Unit 1.1: Graded Assignment: Rsync

Group Members:

Syed Muhammad Raqim Ali Shah (2303.KHI.DEG.008)

Umaima Siddiqui (2023.KHI.DEG.033)

Assignment:

On a linux server setup a cron job for copying example data with rsync periodically.
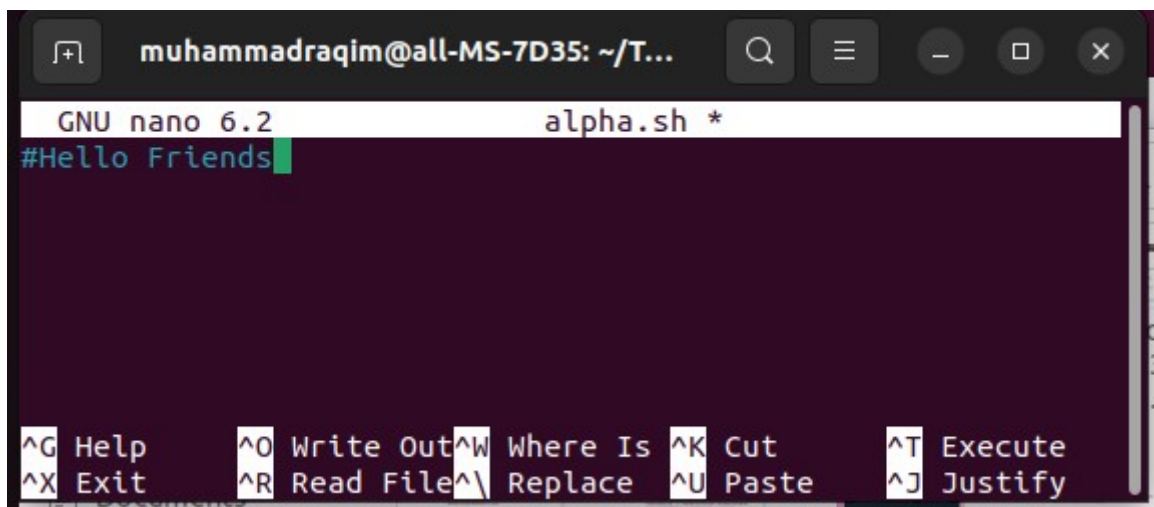
Ensure the copying is handled in the background and independently of the user session.

Solution:

We open terminal and created a file named "alpha.sh" using nano command

And wrote a demo text on "alpha.sh"

To create or edit a crontab file we passed the command:

```
muhammadraqim@all-MS-7D35:~$ crontab -e
```

In the file, We added the rsync command for copying the data. As we want to copy a file named `/home/muhammadraqim/alpha.sh` to another location that is /home/muhammadraqim/Templates/. We also specified the schedule for running the command, as mentioned manner:

```
#      ──────────── minute (0 - 59)
# │  ┌────────── hour (0 - 23)
# │  │  ┌──────── day of the month (1 - 31)
# │  │  │  ┌────── month (1 - 12)
# │  │  │  │  ┌──── day of the week (0 - 6) (Sunday to Saturday;
# │  │  │  │  │                        7 is also Sunday on some systems)
# │  │  │  │  │
# │  │  │  │  │
# * * * * * command to execute
```

Figure 1: Reference: https://www.hostinger.com/tutorials/cron-job? ppc_campaign=google_search_generic_hosting_all&bidkw=defaultkeyword&lo=1011081&gclid= Cj0KCQjw8qmhBhClARIsANAtboeiMrnahlOFpXY- ObomdRESRuljTAoJGvv4ev0hHZFuVigaQdglBM0aAlNdEALw_wcB
As we want it to perform backup in every 1 minute

GNU nano 6.2   /tmp/crontab.vBKdT4/crontab
```
# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').
#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h  dom mon dow   command
#1 * * * * rsync -a/path/to/Home/my_backup_script.sh /path/to/Home/Templates
1 * * * * rsync -a /home/muhammadraqim/alpha.sh /home/muhammadraqim/Templates/
```

[Wrote 27 lines]

^G Help   ^O Write Out   ^W Where Is   ^K Cut   ^T Execute   ^C Location   M-U Undo   M-A Set Mark
^X Exit   ^R Read File   ^\ Replace   ^U Paste   ^J Justify   ^/ Go To Line   M-E Redo   M-6 Copy

It will automatically copy "alpha.sh" from Home folder path to the Templates folder in every 1 minute. And beauty of crontab is, it will copy only the modification performed on the "alpha.sh" in Home folder.