**Assignment 3**

**Issiah Deleon**

## Abstract

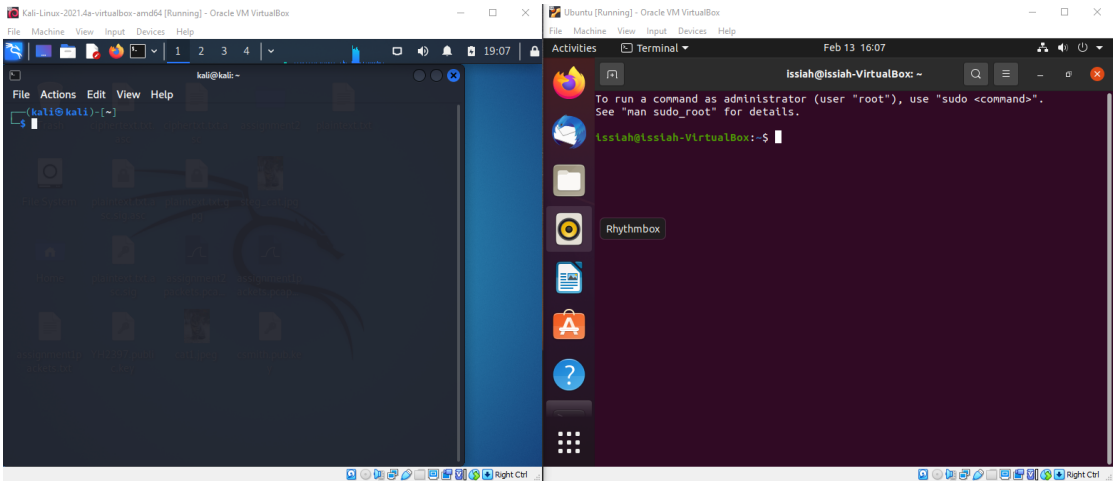
The purpose of this lab is to perform a brute force password attack against an SSH server using software called Hydra. This lab will help us better understand the vulnerability of passwords contained in “known password” lists which are particularly vulnerable to such brute force attacks.

## Introduction

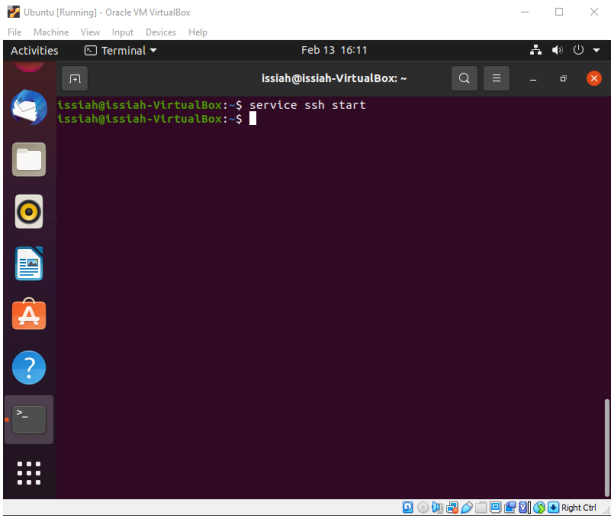
A virtual Kali Linux machine and a virtual Ubuntu machine will be used with Oracle VM VirtualBox Manager. The Kali machine will be the attacking system while the Ubuntu machine will serve as the target system. Wireshark will be used to display the capturing of packets. To perform the brute force attack, Hydra will be used while utilizing two different wordlists filled with potential passwords. SSH is as a network communication protocol that will enable the Ubuntu and the Kali Linux machines to communicate.

## Summary

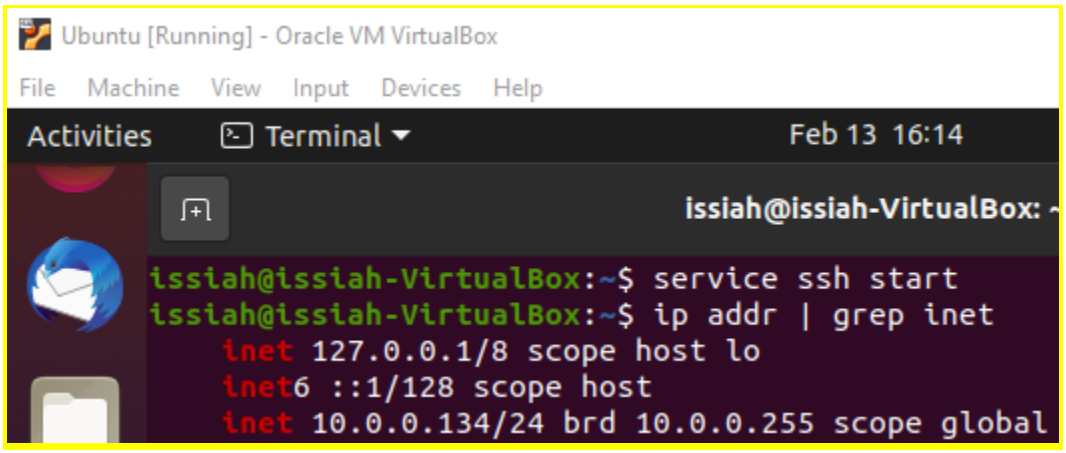
Begin by opening two virtual machines. In this demonstration we use an Ubuntu machine and a Kali Linux machine.



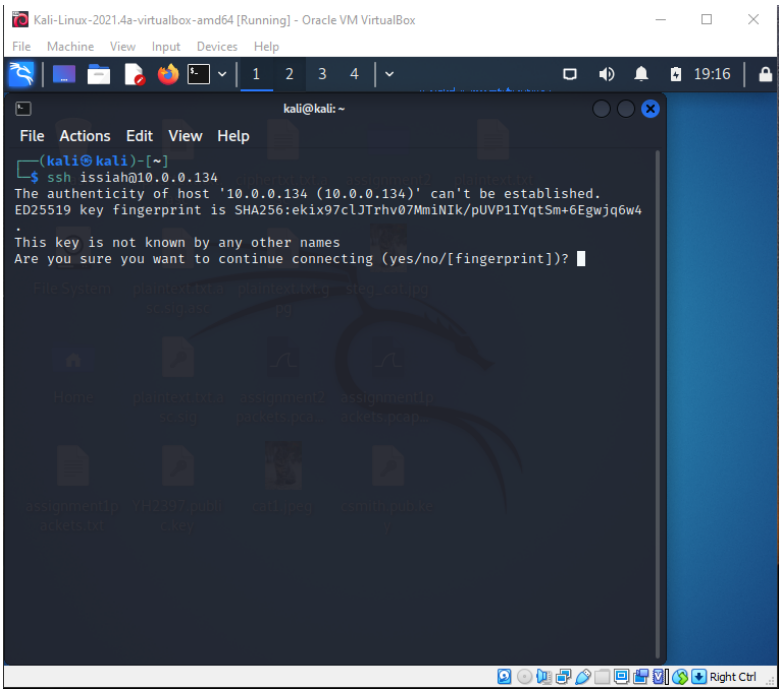
On the Ubuntu machine, open up a terminal window and type in the following command: service ssh start. Now SSH is running.

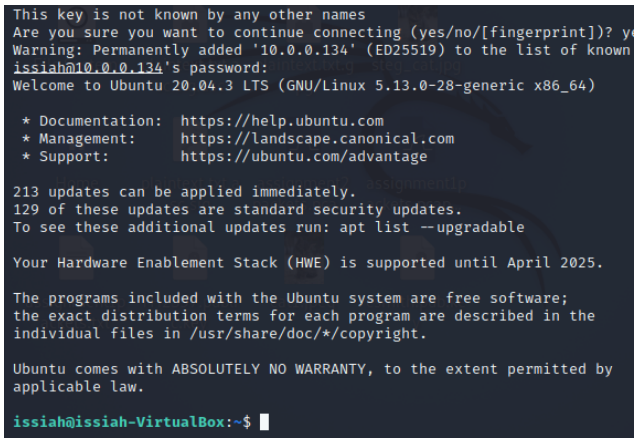



We will then need to determine the IP address of the Ubuntu system. To do so, enter the command: ip addr | grep inet. The IP address of our Ubuntu system is 10.0.0.134.

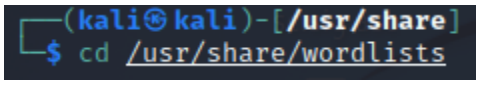


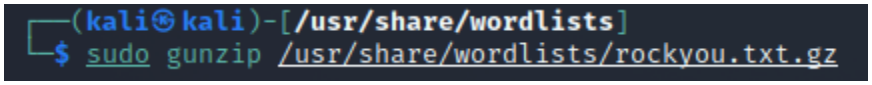
Now on the Kali Linux machine, open up a terminal window and type in the command: ssh userid@ipaddress. You will be prompted to confirm the connection request and enter the password of the Ubuntu system.



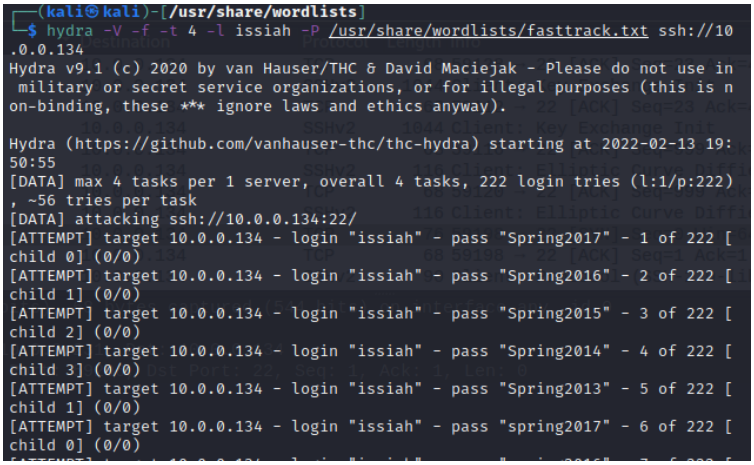

Next we begin the process of brute force attacking. In our Kali Linux machine, logout from SSH using “logout” and then enter the command: cd /usr/share/wordlists. This will change our directory to the directory containing the word list used in the brute force attack. After changing the directory, we need to unzip the rockyou.txt.gz file. To do so, enter the following command: sudo gunzip /usr/share/wordlists/rockyou.txt.gz.



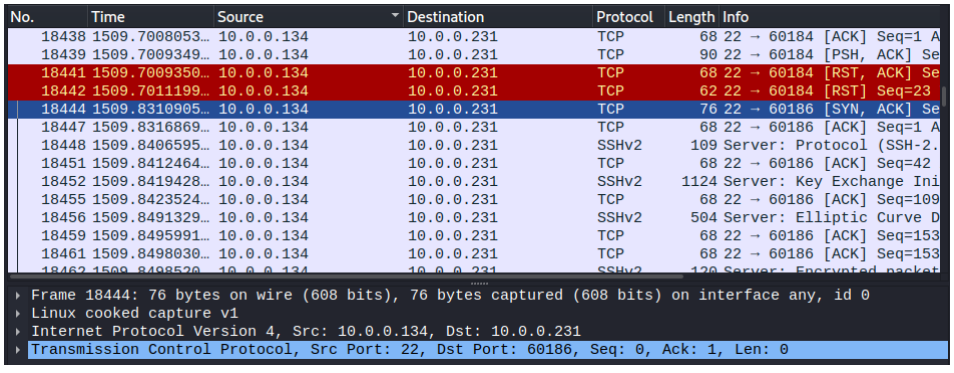




After unzipping the rockyou.txt file, we are ready to begin the brute force attack. In your Kali Linux machine, enter the following command: hydra -v -f -t 40 -l isaiah -P /usr/share/wordlists/fasttrack.txt ssh://10.0.0.134. This will begin the process of brute force

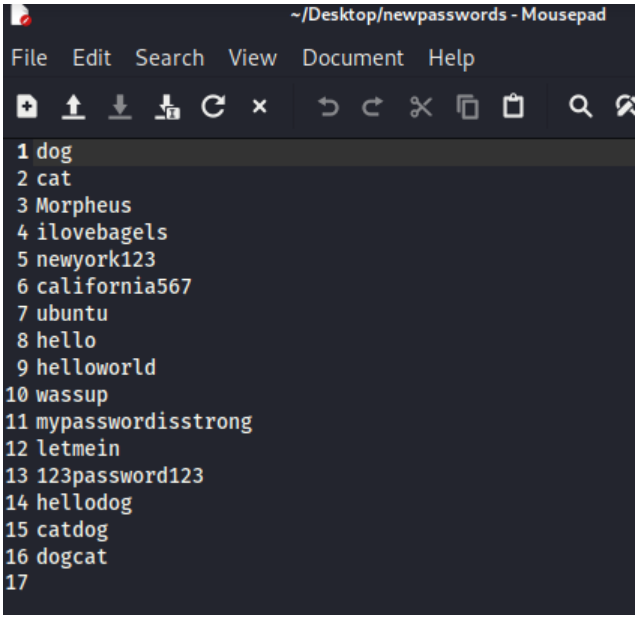
attacking, and 4 words will be attempted at once in a “multithreaded” attack that attempts numerous passwords at the same time. This uses a smaller default list of passwords than rockyou.txt.



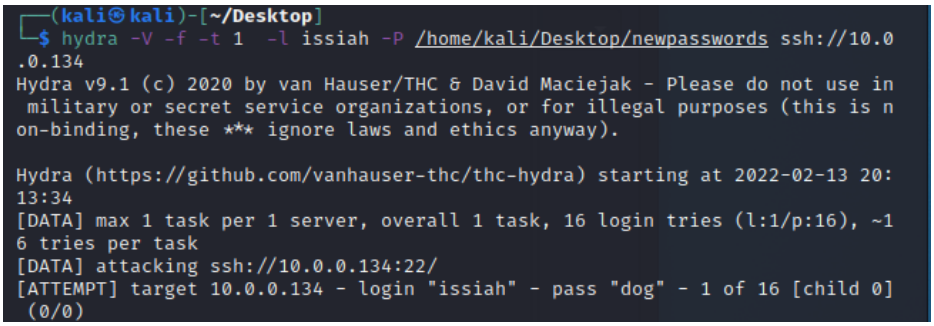

If we run wireshark on our Kali Linux machine while performing the hydra brute force attack, we can see these “attempts” packets as those with a source ip address corresponding to the id address of the attacking machine (Kali Linux) and the destination ip address having an ip address corresponding to the ip address of the “victim” machine (Ubuntu). When the Hydra attack using the fasttrack.txt file ends, we take note of the fact that it was an unsuccessful brute force attack, with 0 valid password matches.



Now we will perform the brute force attack again with a custom made list. Right click the Kali Linux desktop screen and select “New Document”. This document will hold our arbitrary passwords. Fill the file with at least 15 fake passwords, including the actual password of your Ubuntu system.




In your Kali Linux terminal, change directories to the one containing this new custom file. In our case, this is: cd /home/kali/Desktop. After running that command, enter this one to begin the new hydra brute force attack on our Ubuntu system: hydra -V -f -t 4 -l isaiah -P /home/kali/Desktop/filename ssh://10.0.0.134



```
[22][ssh] host: 10.0.0.134   login: issiah   password: ubuntu
[STATUS] attack finished for 10.0.0.134 (valid pair found)
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2022-02-13 20:
14:20
```

After this new brute force attack runs, we notice how Hydra notifies us of a successful password find. If we “Stop” our packet collection in wireshark immediately after Hydra notifies us of a successful login attempt, we can narrow down which packets are associated with our brute force attack. As we can see below, the source ip address matches with the ip address of our attacking Kali linux machine, and the destination ip address matches up with our Ubuntu “victim” machine.

| o. | Time | Source | Destination | Protocol | Length | Info |
|---|---|---|---|---|---|---|
| 30585 | 3660.6919721… | 10.0.0.134 | 10.0.0.231 | TCP | 68 | 22 → 60846 [ACK] Seq |
| 30586 | 3660.6977664… | 10.0.0.134 | 10.0.0.231 | SSHv2 | 504 | Server: Elliptic Cur |
| 30589 | 3660.6982699… | 10.0.0.134 | 10.0.0.231 | TCP | 68 | 22 → 60846 [ACK] Seq |
| 30591 | 3660.6986136… | 10.0.0.134 | 10.0.0.231 | TCP | 68 | 22 → 60846 [ACK] Seq |
| 30592 | 3660.6986755… | 10.0.0.134 | 10.0.0.231 | SSHv2 | 120 | Server: Encrypted pa |
| 30595 | 3660.6989304… | 10.0.0.134 | 10.0.0.231 | TCP | 68 | 22 → 60846 [ACK] Seq |
| 30596 | 3660.7082437… | 10.0.0.134 | 10.0.0.231 | SSHv2 | 120 | Server: Encrypted pa |
| 30598 | 3660.7085459… | 10.0.0.134 | 10.0.0.231 | TCP | 68 | 22 → 60846 [ACK] Seq |
| 30606 | 3663.0421486… | 10.0.0.134 | 10.0.0.231 | SSHv2 | 120 | Server: Encrypted pa |
| 30608 | 3663.0424892… | 10.0.0.134 | 10.0.0.231 | TCP | 68 | 22 → 60846 [ACK] Seq |
| 30609 | 3663.0522231… | 10.0.0.134 | 10.0.0.231 | SSHv2 | 104 | Server: Encrypted pa |
| 30611 | 3663.0958435… | 10.0.0.134 | 10.0.0.231 | TCP | 68 | 22 → 60846 [ACK] Seq |
| 30612 | 3663.3039950… | 10.0.0.134 | 10.0.0.231 | TCP | 68 | 22 → 60846 [FIN, ACK |

**Summary**

In conclusion, we have demonstrated the inherent vulnerability of weak passwords when said weak password encounters the forces of a brute force attack. Over the years attackers can gather larger and larger lists of compromised passwords and use them in these kinds of brute force attacks to break into protected accounts. This is why it is integral to create safe and secure passwords of our own as our passwords are typically the only line of defense against these kinds of attacks.

Our attempt at brute forcing ourselves into our Ubuntu system was successful, but only because we included its password in our second custom list of passwords. The advantages of this

kind of attack are that we can use massive lists of compromised passwords and quietly brute force our way into the system in the background. If we take our time and do this nice and slow, the victim system should not detect that we are attempting to break in. The disadvantages of this kind of attack are that since we are using lists of compromised passwords, it can take a very long time for our hydra attack to run through the entirety of the list. This is not a fast or quick process, and strong passwords are unlikely to be contained within these compromised password lists.

According to the Hydra manual, hydra can also perform brute force attacks through avenues other than SSH, such as SSL, or Secure Sockets Layer. This is a protocol that links networked computers.

We can prevent these kinds of attacks by simply using secure and safe passwords. These passwords will contain special characters, numbers, letters, and uppercase letters. As well as using strong passwords, users should change these passwords on a frequent basis to avoid their password being compromised and added to a list of compromised passwords.

Hydra was the main tool used in this lab. As such, Hydra does not offer a form of authentication, access control, data confidentiality, or data integrity. It does offer non-repudiation by providing a sort of “attack receipt” following an attack, providing the name of the user who mounted the attack and the time that it completed as well as the host’s ip address and the port with which they were connected to. This makes it hard to deny that an individual ever used Hydra.