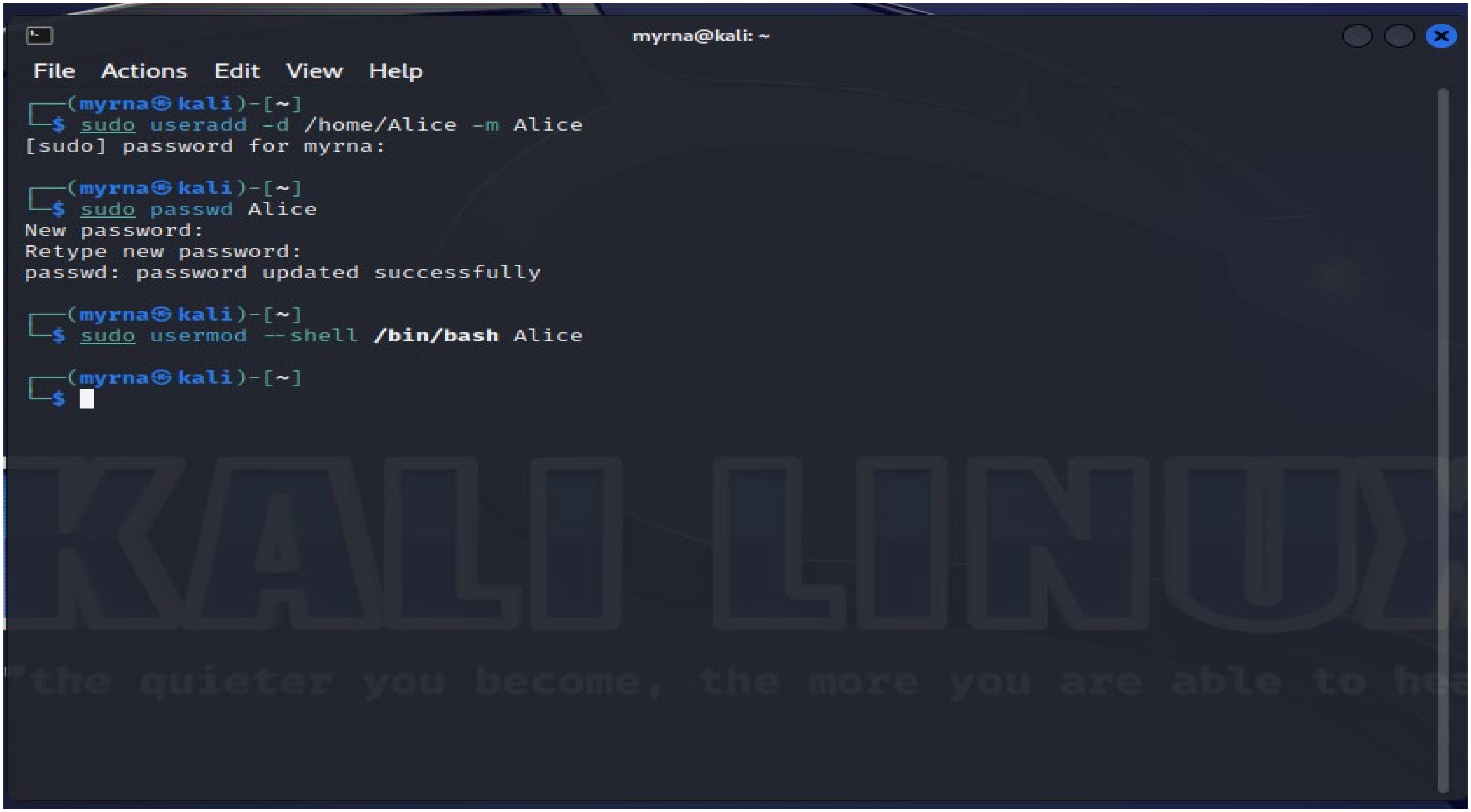
Myrna E Santiago

CYSE 270_20306

November 7, 2023

Task A, Step 1: Alice account was created with home directory and password assigned to Thunder32!.

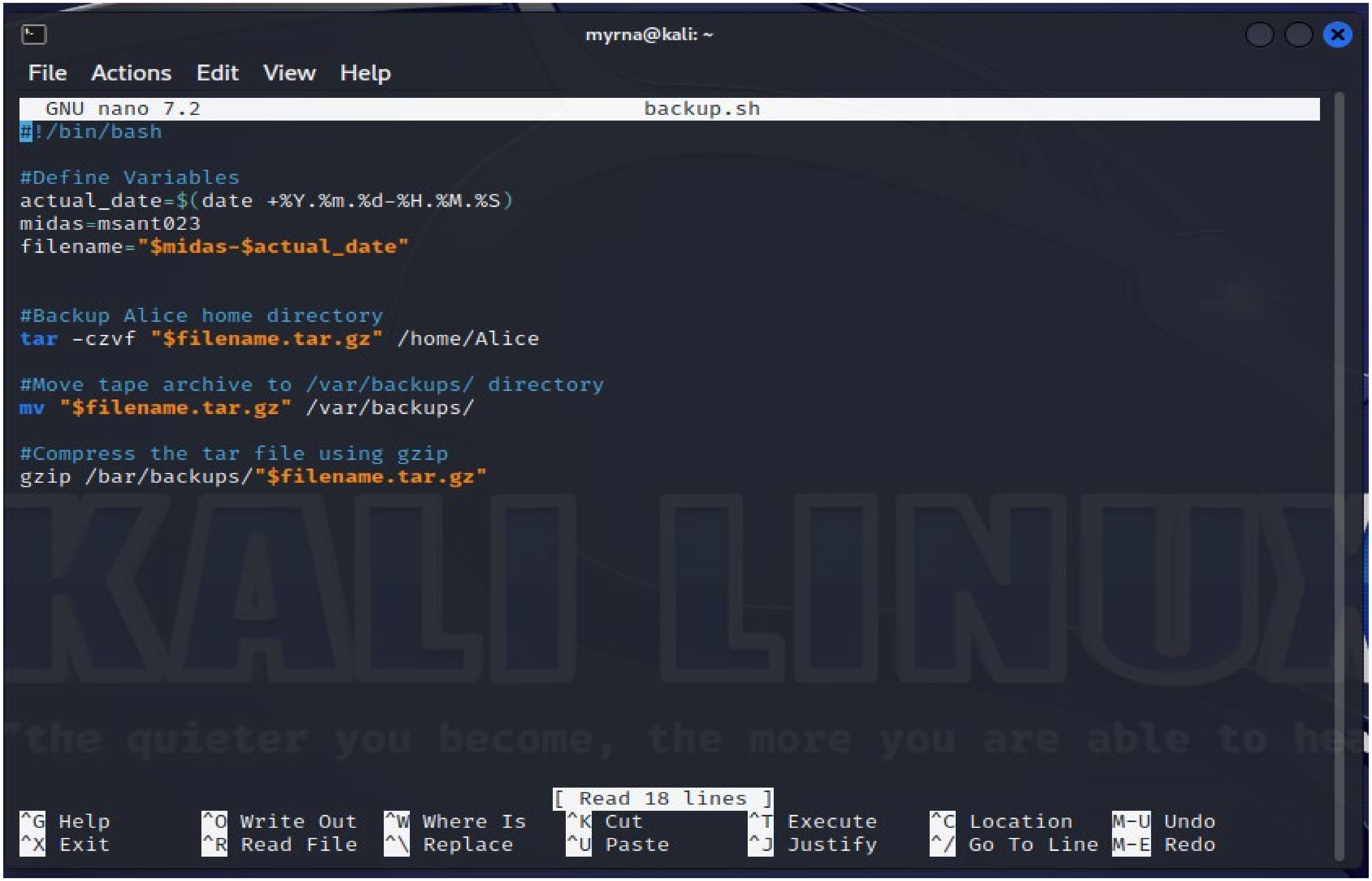


myrna@kali: ~
File Actions Edit View Help

```
┌──(myrna㉿kali)-[~]
└─$ sudo useradd -d /home/Alice -m Alice
[sudo] password for myrna:

┌──(myrna㉿kali)-[~]
└─$ sudo passwd Alice
New password:
Retype new password:
passwd: password updated successfully

┌──(myrna㉿kali)-[~]
└─$ sudo usermod --shell /bin/bash Alice

┌──(myrna㉿kali)-[~]
└─$ 
```

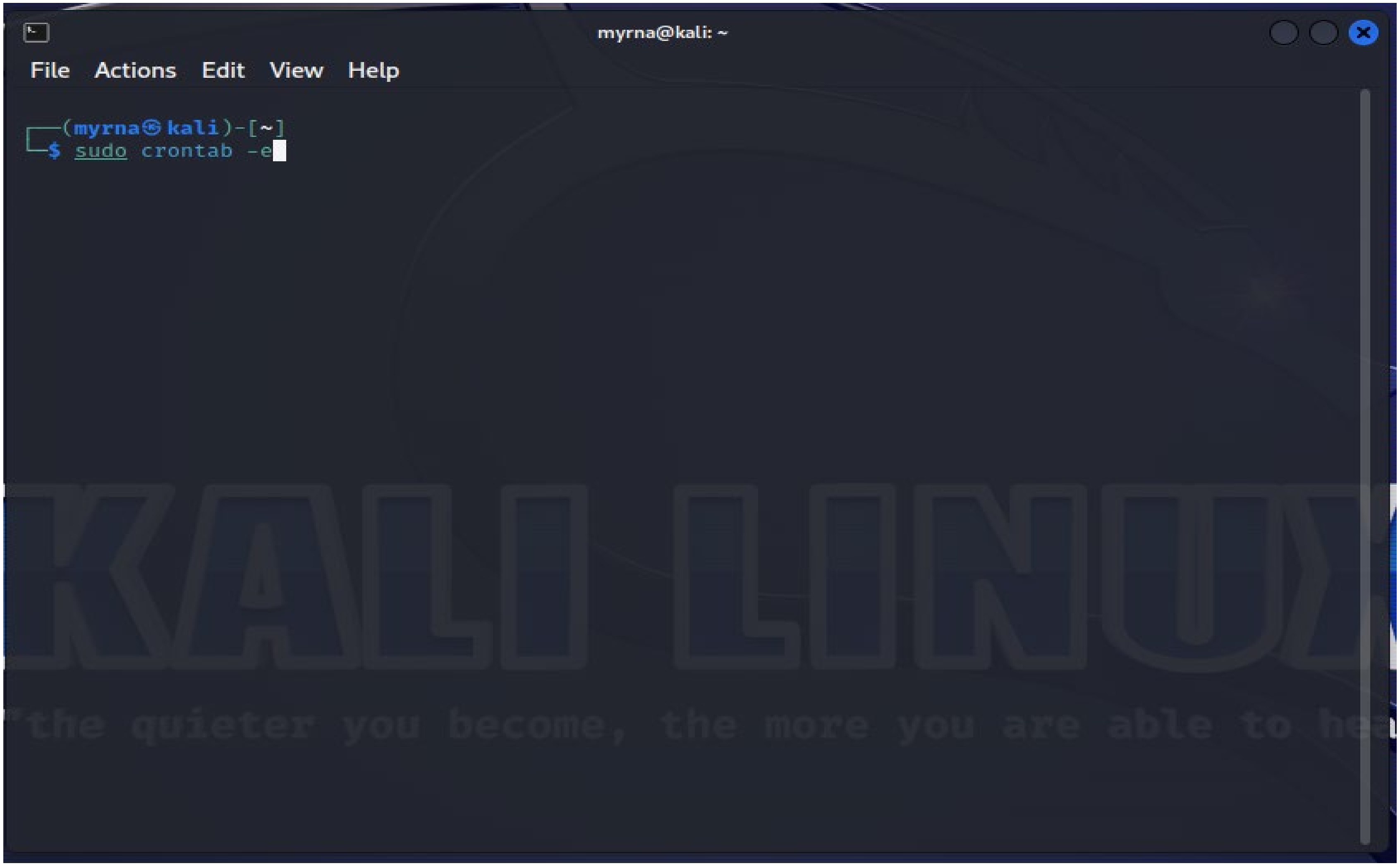
KALI LINUX
"the quieter you become, the more you are able to hea

Step 2A-B-C: A shell script was written to take two inputs to assign them to filename variable in the format of Midas and date-time. A tape archive was also created. After which the archive file was moved to the /var/backups and compress to maximize storage.



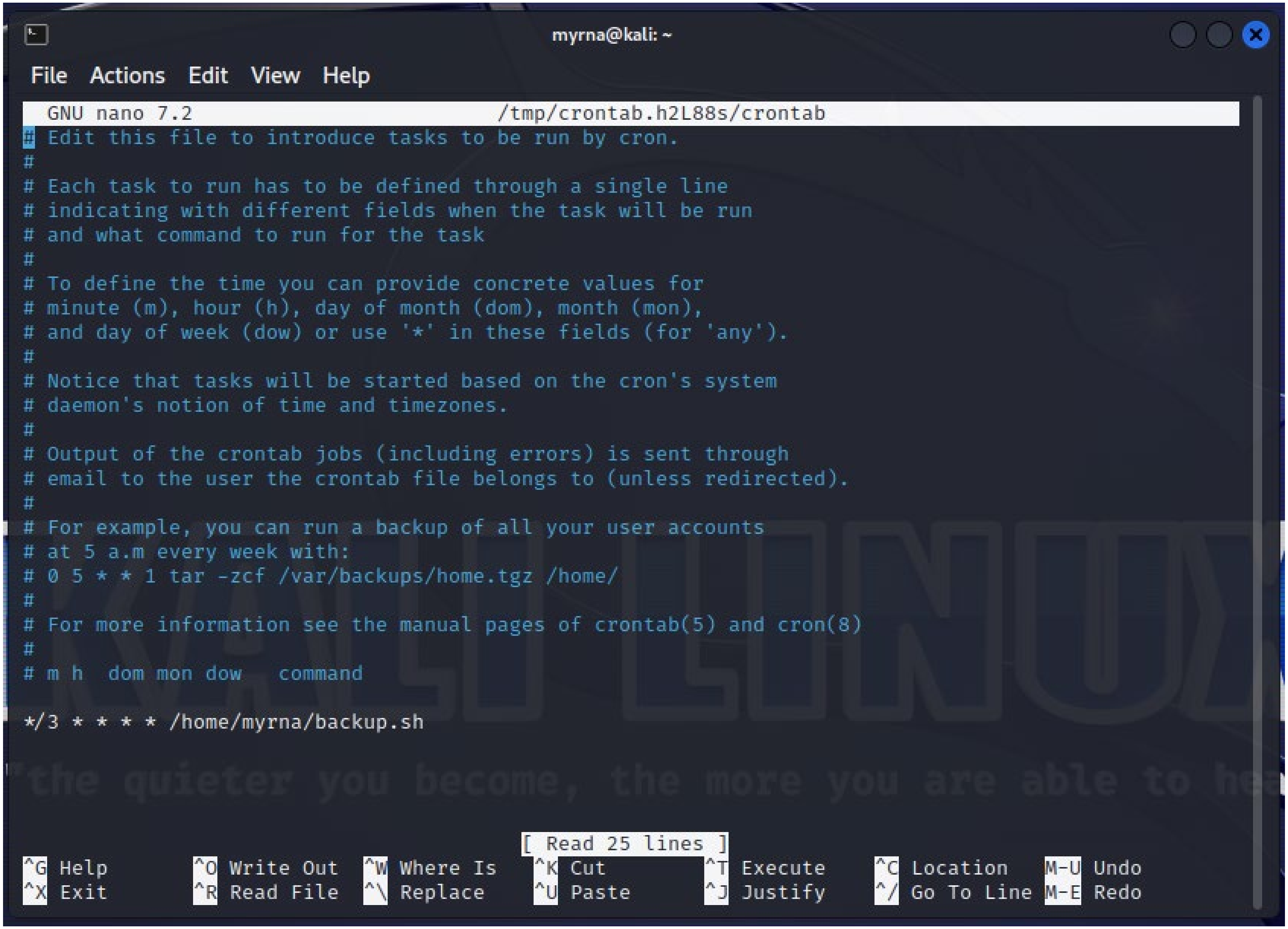
```
#!/bin/bash

#Define Variables
actual_date=$(date +%Y.%m.%d-%H.%M.%S)
midas=msant023
filename="$midas-$actual_date"


#Backup Alice home directory
tar -czvf "$filename.tar.gz" /home/Alice

#Move tape archive to /var/backups/ directory
mv "$filename.tar.gz" /var/backups/

#Compress the tar file using gzip
gzip /bar/backups/"$filename.tar.gz"
```

Step 3: A crontab file was created using crontab -e to schedule the task to run in 3 minutes intervals and save the result on /var/backups.



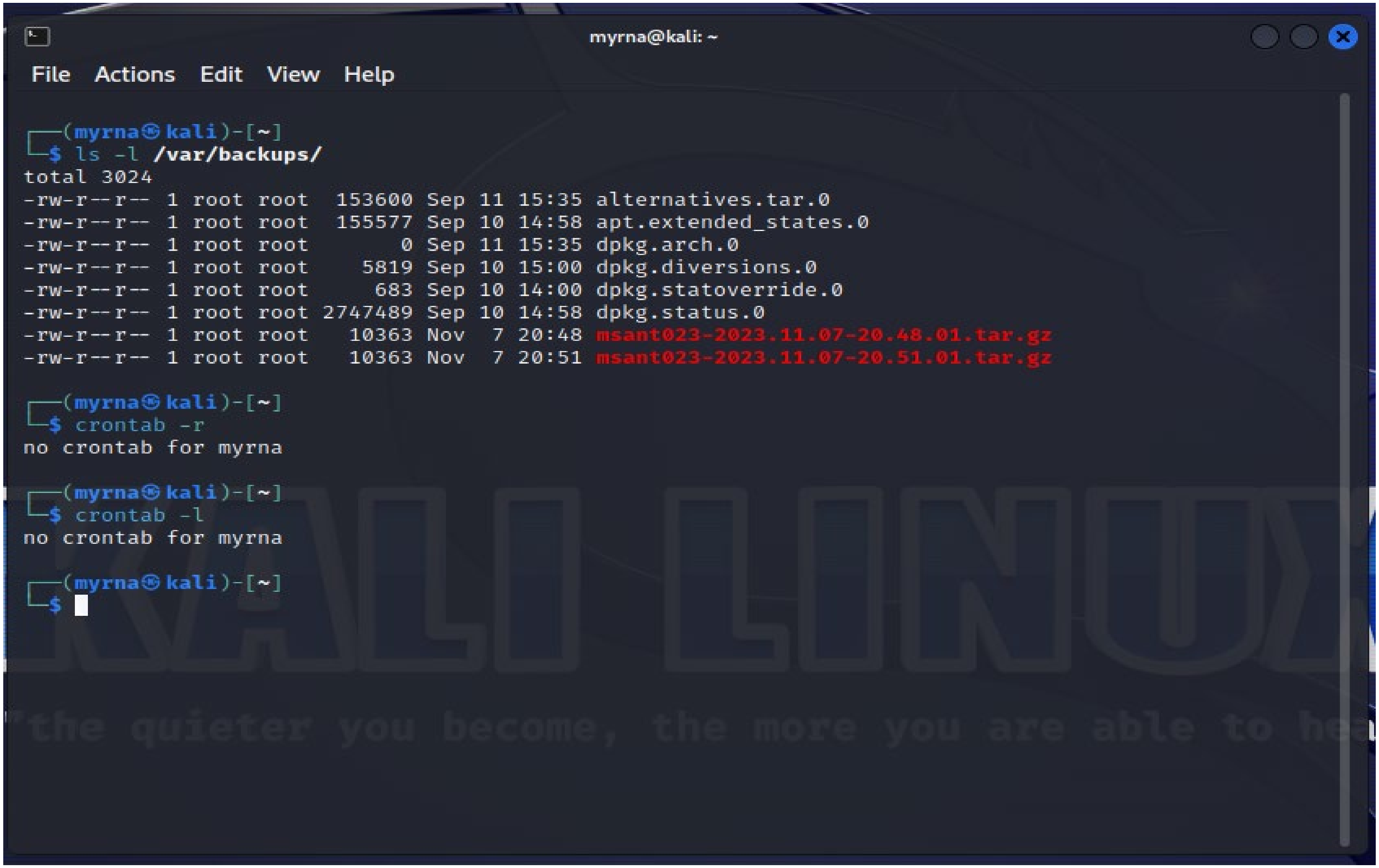
myrna@kali: ~
File Actions Edit View Help
KALI LINUX
"the quieter you become, the more you are able to hea

```
┌──(myrna㉿kali)-[~]
└─$ sudo crontab -e
```



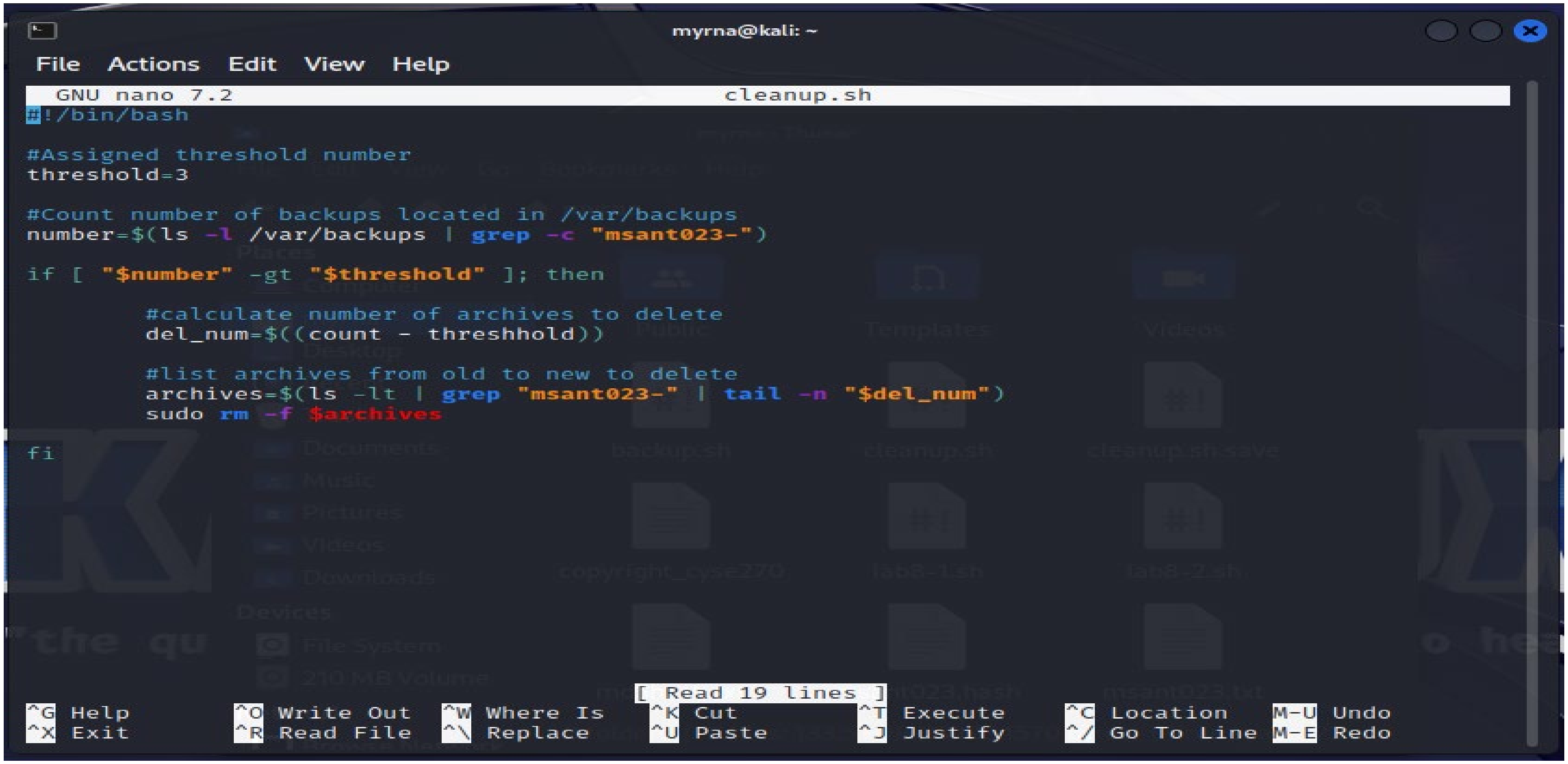
```
  GNU nano 7.2                         /tmp/crontab.h2L88s/crontab
# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define 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

*/3 * * * * /home/myrna/backup.sh



                                [ Read 25 lines ]
^G Help        ^O Write Out   ^W Where Is    ^K Cut         ^T Execute     ^C Location    M-U Undo
^X Exit        ^R Read File   ^\ Replace     ^U Paste       ^J Justify     ^/ Go To Line  M-E Redo
```

Step 3-4: The results of the backups can be seen in red. Once we obtain them we use cronbat -r to eliminate the cronbat file and cronbat -l to verify that the file was erase.



```
┌──(myrna㉿kali)-[~]
└─$ ls -l /var/backups/
total 3024
-rw-r--r-- 1 root root  153600 Sep 11 15:35 alternatives.tar.0
-rw-r--r-- 1 root root  155577 Sep 10 14:58 apt.extended_states.0
-rw-r--r-- 1 root root       0 Sep 11 15:35 dpkg.arch.0
-rw-r--r-- 1 root root    5819 Sep 10 15:00 dpkg.diversions.0
-rw-r--r-- 1 root root     683 Sep 10 14:00 dpkg.statoverride.0
-rw-r--r-- 1 root root 2747489 Sep 10 14:58 dpkg.status.0
-rw-r--r-- 1 root root   10363 Nov  7 20:48 msant023-2023.11.07-20.48.01.tar.gz
-rw-r--r-- 1 root root   10363 Nov  7 20:51 msant023-2023.11.07-20.51.01.tar.gz

┌──(myrna㉿kali)-[~]
└─$ crontab -r
no crontab for myrna

┌──(myrna㉿kali)-[~]
└─$ crontab -l
no crontab for myrna

┌──(myrna㉿kali)-[~]
└─$ 
```

Task B: System Cleanup was created by shell script. Using the format of “Midas-“ we scan /var/backups directory to find all the instances that the backup was done. Once identified from oldest to most recent the scrip will erase the oldest ones and leaves the most recent 3.



```
  GNU nano 7.2                                   cleanup.sh
#!/bin/bash

#Assigned threshold number
threshold=3

#Count number of backups located in /var/backups
number=$(ls -l /var/backups | grep -c "msant023-")

if [ "$number" -gt "$threshold" ]; then

        #calculate number of archives to delete
        del_num=$((count - threshhold))

        #list archives from old to new to delete
        archives=$(ls -lt | grep "msant023-" | tail -n "$del_num")
        sudo rm -f $archives

fi

                                  [ Read 19 lines ]
```