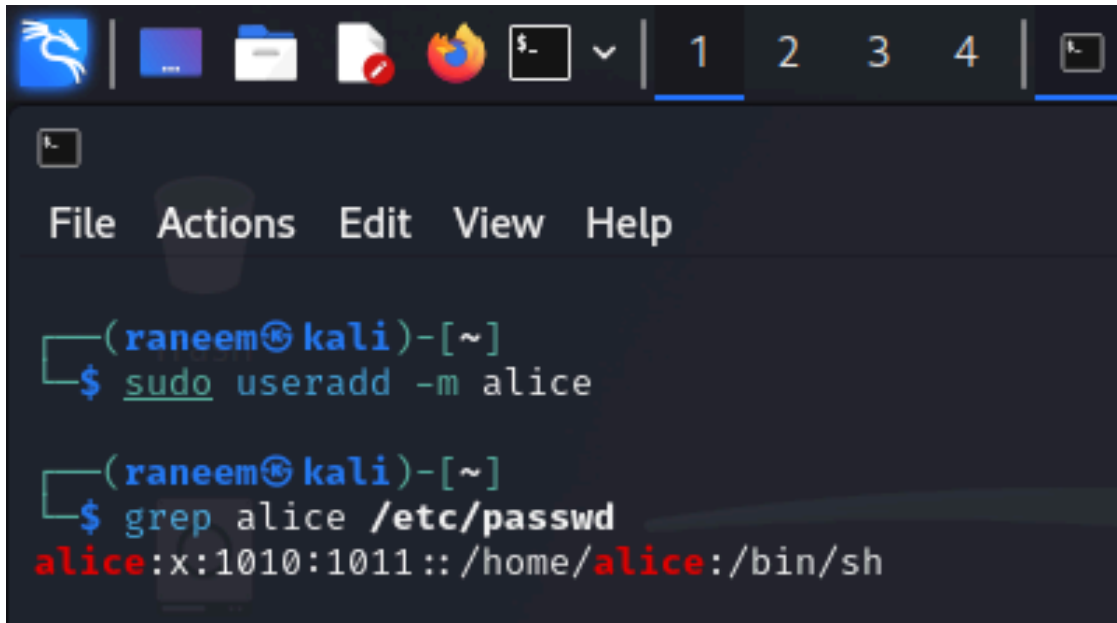


TASK A – Backup Your System (Using Crontab)

1. Create a new user Alice (with home directory)



```
(raneem@kali)-[~]
└─$ sudo useradd -m alice

(raneem@kali)-[~]
└─$ grep alice /etc/passwd
alice:x:1010:1011::/home/alice:/bin/sh
```

2. Write a shell script that backups Alice's home directory by creating a tar file (tape archive), using the following steps:
 - A.
 - Take 2 inputs with their values- your MIDAS name and current date (midas = ralar002)
 - Create a variable named as filename that should be assigned the value as MIDAS-date
 - Using tar command, create a tape archive for Alice's home directory (/home/Alice) and the filename created above (in step-2-ii)
 - B. Move the tape archive file/tar file (created in step 2-iii) to /var/backups/ directory using correct command in linux
 - C. To optimize the disk usage, pick a compression algorithm (bz2, gzip, or xv) to compress the tar file you created in /var/backups/ in the previous step-2b

```
File Actions Edit View Help

(raneem@kali)-[~]
└─$ vi backup_alice.sh
```

```
File Actions Edit View Help
raneem@kali: ~

#!/bin/bash

echo "Enter your MIDAS name: "
read midas

#get current date/time
current_date=$(date '+%Y.%m.%d-%H.%M.%S')
filename="${midas}-${current_date}"
echo $filename

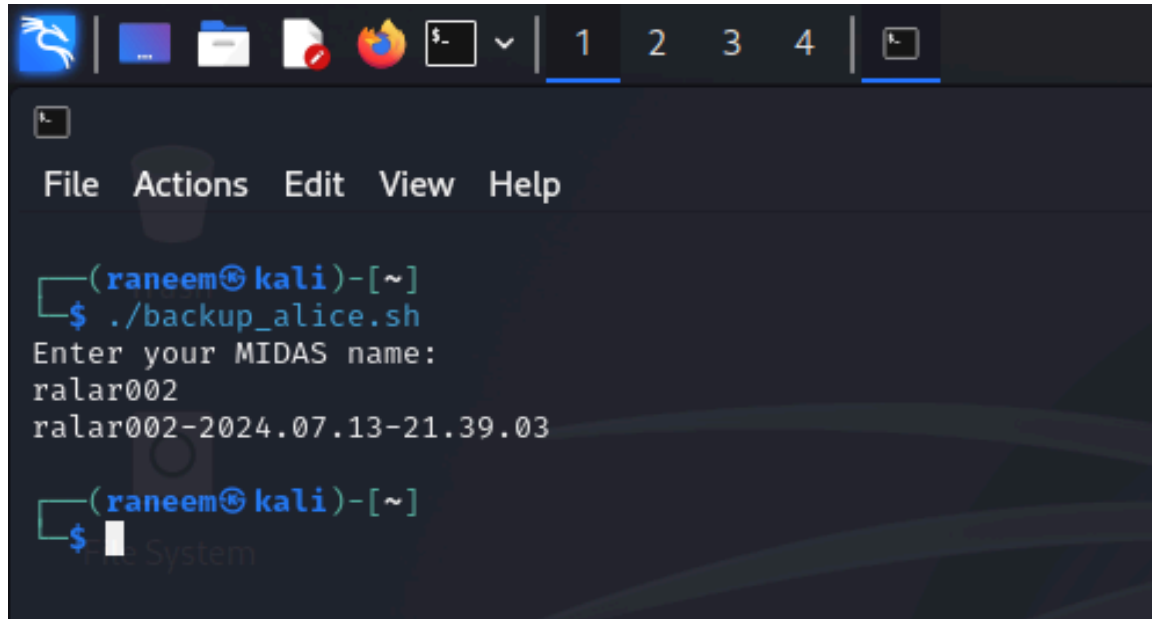
#define path to tar file
tar_file="/home/alice/${filename}.tar"

#create tar file
sudo tar -cvf "$tar_file" -C /home/alice --exclude="${filename}.tar"

#move tar file to /var/backups/
sudo mv "$tar_file" /var/backups

#compress file
sudo gzip /var/backups/${filename}.tar

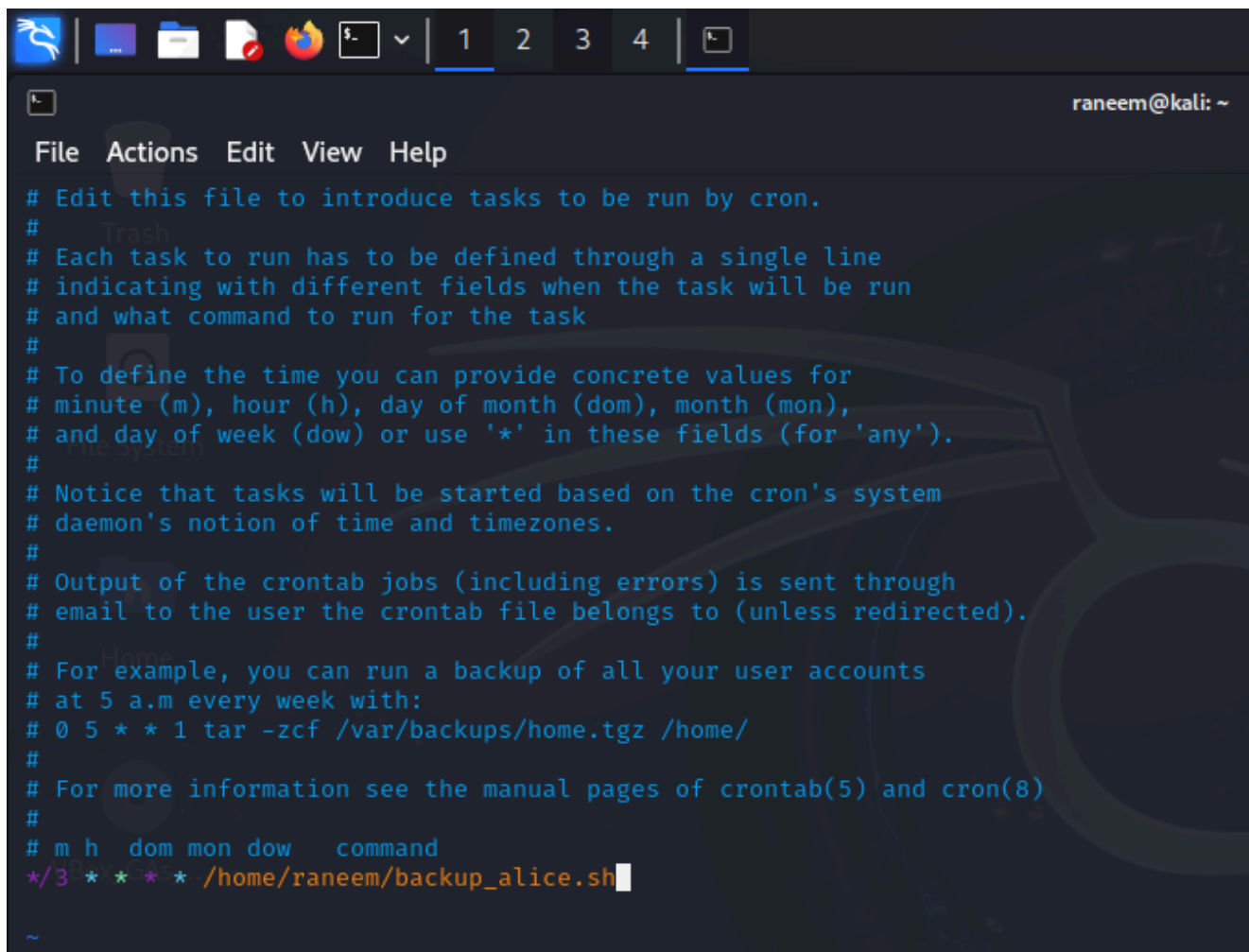
~
```



A terminal window on a Kali Linux system. The window title is "(raneem@kali)-[~]". The user has executed the command `./backup_alice.sh`. The script prompts for a MIDAS name, and the user enters `ralar002`. The script then outputs `ralar002-2024.07.13-21.39.03`. The terminal prompt returns to `(raneem@kali)-[~]`.

```
(raneem@kali)-[~]
└─$ ./backup_alice.sh
Enter your MIDAS name:
ralar002
ralar002-2024.07.13-21.39.03
(raneem@kali)-[~]
└─$
```

3. **Create a crontab file** to keep the scheduled task running for 3 minutes, then check the contents in the `/var/backups` directory



A terminal window on a Kali Linux system showing the content of the crontab file. The window title is "raneem@kali: ~". The terminal displays the following text:

```
# 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/raneem/backup_alice.sh
```

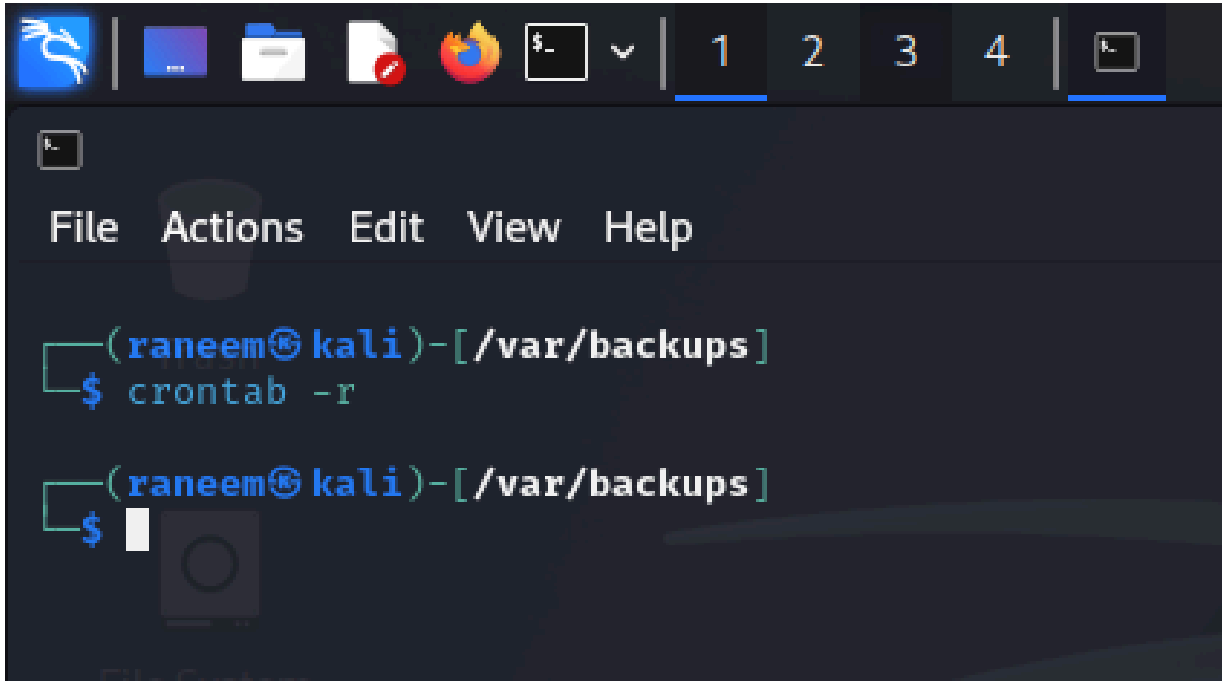
```
(raneem@kali)-[~]
└─$ crontab -e
crontab: installing new crontab

(raneem@kali)-[~]
└─$
```

```
(raneem@kali)-[/var/backups]
└─$ ls /var/backups
Alice                dpkg.arch.0          ralar002-2024.07.13-15.51.26.tar.gz  raneem-2024.07.13-21.54.31.tar.gz
alternatives.tar.0  dpkg.diversions.0   ralar002-2024.07.13-15.53.41.tar.gz  tmp
apt.extended_states.0  dpkg.statoverride.0 ralar002-2024.07.13-21.55.16.tar.gz
apt.extended_states.1.gz  dpkg.status.0       ralar002-2024.07.13-21.57.27.tar.gz

(raneem@kali)-[/var/backups]
└─$
```

4. **Cancel the crontab** jobs



```
(raneem@kali)-[/var/backups]
└─$ crontab -r

(raneem@kali)-[/var/backups]
└─$
```