graphical user interface, text

Description automatically generated

Screen shot of C:\Windows\panther directory.

The xml file contains a coded password for the user IEUser. The password in encoded in base 64 so I will use an online decoder to get the plain text. The password is decoded to “qwerty12345”.

Text

Description automatically generated

Screenshot of the xml file.

Graphical user interface, text, application, email

Description automatically generated

Screenshot of the decoded password.

I took the username and password as was unsuccessful with several login attempts. I used it to both ftp and ssh into 10.0.2.20, with both attempts producing no logins. I then checked it against smb, and it did login, or at least had working credentials for smb. I went back and reread the note to Richmond and realized I could use to log into the IEUser on the desktop to obtain the note and so that is what I did.

Graphical user interface, text, application

Description automatically generated

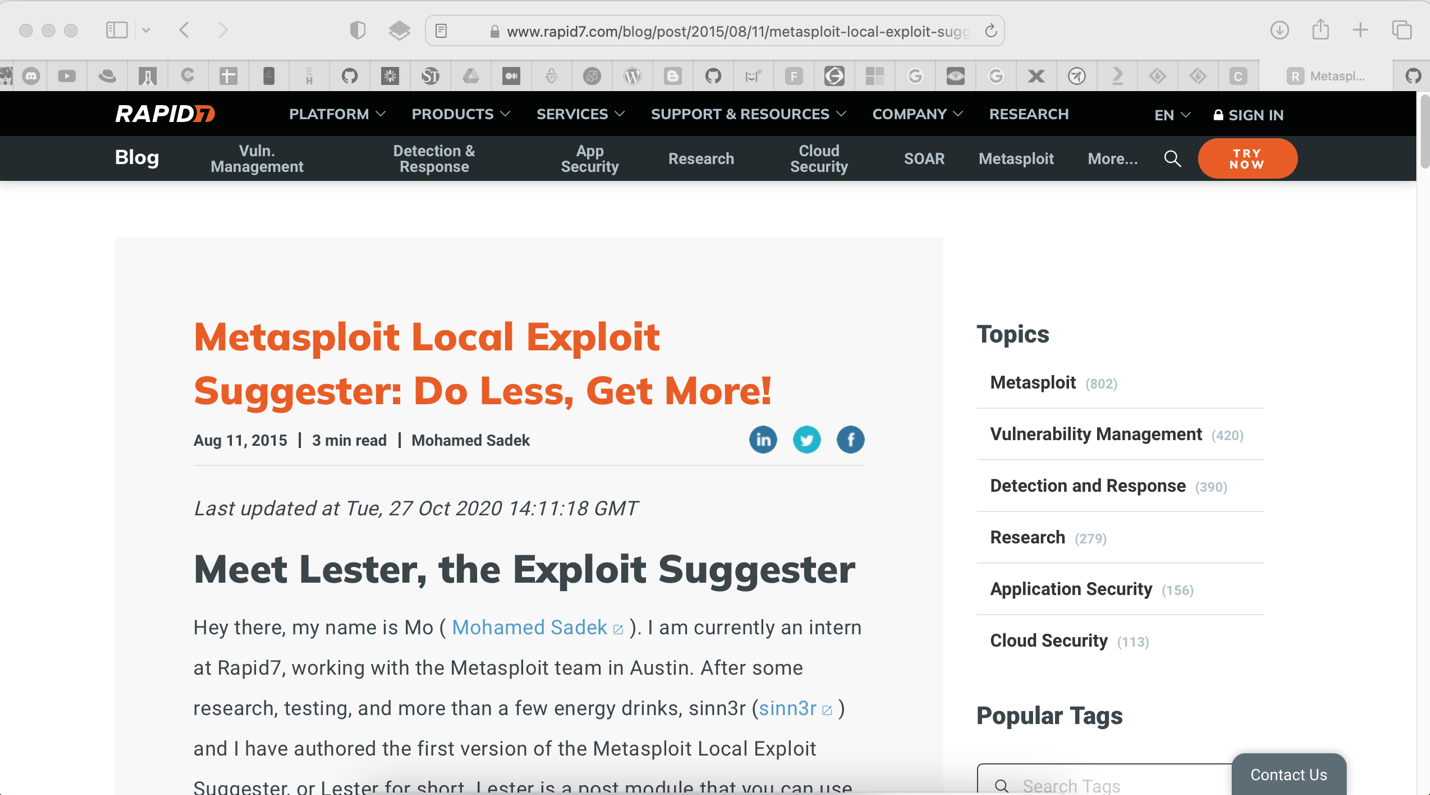
Screenshot of the desktop login an obtaining the Black Belt flag.



Screenshot of Richmond’s permissions.

I was a little frustrated as I tried Juicy Potato and thought that should have worked to give me privilege escalation at the meterpreter level. I was wrong because in my tired state I confused the privileges needed. So, I wasted my time trying several CLSID’s manually with no success logging in. So, I returned to google and tried several options. In the meterpreter shell I tried use priv and getsystem, with both producing no successful results. After more googling I finally started searching for exploit/windows/local. I tried to use this in the meterpreter shell but had close to 100 results returned. Google provided me with the following website.

<https://www.rapid7.com/blog/post/2015/08/11/metasploit-local-exploit-suggester-do-less-get-more/>



Screenshot of rapid7 website.

Reading the site, it stated there was an exploit suggester scrip in Metasploit for privilege escalation. Using this script produced 9 suggestions that might work for privilege escalation.

Text

Description automatically generated

Screenshot of exploit suggester.

Text

Description automatically generated

List of available exploits to use.

After trial-and-error exploit number 8 finally worked. There was one other exploit on the list that started to produce a shell, but it shut down each time. This did produce a reverse shell at the system level, and I was able to obtain the belt flag through the command line.

Text

Description automatically generated

Screenshot of obtaining system level access.

Text

Description automatically generated

Screenshot of obtaining flag through the command line.

MySQL

This one had my stumped for several days. I knew there was a misconfiguration of the login credentials. I could login with about any name in the world without a password, but root required a password. I was obviously using the wrong password list to try to brute force the password because every time in the process the target machine would shut down and I would have to start over. After reading an article over breakfast it came to me to try “root” as the password.

Text

Description automatically generated

Screenshot of successful root login.

This shows an additional database that is not available to another name used to login. This is the “MySQL” database. I had spent enough time I returned to Metasploit and decided on the ﻿exploit(multi/mysql/mysql\_udf\_payload) as my method of choice to start with. I had been though all the options previously and thought this would give me the best results.

Text

Description automatically generated

Screenshot of exploit used.

I set the required options and ran the exploit.

Text

Description automatically generated

Screenshot of options set.

Upon running the exploit, it was successful and returned to a meterpreter shell with NT Authority privileges. From here I located the 2nd black flag.

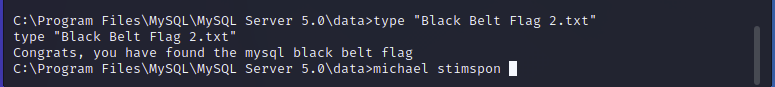
Text

Description automatically generated

Graphical user interface, text

Description automatically generated

Screenshot of successful login at meterpreter shell.



Screenshot of second black belt.

So, the MySQL server is in fact misconfigured and can be exploited. The password for root needs to be changed to a long strong password and the ability of any other user to sign in without a password needs to be removed.

One item I failed to document in my original documentation was clean up. I removed all files installed on the target machine by removing them from the meterpreter shell.

Graphical user interface, text

Description automatically generated

Screenshot of deleting files.

Conclusion, Matthew’s system is both vulnerable to local and privilege exploitation. I would disable ftp anonymous login. If it was left in place, I would not leave a note letting someone know where to go next. If a hidden note was left behind for Richmond in the .jpg file, it would be better have given the cipher in person instead of leaving in the note. This allows bad actors, like me, to use the clues to home in on the cipher used and easily crack the passwords. To prevent privilege escalation, I would suggest hardening the system. I would run a Nessus scan or use another tool to determine all the weakness on the system and fix accordingly. MySQL needs to have a long strong password in place for the root account and all other login options should be disabled.

This was challenging, specifically when it came to privilege escalation. What I learned is if I’m not finding information I need, I need to change the way I ask or search for the question.