TryHackMe Journal - Rowan King

Instructions

(1) Review the sample journal entry provided below
(2) Scroll down to find the name of the room you have been assigned/are working on
(Pro Tip: Turn on "Outline View" so you can navigate more easily - go to View → Show Outline)
(3) Complete the required rooms on TryHackMe, compiling notes as you work through the room. This might include:
(a) Commonly used Code/Commands
(b) Definitions/Explanations of important terms and concepts
(c) Screenshots of useful diagrams
(4) Once you've completed the module, capture 2-4 important takeaways.
(5) After you get the hang of things, delete these instructions and the sample you were provided!

Entry 1

Room Name: Linux Fundamentals 1

Date Completed: 4/27/2024
Notes During the Room: The basics of Linux and the CLI

Introduced several commands;

Command	**Description**
echo	Output any text that we provide
whoami	Find out what user we're currently logged in as!
ls	listing
cd	change directory
cat	concatenate
pwd	print working directory
find	search for a specified file or folder
grep	search the contents of files for specific values that we are looking for

As well as Operators:

Symbol / Operator	**Description**
&	This operator allows you to run commands in the background of your terminal.
&&	This operator allows you to combine multiple commands together in one line of your terminal.
>	This operator is a redirector - meaning that we can take the output from a command (such as using cat to output a file) and direct it elsewhere.
>>	This operator does the same function of the `>` operator but appends the output rather than replacing (meaning nothing is overwritten).

Important Takeaways:

Entry 2

Room Name: Linux Fundamentals 2

Date Completed: 4/27/2024
Notes During the Room:
Introduced remote login using the SSH command

SSH *	The syntax to use SSH is very simple. We only need to provide two things: 1. The IP address of the remote machine

	2. Correct credentials to a valid account to login with on the remote machine

Other Filesystem Commands

Command	**Full Name**	**Purpose**
touch	touch	Create file
mkdir	make directory	Create a folder
cp	copy	Copy a file or folder
mv	move	Move a file or folder
rm	remove	Remove a file or folder
file	file	Determine the type of a file

Covered common directories such as /ect, /var, /root, & /tmp

Important Takeaways:

Entry 3

Room Name: Linux Fundamentals 3

Date Completed: 4/27/2024
Notes During the Room:

Learned about nano and vim.

Learned about wget and its variables;

Variable	**Value**
The IP address of the remote system	192.168.1.30
User on the remote system	ubuntu
Name of the file on the local system	important.txt
Name that we wish to store the file as on the remote system	transferred.txt

Learned about running and killing processes

Learned about cron files used to automate tasks;

Value	**Description**
MIN	What minute to execute at
HOUR	What hour to execute at
DOM	What day of the month to execute at
MON	What month of the year to execute at
DOW	What day of the week to execute at
CMD	The actual command that will be executed.

Learned about adding and removing repositories

Learned how to pull system logs

Important Takeaways:

Entry 4

Room Name: Linux Strength Training

Date Completed:
Notes During the Room:
As a security researcher you will often be required to find specific files/folders on a system based on various conditions ranging from, but not limited to the following:

- **filename**
- **size**
- **user/group**
- **date modified**
- **date accessed**
- **Its keyword contents**

Therefore, we can do this using the following syntax:

What we can do	Syntax	Real example of syntax
Find files based on filename	find [directory path] -type f -name [filename]	find /home/Andy -type f -name sales.txt
Find Directory based on directory name	find [directory path] -type d -name [filename]	find /home/Andy -type d -name pictures

Find files based on size	find [directory path] -type f -size [size]	find /home/Andy -type f -size 10c (c for bytes, k for kilobytes M megabytes G for gigabytes type:'man find' for full information on the options)
Find files based on username	find [directory path] -type f -user [username]	find /etc/server -type f -user john
Find files based on group name	find [directory path] -type f -group [group name]	find /etc/server -type f -group teamstar

Find files modified after a specific date	find [directory path] -type f -newermt '[date and time]'	find / -type f -newermt '6/30/2020 0:00:00' (all dates/times after 6/30/2020 0:00:00 will be considered a condition to look for)
Find files based on date modified	find [directory path] -type f -newermt [start date range] ! -newermt [end date range]	find / -type f -newermt 2013-09-12 ! -newermt 2013-09-14 (all dates before 2013-09-12 will be excluded; all dates after 2013-09-14 will be excluded, therefore this only leaves 2013-09-13 as the date to look for.)

Find files based on date accessed	find [directory path] -type f -newerat [start date range] ! -newerat [end date range]	find / -type f -newerat 2017-09-12 ! -newerat 2017-09-14 (all dates before 2017-09-12 will be excluded; all dates after 2017-09-14 will be excluded, therefore this only leaves 2017-09-13 as the date to look for.)
Find files with a specific keyword	grep -iRl [directory path/keyword]	grep -iRl '/folderA/flag'
read the manual for the find command	man find	man find

Note: There are many more useful commands aside from the examples above. If you ever have trouble understanding any of the syntax or getting it to work, head on over to explainshell.com to check the syntax and see how this tool can help you on your journey to Linux greatness.

Further notes: if you do not know already, typing CTRL+L allows you to clear the screen quicker rather than typing 'clear' all the time. Additionally, hitting the up arrow allows you to return to a previously typed command so you do not have to spend time retyping it again if you made an error. Cool. Finally, placing: **2>/dev/null** at the end of your find command can help filter your results to exclude files/directories that you do not have permission to.

You should be somewhat familiar already with working with files. Similar to windows, we can do the following:

- **copy files and folders**
- **move files and folders**
- **rename files and folders**
- **create files and folders**

For a quick recap to train your mental memory on the commands please refer to the below information:

What we can do	**Syntax**	**Real example of syntax**

copy file/folder	cp [filename/folder] [directory] (remember, if the filename/folder name has spaces then you will need to encase the filename with speech marks such as cp "[filename with spaces]" [directory]. This applis to other commands such as mv.)	cp ssh.conf /home/newfolder

move file/folder	mv [filename] [directory]	mv ssh.conf /home/newfolder
move multiple files/folders simultaneously	mv [file1] [file2] [file3] -t [directory to move to]	mv logs.txt keys.conf script.py -t /home/savedWork
Move all files from current directory into another directory	mv * [directory to move files to]	mv * /home/scripts
rename files/folder	mv [current filename] [new filename]	mv ssh.conf NewSSH.conf
create a file	touch [filename]	touch newFile.txt
create a folder	mkdir [foldername]	mkdir newFolder
open file for editing	nano [filename]	nano keys.conf
output contents of file	cat [filename]	cat keys.conf
upload file to a remote machine	scp [filename] [username]@[IP of remote machine]:/[directory to upload to]	scp example.txt john@192.168.100.123:/home/john/
run an bash script program	./[name of script]	./timer
open a file for reading/editing	nano [filename]	nano readME.txt

A few additional things to remember is that occasionally you may encounter files/folders with special characters such as - (dash). Just remember that if you try to copy or move these files you will encounter errors because Linux interprets the - as a type of argument, therefore you will have to place -- just before the filename. For example: cp -- -filename.txt /home/folderExample.

Important Takeaways:

Entry 5

Room Name: Intro to Logs

Date Completed:
Notes During the Room:

Important Takeaways:

Entry 6

Room Name: Wireshark Basics

Date Completed:
Notes During the Room:

Important Takeaways:

Entry 7

Room Name: Wireshark 101

Date Completed:
Notes During the Room:

Important Takeaways:

Entry 8

Room Name: Windows Fundamentals 1

Date Completed:
Notes During the Room:

Important Takeaways:

Entry 9

Room Name: Windows Fundamentals 2

Date Completed:
Notes During the Room:

Important Takeaways:

Entry 10

Room Name: Windows Fundamentals 3

Date Completed:
Notes During the Room:

Important Takeaways:

Entry 11

Room Name: Windows Forensics 1

Date Completed:
Notes During the Room:

Important Takeaways:

Entry 12

Room Name: Windows Forensics 2

Date Completed:
Notes During the Room:

Important Takeaways:

Entry 13

Room Name: Intro to Log Analysis

Date Completed:
Notes During the Room:

Important Takeaways:

Entry 14

Room Name: Splunk Basics

Date Completed:
Notes During the Room:

Important Takeaways:

Entry 15

Room Name: Incident Handling with Splunk

Date Completed:
Notes During the Room:

Important Takeaways:

Entry 16

Room Name: Splunk 2

Date Completed:
Notes During the Room:

Important Takeaways:

Entry 17

Room Name: Splunk 3

Date Completed:
Notes During the Room:

Important Takeaways: