RickdiculoslyEasy Walkthrough

By: Pierson Carulli

3 January 2019

Rickdiculosly Easy Walk Through

Rickdiculosly Easy was a fun machine to exploit. The objective of the Rick and Morty themed box from Vulnhub was to capture the flags hidden throughout the machine. Organization is key if you expect to capture the flags.

Prerequisites

In order to follow along you will need to have Virtual Box installed, which can be downloaded from www.virtualbox.org. You will also need a machine to use as the attacking machine. Kali Linux is what was used in this tutorial and can be downloaded from www.kali.org. Setting up the attacking and target machines is beyond the scope of this walk through.

Gathering Information

In any penetration test the information gathering phase is crucial. During the information gathering phase a penetration tester will gather as much information about the target as possible. This usually involves uncovering email addresses of employees, phone numbers, and websites deployed by the target. However, since this is a single virtual machine, we can skip most of the information gathering phase, and start to scan for open ports running on the target. Running a basic Nmap scan against the target machine (192.168.53.3) yielded the following results:

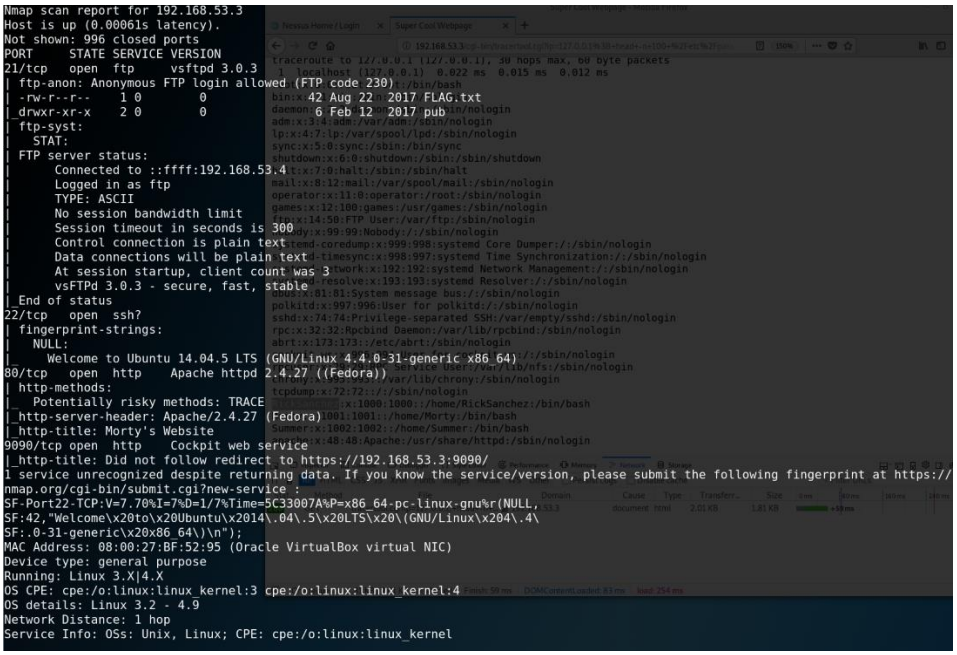


Figure 1:

The figure, shown on the left, displays the results of the Nmap port scan. The command issued at the command line was nmap -A -T4 192.168.53.3.

The first thing to note is that the target machine allows anonymous FTP access, which can allow anyone to login to the FTP server. Furthermore, the Nmap scan results reveal a FLAG in the ftp server's anonymous login directory. Running the command ftp 192.168.53.3 allows access to the ftp server (NOTE: if FTP is not installed on the attacking machine, run the command apt-get install ftp before continuing).

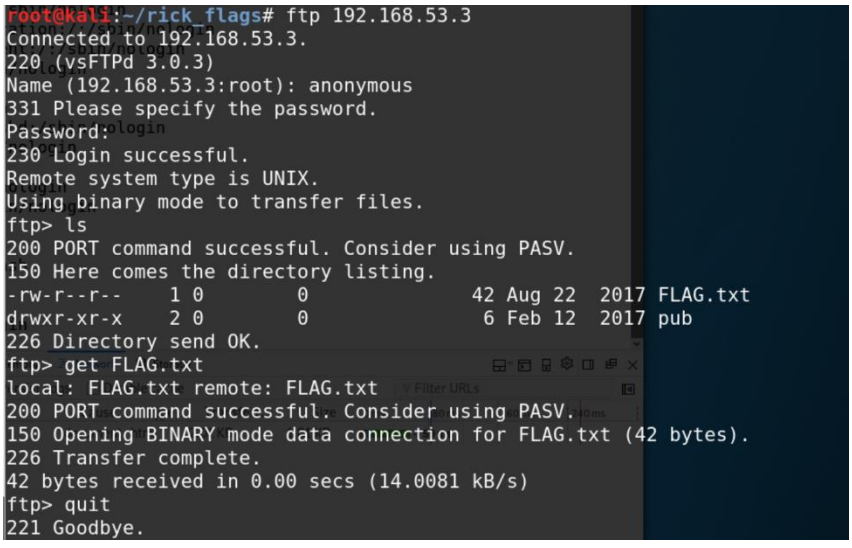


Figure 2:

Figure two, displays the attacker logging into Rickdiculosly easy, using ftp, and downloading a file titled FLAG.txt.

Once the flag has been obtained it can be viewed via the cat command (cat FLAG.txt).
Important note: make sure to keep the captured flags, and any information gathered about the target in a separate directory. Being organized will save time and is vital for successful exploitation of this machine.

Referring back to figure one reveals that the target machine is running an Apache server on port 80. Navigating to the webpage using Firefox yields the following.

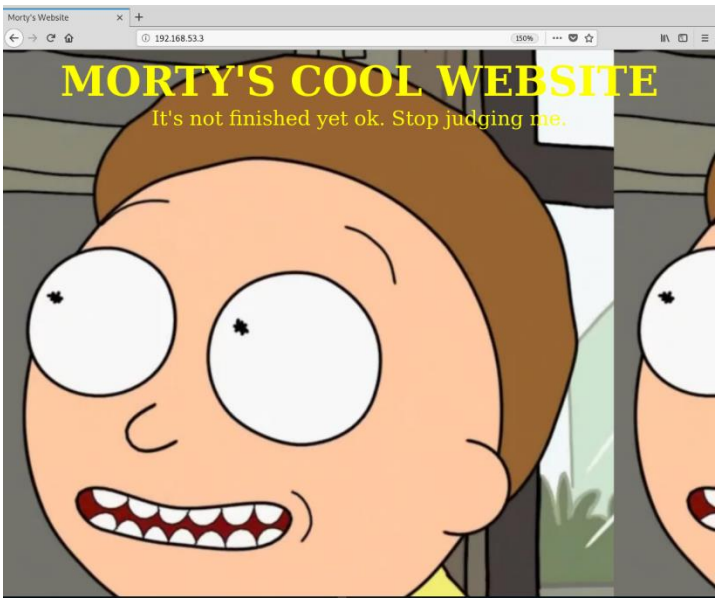


Figure 3:

Figure three, shown on the left, shows the homepage for Rickdiculosly easy. As is noted in the figure the webpage is a work in progress. Since the site is unfinished there is a high probability that security is minimal, and bugs are prevalent.

At first glance there appears to be nothing of interest here. There are no links to other pages and no useful information is being displayed. However, further inspection (using dirbuster) reveals some hidden directories housed within the webserver.

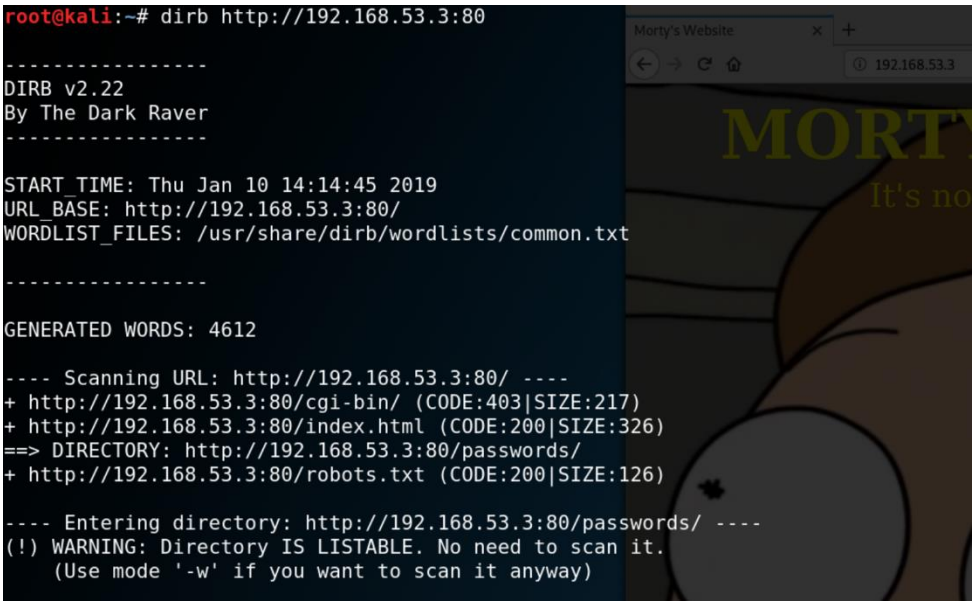


Figure 4:

Figure four, shown on the left, reveals several hidden directories that exist on Rickdiculosly Easy's webserver. Looking at the results reveals a file called passwords and a file called robots.txt. The passwords file is extremely interesting because there is a possibility that actual passwords are being stored there! The robots.txt file is a file that is used by webservers to tell Google, and other browsers to ignore the files listed within the robots.txt folder.

As can be seen in the scan above the target contains a directory called passwords, as well as, a directory titled robots.txt. The passwords directory contains a FLAG and a file called passwords.html. Here are the contents of the FLAG file.

```
FLAG{Yeah d- just don't do it.} - 10 Points
```

The passwords.html file contains a note addressed to Morty chastising him for storing his password in a file called passwords.html (real clever Morty). The note also mentions that the password was hidden. Inspecting the webpage (right click and select inspect element) reveals a comment that contains the password winter.

Figure 5:

Figure five is shown on the left and contains a note to a person named Morty. Part of the note has been edited to hide vulgar language.

Store the obtained password somewhere safe. Dirbuster also revealed the presence of a file called robots.txt. Robots.txt is commonly found on webservers and is used to inform the web browser what directories it is not allowed to access. Checking the robots.txt file uncovers a list of other files on the system.

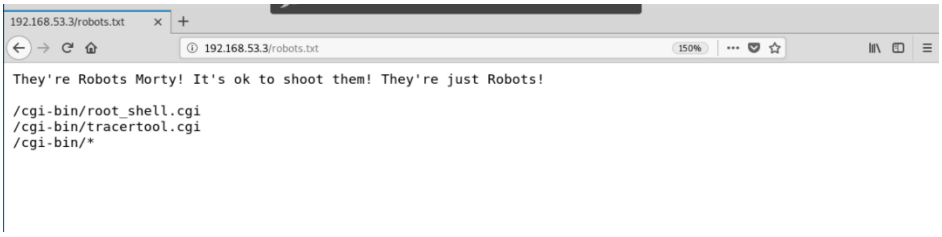


Figure 5.5:

Figure 5.5 displays the contents of robots.txt

Exploiting Morty's Machine Tracer Machine

The robots.txt file seems to contain paths to applications created by the site administrators. The /cgi-bin/root_shell.cgi looked promising but the shell provided by the webserver ended up

being unusable (was not set-up yet). Moving on to the next item in the list, the /cgi-bin/tracertool.cgi reveals an application that is vulnerable to command injection (see figure 6).

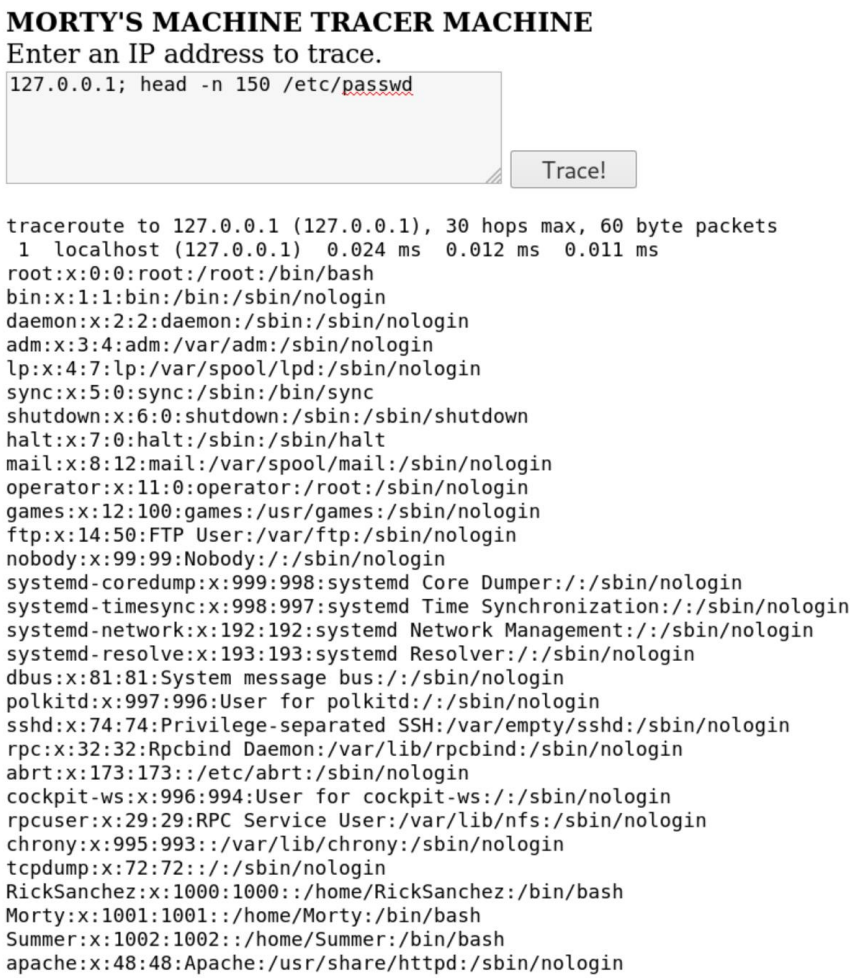


Figure 6:

Figure six displays the abuse of Morty's program, which was designed to trace the path to a provided IP address. However, due to the lack of input control, the program was made to print the contents of the /etc/passwd file, which holds the information about the users on a Unix system. The passwords are stored in a separate file called /etc/shadow, which Morty's program does not have permission to view. The usernames that exist on the system can be seen on the far left. Users that can be accessed include root, Morty, Summer, and RickSanchez.

Thanks to Morty's "Machine Tracer Machine" program the contents of the /etc/passwd folder, which resides on the victim's machine is now able to be read. Unfortunately, the passwords in /etc/passwd are not visible to us and the tool shown above does not allow us to view /etc/shadow, which would have provided us a list of encrypted passwords used on the machine. However, since the password for one of the users is already known a brute force attack could prove quite effective. Attempting to login via the SSH service running on port 22 of the target machine (ssh -l username ip address) reveals that the SSH service running on port 22 is not working.

Exploring Other Options

Referring to the Nmap scan results, shown in figure one, reveals that port 9090 is also open and is running webservice called cockpit.

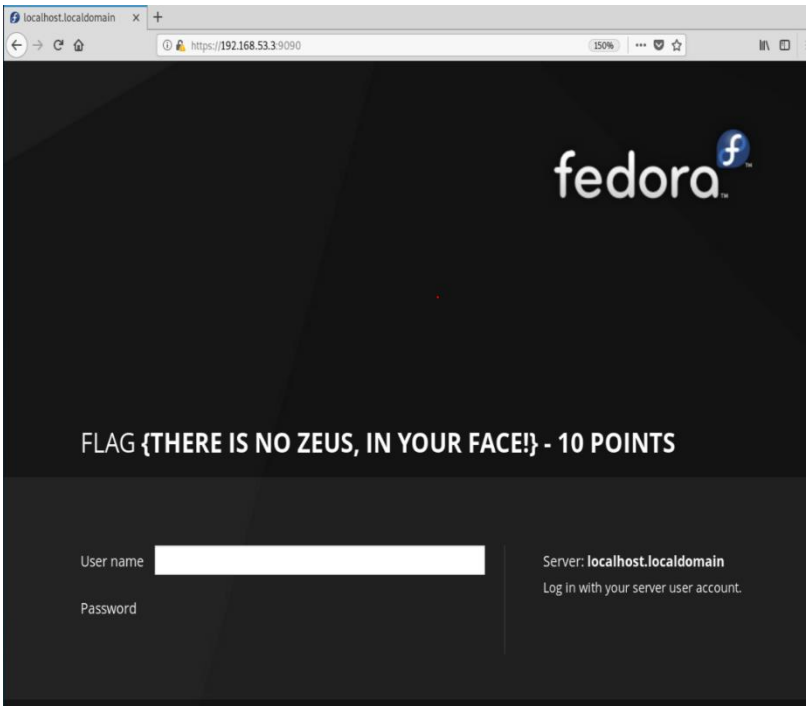


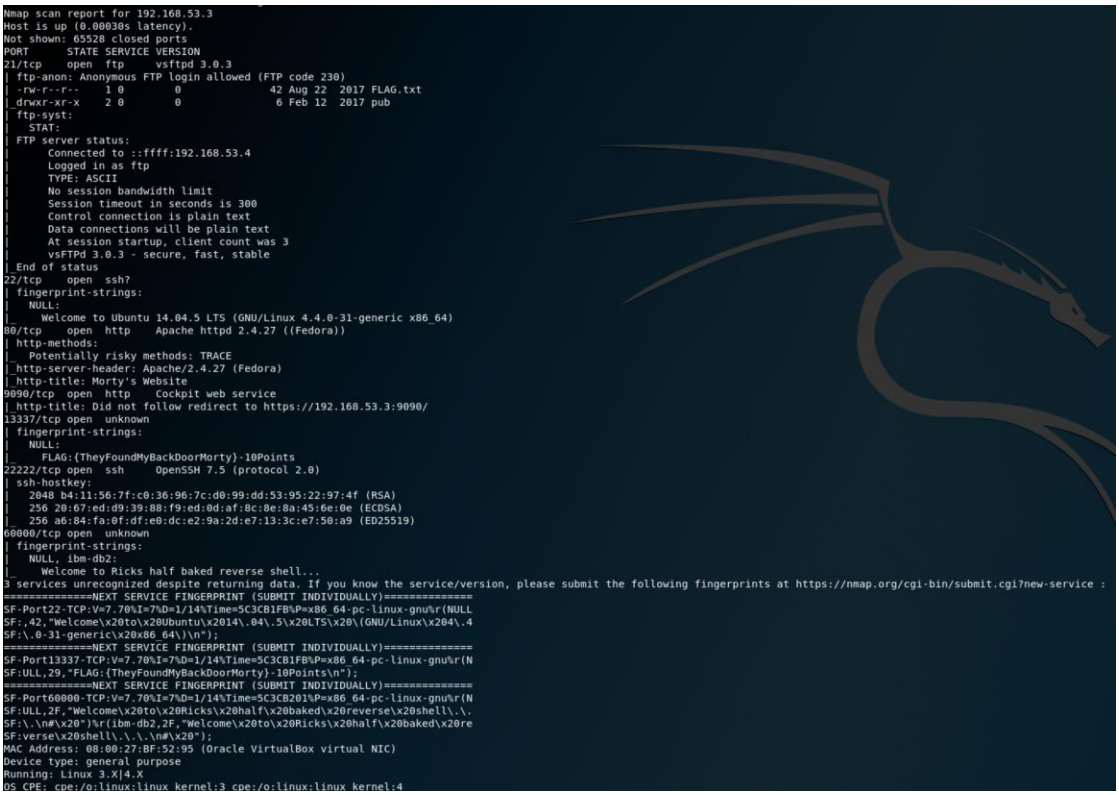
Figure 7:

Figure seven, shown on the left, shows the attacker accessing another website housed on the victim's machine.

Excellent! The server on port 9090 contains a flag and what appears to be a login forum. Unfortunately, the login form has not been setup yet. Since there are no more services to probe (at least not in the original Nmap scan) another, more in-depth, NMAP scan must be run against the target. The last Nmap scan checked the top 1000 commonly used ports; however, there are 65535 TCP ports on a computer (TCP refers to Transmission Control Protocol). The image, shown on the next page contains the result of the full port scan. To reproduce these results run nmap -A -T4 -p 1-65535 192.168.53.3 (replace the IP address with the address of the target).

Figure 8:

Figure eight (displayed below) shows the results of the full port scan. The ports 22222, 13337, and 60000 were not uncovered by the original port scan, which is shown in figure one



The results of the above scan reveal several other opened ports. Of interest is the service running on port 13337. We can use Netcat to make a connection and see what is there.

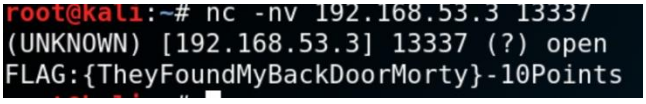


Figure 9 (see left): displays the results of the netcat command. Netcat creates a connection from one computer to another computer using the port specified as an argument.

Port 60000 is also interesting because the service is unknown. Attempting to use Netcat to connect to port 60000 yields a reverse shell and a flag. The reverse shell appears to have root privileges (but after further exploration is just an unescapable void).

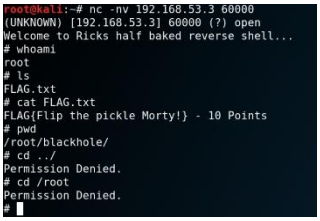


Figure 10:

Figure ten shows the shell that was obtained when Kali Linux connected to port 60000.

Assembling the Pieces and Gaining Access

So far, we have captured four flags, obtained a list of known usernames, found a password, and gained access to a portion of the target machine. Let's attempt to login using the password that was obtained earlier (winter). The tool Hydra, which is a brute force password cracker, could be used here; however, there are only four accounts on the target machine so it is easier to simply attempt to login to each account using the password winter and the usernames obtained from Morty's Machine Tracer Machine (refer to figure 6).

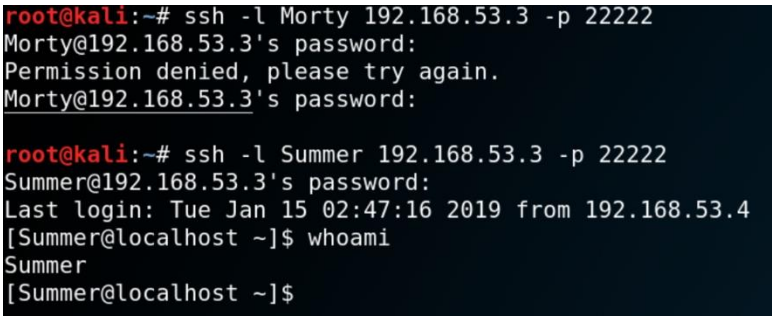


Figure 11:

Since there are only four user accounts to try the attacker opted to perform this phase by hand, which should be done when possible because password crackers tend to cause logfile expansion, which could bring the attack to the attention of an administrator.

Success! The user Summer is using the password winter. Performing the head command, we can see the contents of another flag (shown below).

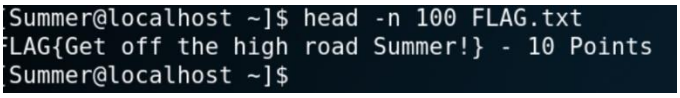


Figure 12: shown on the left, unearths the fifth flag

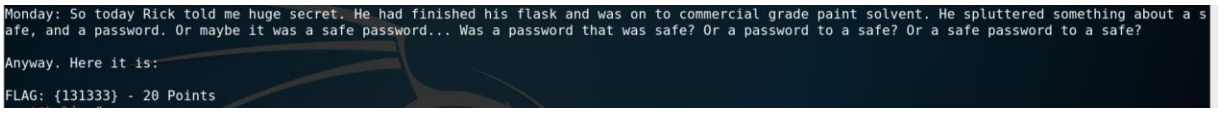
Expanding Influence

We have successfully gained basic access to Rickdiculously Easy. However, to be successful full root privileges should be obtained. Changing to the parent directory (cd ../) reveals the home directories for Rick, Morty and Summer. Morty's directory contains two files journal.txt.zip and Safe_Password.jpg. Both files sound promising. Netcat can be used to move the Safe_Password.jpg file to the local machine.

Figure 13 (shown on the next page): The thirteenth figure shows the attacker transferring a picture from the victim's machine to the attacking machine. The attacker's machine is shown on the left and the victim's machine is shown on the right. NOTE: since the attacker successfully logged into the victim's machine, he has access to it. On the right the attacker is searching for useful information that may be stored within the file Safe_Password.jpg.

The strings command can now be used to parse all strings that are at least 4 letters long from the .jpg file. The output of the strings command reveals the password for journal.zip. Moving the journal back to the attacking machine and unzipping it (with the password Meeseek) reveals another flag and additional information about the user, Rick.

Figure 14: Shown below displays another one of Morty's notes, which contains a hint pertaining to Rick's password.



Moving to Rick Sanchez's directory and performing the ls command yields another flag.

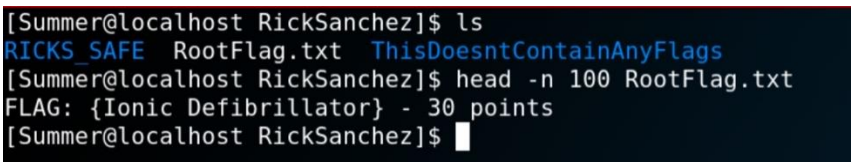


Figure 15: Shows a flag contained in Rick's home directory. It should be noted that the privilege settings for the users on this system has made it easy to find additional information.

The "ThisDoesntContainAnyFlags" folder seems to be another one of Rick's jokes (it does not contain a flag). This leaves RICKS_SAFE (sounds promising). The directory contains a file called safe, which is unable to run from Ricks directory. Luckily, Netcat can be used to transfer the file directly to the attacking machine. This was by far the most difficult part of the machine. Attempting to run the safe program on Kali Linux produced an error as expected. The file command shows that the file safe is an ELF 64-bit executable file for GNU/Linux 2.6.32(This type of file can only run on the machine that it was compiled on). Using FTP, ftp 192.168.53.3, the file, safe, was placed in Summer's home directory and successfully executed.

This program appears to contain a note to remind the user to use command line arguments. After some time, it became clear that the command line argument required was a password, which was provided by the note in Morty's journal, see figure 14. Providing the password as a command line argument yields another flag containing a hint:

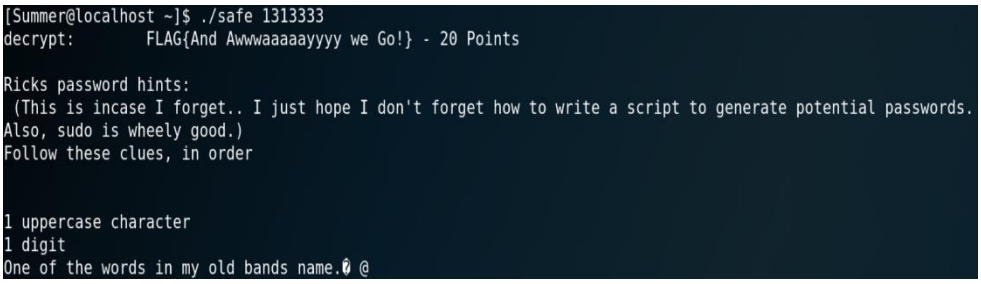


Figure 16:

Figure 16 displays the output of Rick's "safe" program after the correct password is provided.

The tool, crunch, can be used to generate a list of passwords that follow the set of rules provided in the clue (Rick's old band name is The Flesh Curtains, which can be found with a google search). After the list is generated Hydra will have no problem cracking the password to Rick's account. The output of both tools can be viewed below.

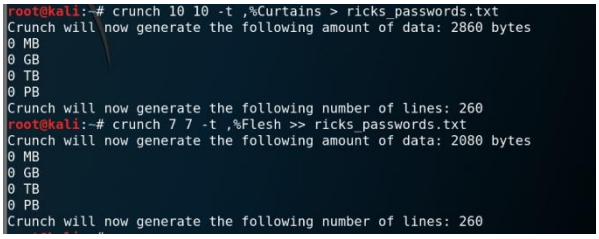
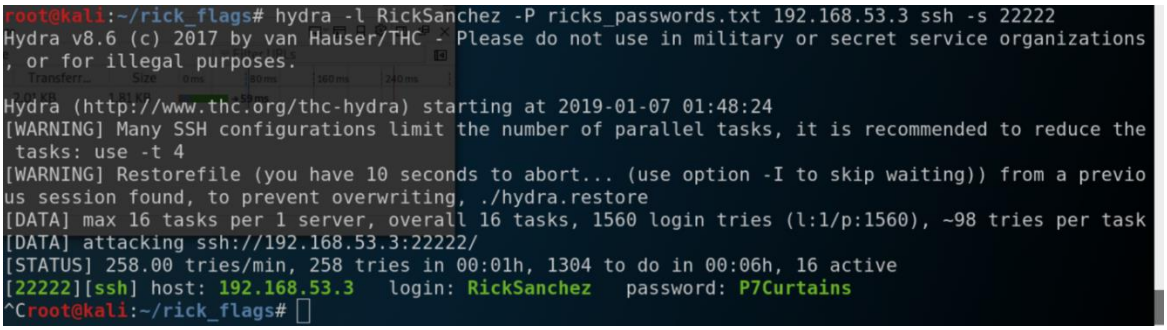
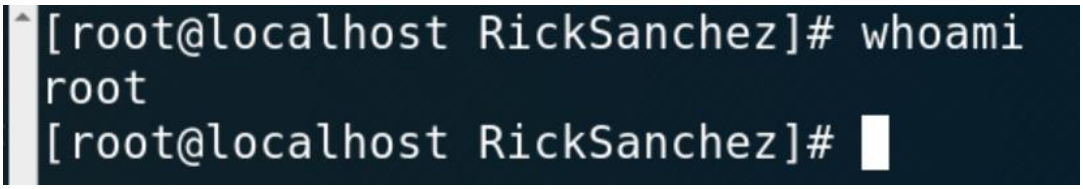


Figure 17/18: Figure 17, shown on the left, shows correct usage of the crunch utility to create two wordlists. The wordlists contain words made from one capital letter, a number and a single word from Rick's old band. Figure 18 is shown below. Figure 18 shows the tool hydra finding the password to Rick's account using the wordlists created in figure 17.



Rick does not have root privileges, but he is a sudo user. Since being a sudoer grants root privileges we can use the sudo command along with useradd to create a new root user on the system. The command required is sudo useradd -o -u 0 -g 0 <username>. The -o allows multiple users with the same user ID (UID) to be created. This is useful because the root user has a UID of zero. The -u and -g options are for setting the future users UID and GID (User ID and Group ID). Since we want the new user to have root privileges, we will set them both to zero. Once this is done use the passwd command to set a password and the su command too switch to the new account.



Turns out there are no additional flags in the root directory, but its always nice to have unrestricted access to a machine.