Miles Ogrady

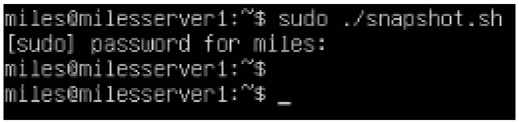
Scripts pt2

Github link: https://github.com/miles5k/scriptspt2

Updates needed on system to run: none

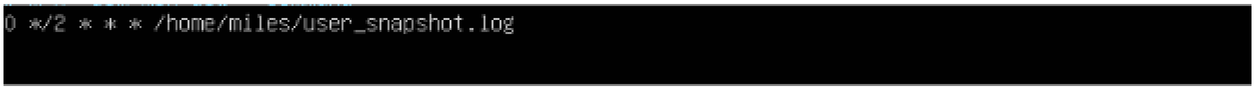
Scripts run in ubuntu:

snapshot.sh: to run this script type sudo ./snapshot.sh



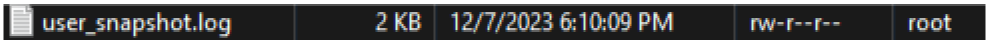
```
miles@milesserver1:~$ sudo ./snapshot.sh
[sudo] password for miles:
miles@milesserver1:~$
miles@milesserver1:~$ _
```

For this script I useda cronjob that makes the snapshot script run every hour by using crontab -e and then writing this in the file and saving it with ctrl + x.



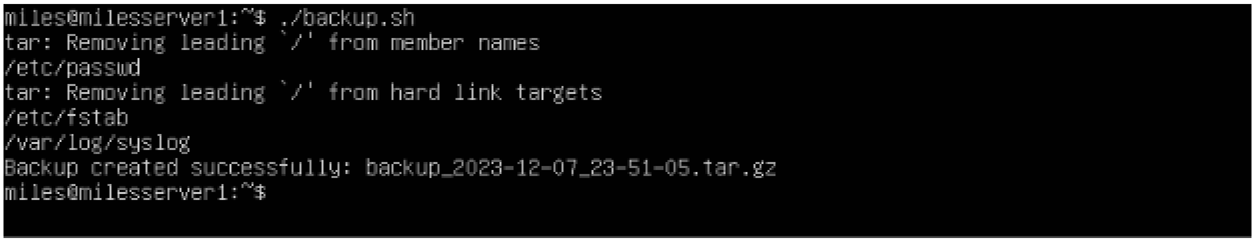
```
0 */2 * * * /home/miles/user_snapshot.log
```

When the script is ran it creates this file



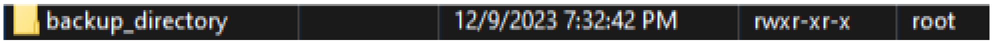
```
user_snapshot.log    2 KB    12/7/2023 6:10:09 PM    rw-r--r--    root
```

backup.sh: to run this script type sudo ./backup.sh



```
miles@milesserver1:~$ ./backup.sh
tar: Removing leading `/' from member names
/etc/passwd
tar: Removing leading `/' from hard link targets
/etc/fstab
/var/log/syslog
Backup created successfully: backup_2023-12-07_23-51-05.tar.gz
miles@milesserver1:~$
```

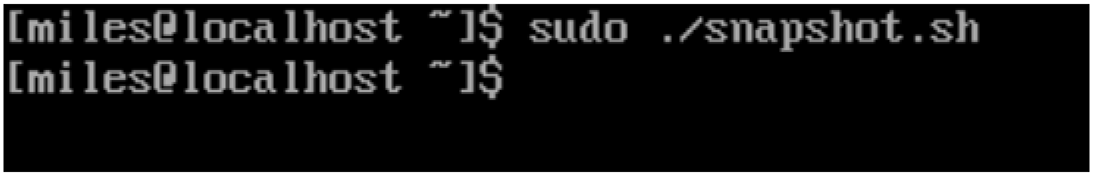
Creates this directory of the backuped files



```
backup_directory    12/9/2023 7:32:42 PM    rwxr-xr-x    root
```

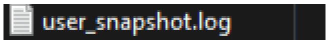
Scripts run in centOS:

snapshot.sh: to run this script type sudo ./snapshot.sh



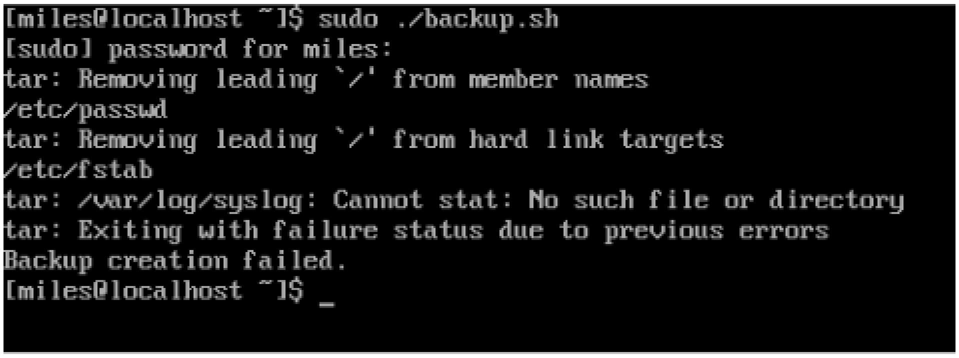
```
[miles@localhost ~]$ sudo ./snapshot.sh
[miles@localhost ~]$
```

File made



user_snapshot.log

backup.sh: to run this script type sudo ./backup.sh



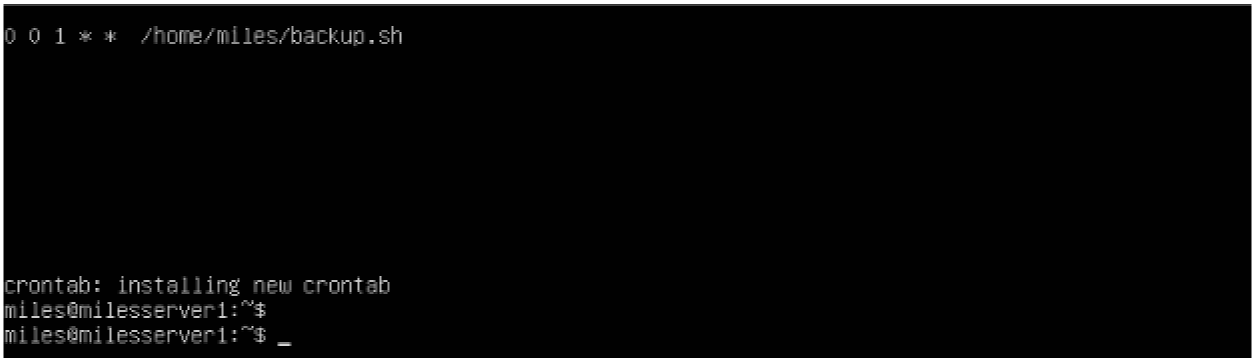
```
[miles@localhost ~]$ sudo ./backup.sh
[sudo] password for miles:
tar: Removing leading `/' from member names
/etc/passwd
tar: Removing leading `/' from hard link targets
/etc/fstab
tar: /var/log/syslog: Cannot stat: No such file or directory
tar: Exiting with failure status due to previous errors
Backup creation failed.
[miles@localhost ~]$ _
```

3. Set up a cron job to run your backup script at specific intervals (daily, weekly, and/or monthly). Document both how you set up the cron job, and make notes on why you've chosen the frequency you have. Frequency of backup MUST include a source on why that timing is a good practice.

To do this I used crontab -e and put in this in the file, this cronjob will have the script run at midnight on the first day of every month. The reason I chose run the backup script every month is because I feel like it would be useful to have these files that are important backed up, but I think weekly and daily is too much.

Source:on page 41 "Periodically test backups (at least monthly for critical data) to verify their integrity and their ability to be restored."

https://web.archive.org/web/20220901001431/https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-209.pdf



```
0 0 1 * *  /home/miles/backup.sh







crontab: installing new crontab
miles@milesserver1:~$
miles@milesserver1:~$ _
```