# Problem 3 (30)

Banner Grabbing involves the process of connecting to and communicating with a server to potentially gain information about the type of server and services being provided. It can be used by White, Black and Gray hat hackers to provide useful inform that could lead to potential exploit(s).

During a user’s usual interaction of these services these bits of information are hidden from the average user. In this section we will see how it could be possible to extract useful information.

Try the following to get an idea of the tool to be built.

## Testing Ground:

1. Install openssh-server

**sudo apt install openssh-server**

1. Run openssh-server

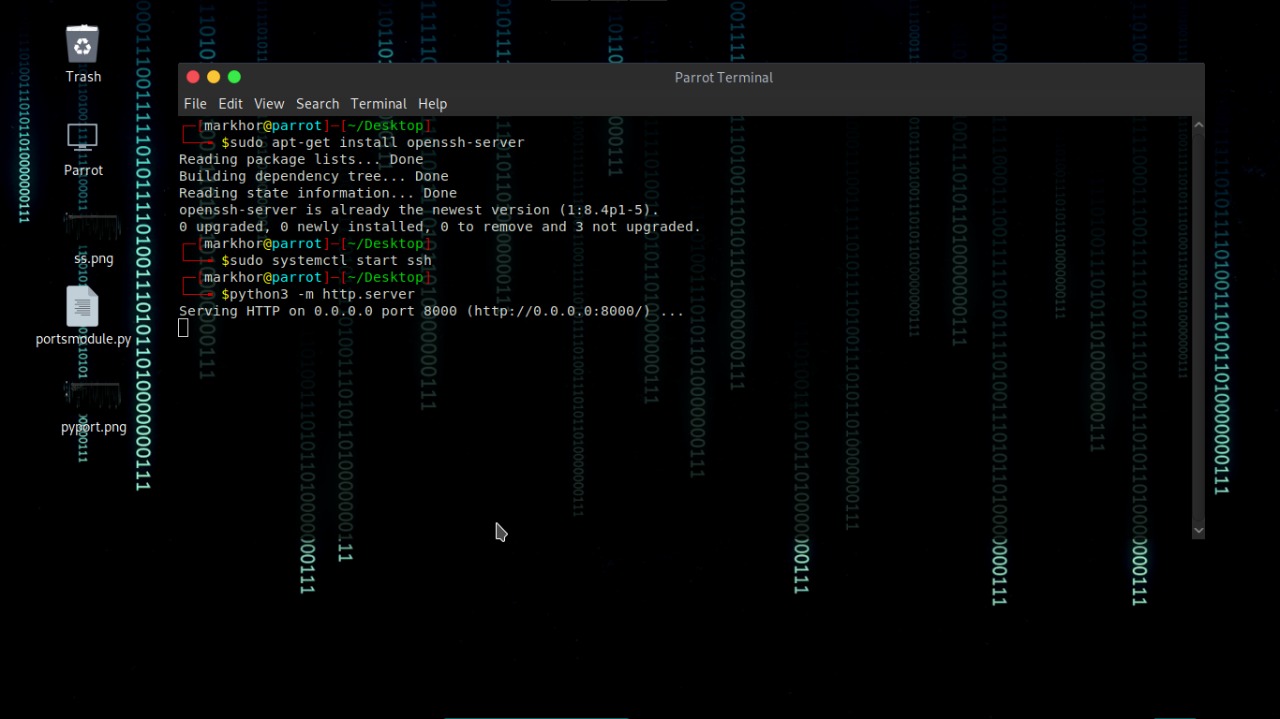
**sudo systemctl start ssh**

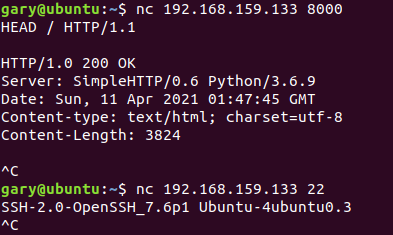
1. Start python http server

*gary@ubuntu:~$* **python3 -m http.server**

*Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000/) ...*

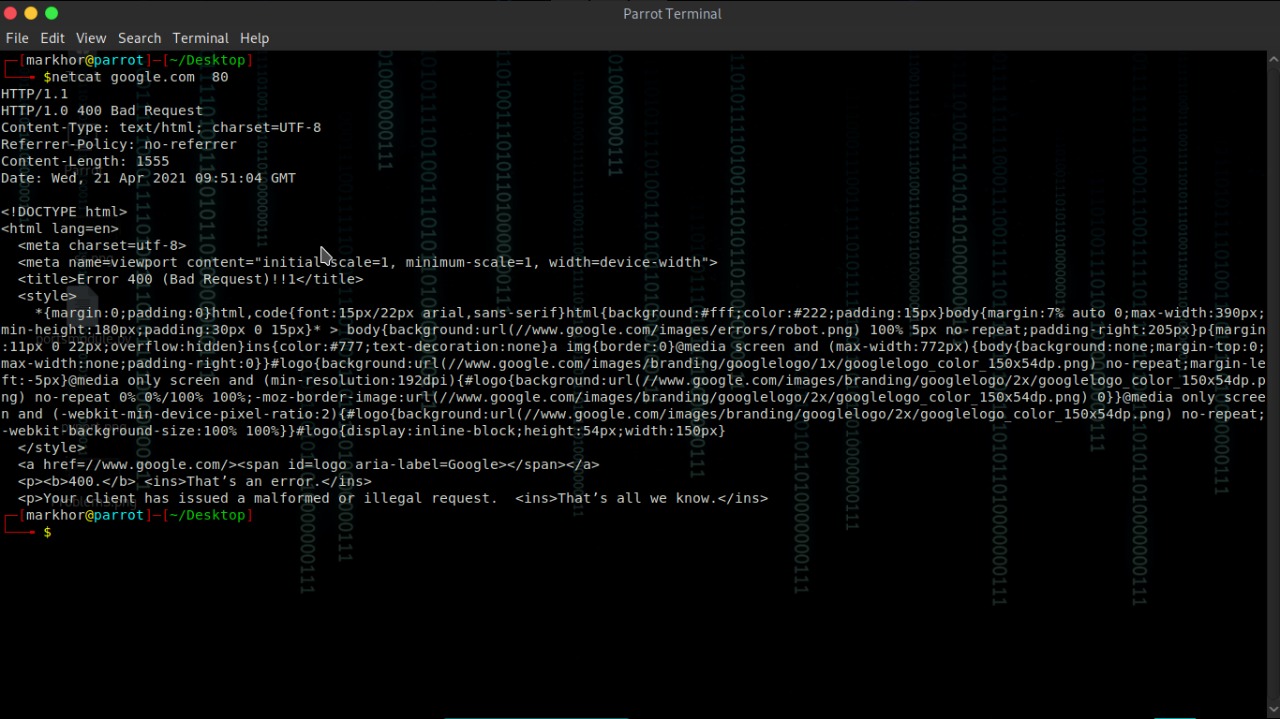
**Ans 1-3:**



1. Get Banner with netcat for both services.

**Ans:**

Following is the picture of getting banner by using netcat.



1. If you are not using kali for this lab you must setup **searchsploit** locally on your VM (takes about 200MB)
2. **cd ~**
3. **git clone https://github.com/offensive-security/exploit-database.git**
4. **sudo apt update**
5. **cd ./exploit-database**
6. **sudo mkdir /opt/exploitdb**
7. **sudo cp files\_\*.csv /opt/exploitdb**

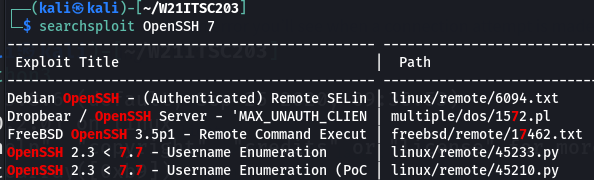
**Ans:**

I am using Parrot Linux in which all these tools are pre-installed, so there is no need of performing these commands.

1. Manually check to see if there’s an exploit using a tool like searchsploit

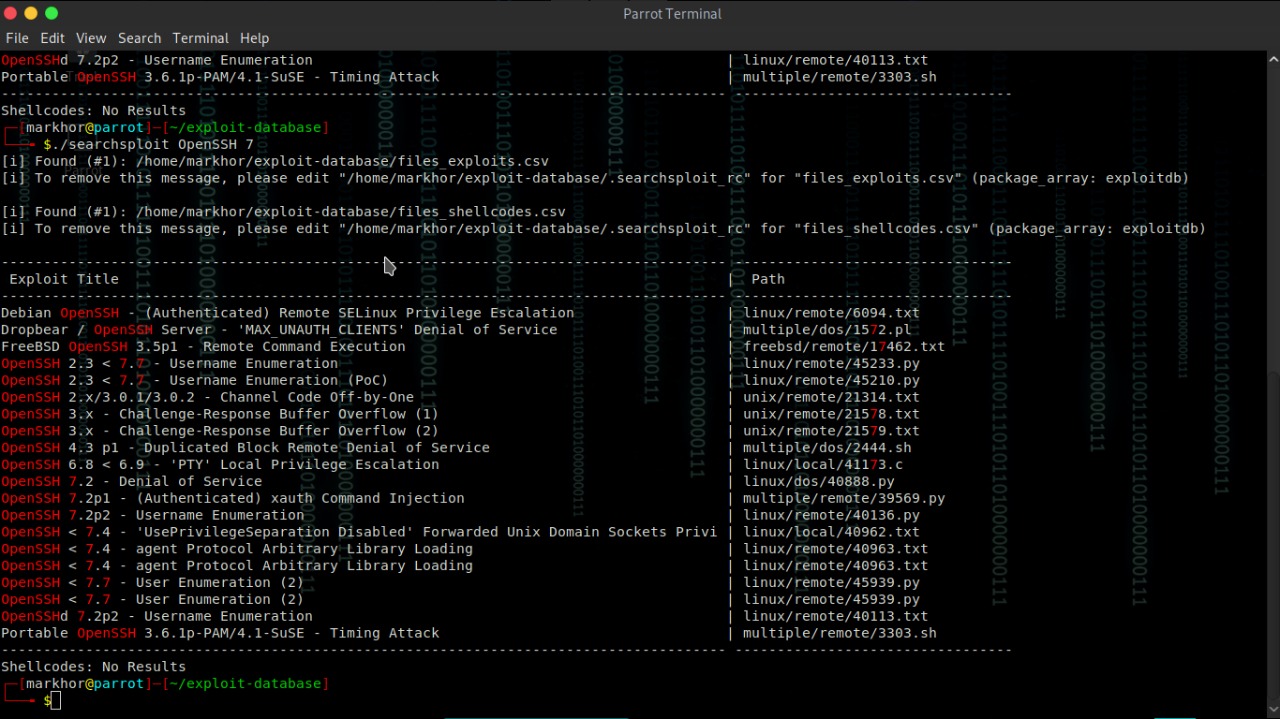
**./searchsploit OpenSSH**

Are there any services available for you to enumerate?



**Ans:**

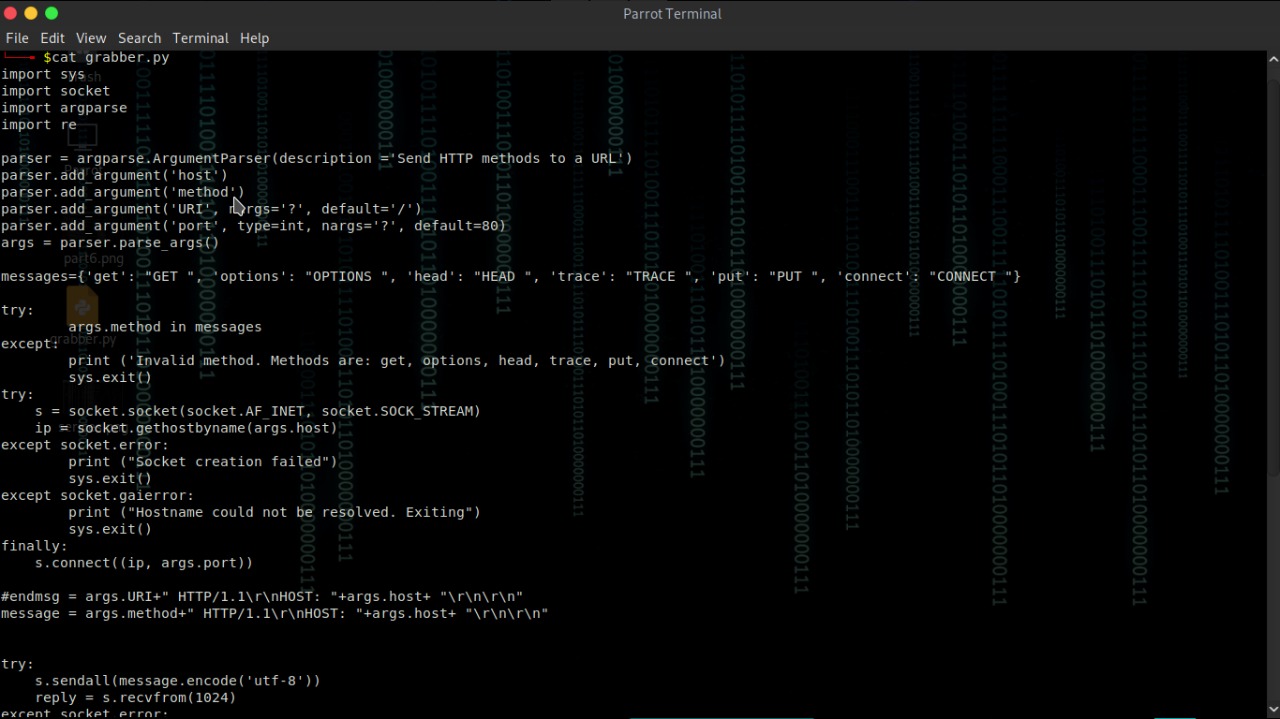
Following are the exploits available in searchsploit.



## **Python Banner Grabber**

1. The tool will go through the entire list of open ports and grab the banners
2. Find all the running services already completed in a previous Problem

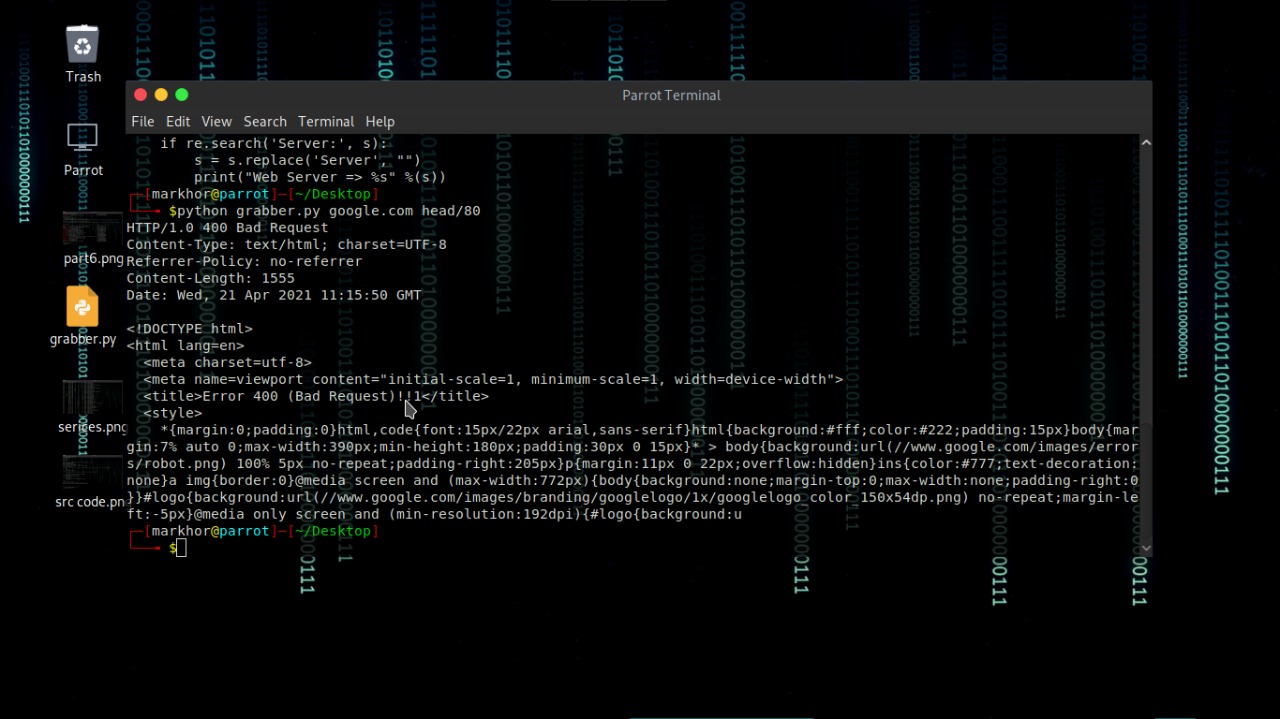
**Ans 1 and 2:**



1. **IMPORTANT**: If the port is an HTTP port; you will need to send the server a request before it will respond. Use the **HEAD** command.

**Ans:**

Requesting a server in HTTP port using HEAD command:



4-5. Print a list of services you found, the port and IP Address that they were found on. Example:

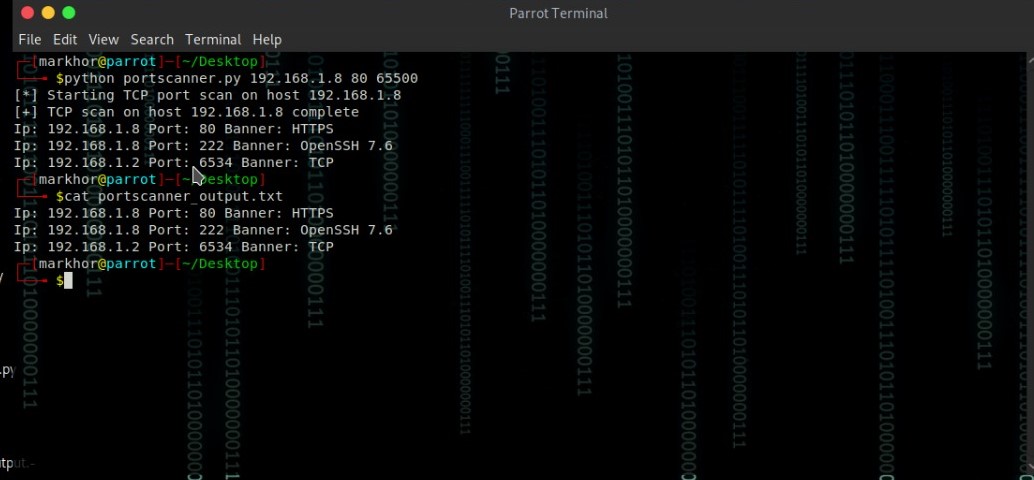
**10.0.2.5 – 22 - OpenSSH 7.6**

**10.0.2.5 – 2222 - OpenSSH 7.6**

**10.0.2.5 – 5543 – HTTP**

**Ans:**

Here is the Output of Port Scanner.



**THE END**