## Week 5 Homework Submission File: Archiving and Logging Data

Please edit this file by adding the solution commands on the line below the prompt.

Save and submit the completed file for your homework submission.

\*\*\*\*DELETED bonus to avoid confusion\*\*\*

---

### Step 1: Create, Extract, Compress, and Manage tar Backup Archives

1. Command to \*\*extract\*\* the `TarDocs.tar` archive to the current directory tar -xvf TarDocs.tar -C /home/sysadmin/Projects

2.Command to \*\*create\*\* the `Javaless\_Doc.tar` archive from the `TarDocs/` directory, while excluding the `TarDocs/Documents/Java` directory:

sudo tar -cvzf Javaless\_Docs.tar.gz --exclude ~/Projects/TarDocs/Documents/Java ~/Projects/TarDocs

3. Command to ensure `Java/` is not in the new `Javaless\_Docs.tar` archive:

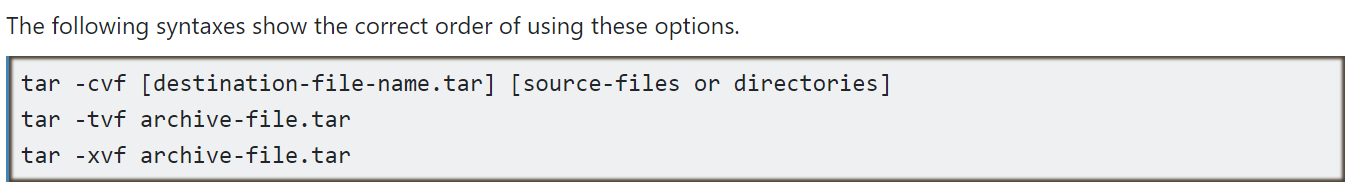
tar tvf Javaless\_Docs.tar.gz | grep -r Java Javaless\_Doc

Note: Files were extracted to a Javaless\_Doc directory before this.

#### Critical Analysis Question

- Why wouldn't you use the options `-x` and `-c` at the same time with `tar`? A user cannot use both the x and c options in the same tar because each option syntax has its own functionality. You cannot systematically create and execute at the same time. That’s like saying you want to turn left and right at the same time. When creating a tar with the “-c” option, the way the command functions is to create the file name/destination(input) to yield an (output) i.e source file or directory (what directories we are backing up).

When using “-x” the new input is the output of the creation stage i.e ***filename. tar*** which then creates a whole new output --- where to extract the file. Takeaway: something must be created first before it can be manipulated, in this case extracted. These events cannot happen at the same time.



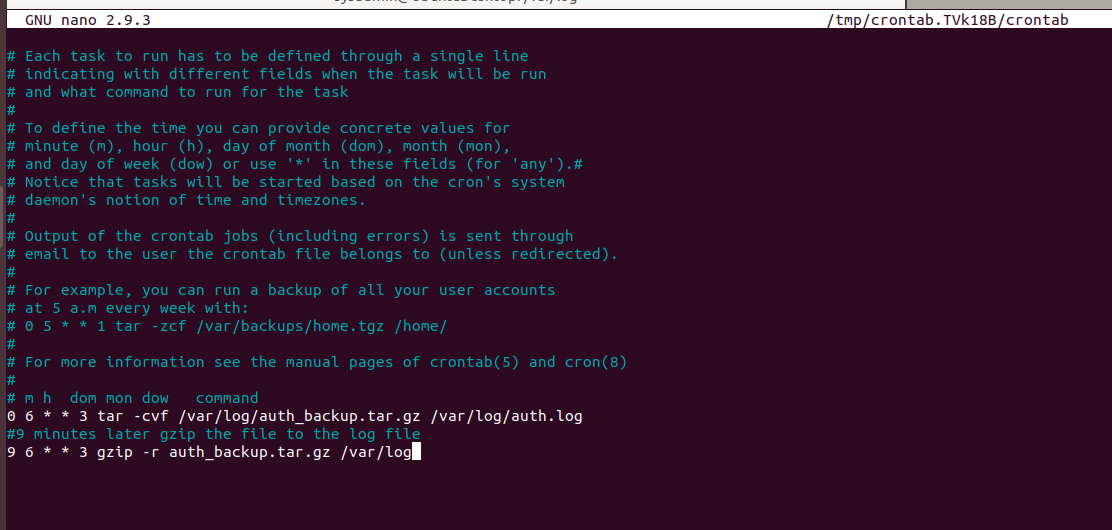
<https://www.computernetworkingnotes.com/linux-tutorials/tar-command-options-and-syntax-explained.html>

\*\* In linux, it's very important to know what directory you are in and where you want to go. Relative and absolute paths are extremely important with all commands in linux. This ensures that commands like tar function properly.

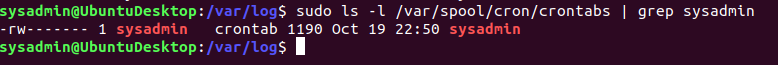
---

### Step 2: Create, Manage, and Automate Cron Jobs

1. Cron job for backing up the `/var/log/auth.log` file:



To confirm the cronjob was made today 10.19.2021



---

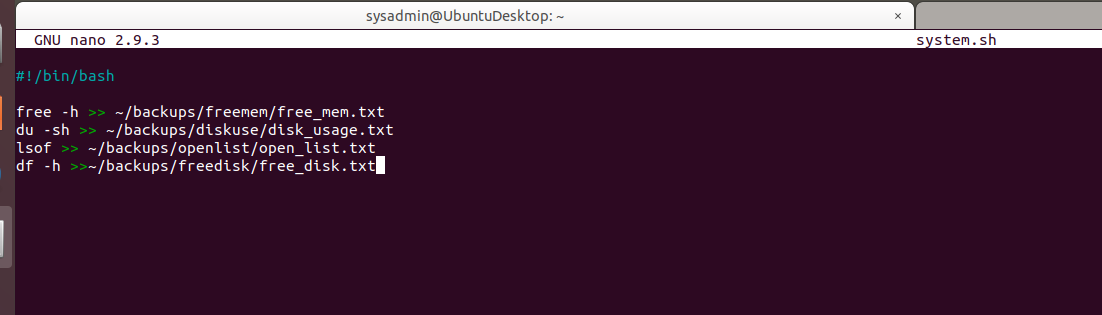
### Step 3: Write Basic Bash Scripts

1. Brace expansion command to create the four subdirectories:

cd ~

mkdir -p backups/{freemem,diskuse,openlist,freedisk}

2. Paste your `system.sh` script edits below:



3. Command to make the `system.sh` script executable:

chmod +x system.sh

\*\*Optional\*\*

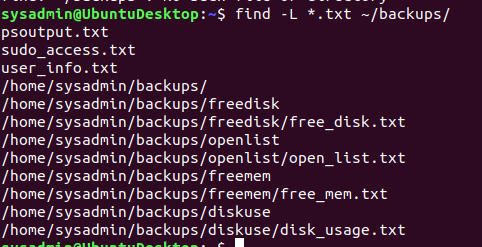
- Commands to test the script and confirm its execution:

./system.sh

You can then CD into each directory to see if the text files were placed in those corresponding directories. Example



Or you can do a find -L \*.txt ~/backups/



---

### Step 4. Manage Log File Sizes

1. Run `sudo nano /etc/logrotate.conf` to edit the `logrotate` configuration file.

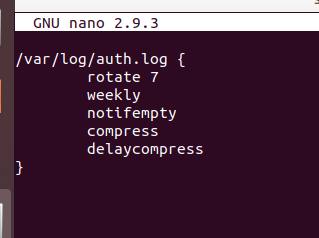
Configure a log rotation scheme that backs up authentication messages to the `/var/log/auth.log`.

- Add your config file edits below:

```bash

[Your logrotate scheme edits here]

```



---

.