QA Consulting

Linux Fundamentals Project - Specification

TRAINING MATERIALS – Linux Assessment Project

Contacts
team.qac.all.trainers@qa.com

1. Introduction

You are to complete the following exercises without the use of previous exercise files created in training. All the exercises concern Linux scripting, navigation and administration – all of which will have been discussed throughout the course.

The assessment will require proofing of scripts working. This should be done through documentation with the use of screenshots, as well as a supply of all scripts written.

You are allowed to use slides and the internet for this exercise, but you are **not** permitted to use exercise solutions produced throughout the course. You will be creating a new virtual machine dedicated to this assessment and should not have access to the exercise scripts you created.

2. Project Requirements

Below is the set of exercise that you must meet by the deadline of this project:

1. Create a bash script that will do the following

- Create a new directory within your current working directory and name it 'QAC [your first name]'.

- Go into this new directory and create 3 new files. Use a text editor (nano, gedit, etc.) to add some text to one of these files, save and close.

- Create another directory within your QAC directory and name it 'childDir'. Copy the file you created and edited into this new directory within the QAC directory.

2. Once you have ran this script, you should have a number of files and directories created. Continuing from this, complete the following:

- In your second file, write a script that will -

    - Call a function that asks for a number, and prints whether it is odd or even

    - Print the result of the script to a file (i.e. the third file you created)

    - There should be a question asked to the user as to whether they want to delete the file. If they do, remove the third file. If they don't, create/touch the file.

- Change the name of this third file to 'changedFileName' and move it to the home directory.

3. Write a script that will say take a user input for a username, and based on the username provided, will print out their user ID and the groups they are in.

4. Write a script that will list all active processes, then ask the user for a process ID. Based on the process ID provided, stop that process.

5. Create a script that will take any amount of user provided numbers. When the user decides they no longer want to enter any more numbers, print out a list of the numbers in order.

6. Create a script that implements an array, which will hold values from 1 to any given user number. From here, loop through all items in the array. If a number divides by 3, echo "Fizz!" instead of the number. If the number divides by 5, echo "Buzz!"

instead of the number.

7. Create a script that reports information on a user-provided name of a directory. Things to report on include:
    a. How many files are in the directory
    b. When the directory was last edited
    c. The largest/smallest file

8. Write a script that will print out all user ID's and usernames from the relevant file, output them into a new file in your home directory, and delete any files that are not owned by your user or the root user. Format this script so that the output into the new file is in a nice format (separated by hyphens, comments, etc.)

9. There is a Java tar file on the LocalInstall of your host machine. Get this into your virtual machine and install Java onto your VM with the use of this file. Extend upon this by writing a script that automates the install with this file.

10. Write a script that will lock a file based on the name of the file provided by the user. Your script should ensure that the user provided name of the file is owned by the user running the script.

3. Advice

This is quite a difficult project to do without the use of previous training exercises, however, you are allowed access to in-built Linux sources, the slides provided and the internet (the man command is a beautiful thing!).

The requirements are set in a sequential order – it's recommended that you follow this, but it's by no means the forced order of work. Feel free to order your work in whatever way you see fit.

You don't need to screenshot every single step of the exercise. Rather, just take screenshots of an output that would justify meeting the exercise goals.