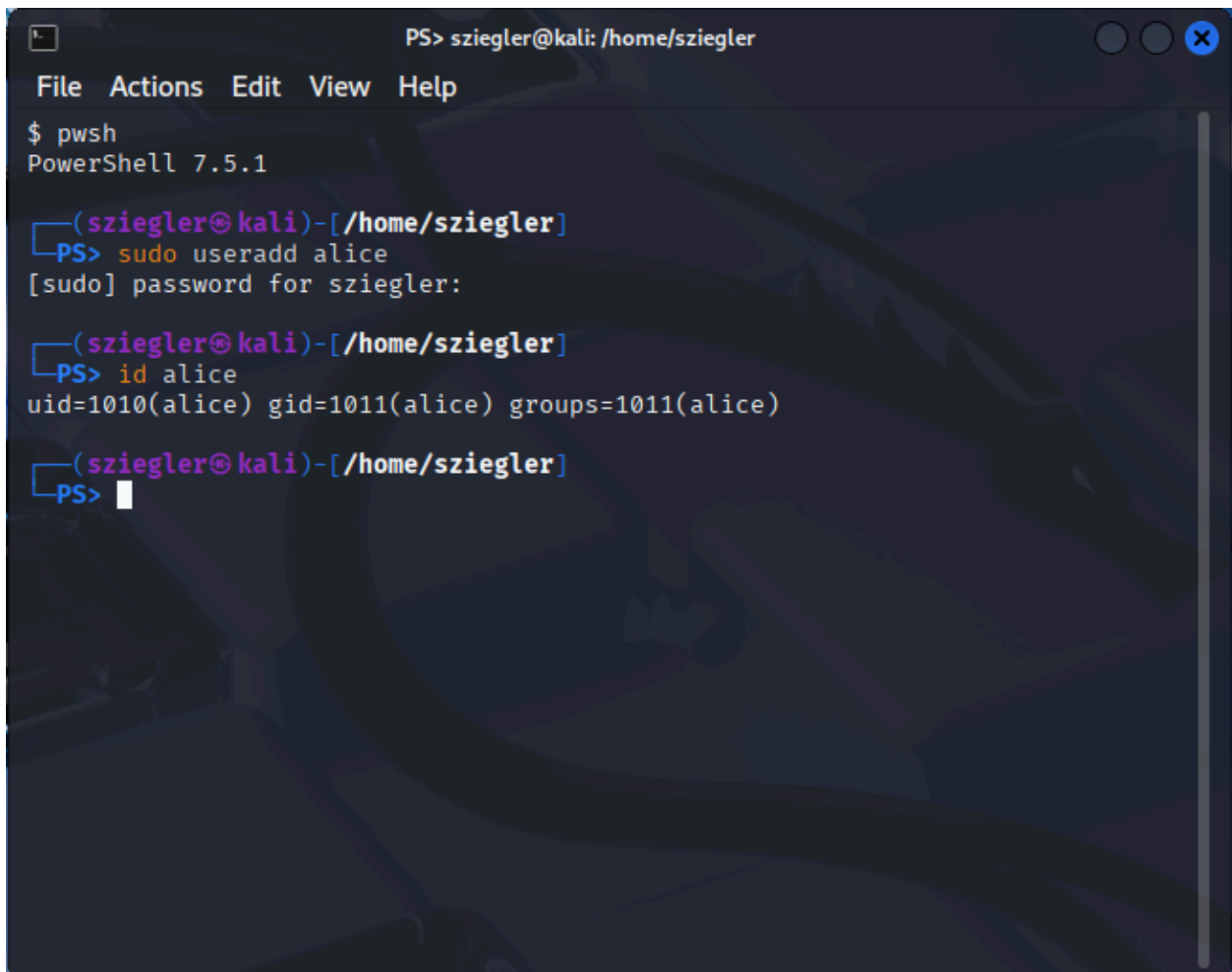


Shawn Ziegler

Task A

Step 1.



```
PS> sziegler@kali: /home/sziegler
File Actions Edit View Help
$ pwsh
PowerShell 7.5.1

(sziegler@kali)-[/home/sziegler]
└─┬─ PS> sudo useradd alice
   │ [sudo] password for sziegler:
   └─┬─ (sziegler@kali)-[/home/sziegler]
      │ └─┬─ PS> id alice
          │ uid=1010(alice) gid=1011(alice) groups=1011(alice)
          └─┬─ (sziegler@kali)-[/home/sziegler]
             │ └─┬─ PS> █
             └─┬─
```

Step 2.

Part A

```
PS> sziegler@kali: /home/sziegler
File Actions Edit View Help

(sziegler@kali)-[/home/sziegler]
└─┬─ PS> vi alice_input.sh

(sziegler@kali)-[/home/sziegler]
└─┬─ PS> sudo chmod +x alice_input.sh
   │ [sudo] password for sziegler:
   └─

(sziegler@kali)-[/home/sziegler]
└─┬─ PS> ls -l alice_input.sh
   │ -rwxrwxr-x 1 sziegler sziegler 1312 Nov  8 13:59 alice_input.sh
   └─

(sziegler@kali)-[/home/sziegler]
└─┬─ PS> █
```

```
sziegler@kali: ~
File Actions Edit View Help
#!/bin/bash

# alice_input.sh

MIDAS_Name=szieg004
echo "Current midas name is $MIDAS_NAME."

CURRENT_DATE=$(date +%Y.%m.%d-%i.%M.%S) #| tr -d '\n'
echo "Current date and time is $CURRENT_DATE."

filename="${MIDAS_Name}-${CURRENT_DATE}.tar"

COMPRESSED_FILENAME="${filename}.gz"
echo ""

echo "Preparing to backup /home/alice ..."
echo "Archive will be named: ${filename}."
echo "The compressed file version is ${COMPRESSED_FILENAME}."
echo ""

echo "Creating tar archive of /home/alice ..."
tar -cvf "${filename}" /home/alice

if [ $? -ne 0 ]; then
    echo "Error: Failed to create tar archive/file. Exiting script."
-- INSERT --
9,32 Top
```

```
PS> sziegler@kali: /home/sziegler
File Actions Edit View Help
echo "Current date and time is $CURRENT_DATE,"
filename="${MIDAS_NAME}-${CURRENT_DATE}.tar"
COMPRESSED_FILENAME="${filename}.gz"
echo ""
echo "Preparing to backup /home/alice ..."
echo "Archive will be named: ${filename},"
echo "The compressed file version is ${COMPRESSED_FILENAME},"
echo ""
echo "Creating tar archive of /home/alice ..."
tar -cvf "${filename}" /home/alice
if [ $? -ne 0 ]; then
    echo "Error: Failed to create a tar archive/file. Existing script."
    exit 1
fi
echo "Tar archive '${filename}' created successfully."
echo ""
-- INSERT -- 35,8 Bot
```

Part B

```
PS> sziegler@kali: /home/sziegler
File Actions Edit View Help
    echo "Error: Failed to create a tar archive/file. Exiting script."
    exit 1
fi

echo "Tar archive '${filename}' created successfully."
echo ""

BACKUP_DIR="/var/backups"

echo "Moving '${filename}' to '${BACKUP_DIR}' ... "
mv "${filename}" "${BACKUP_DIR}/${filename}"

if [ $? -ne 0 ]; then
    echo "Error: Failed to move archive to ${BACKUP_DIR}, Exiting script."
    exit 1
fi

echo "Archive moved to '{BACKUP_DIR}',"
echo ""

-- INSERT --                               52,8                               Bot
```

Part C

```
PS> sziegler@kali: /home/sziegler
File Actions Edit View Help
echo "Moving '${filename}' to '${BACKUP_DIR}' ... "
mv "${filename}" "${BACKUP_DIR}/${filename}"

if [ $? -ne 0 ]; then
    echo "Error: Failed to move archive to ${BACKUP_DIR}, Exiting script."
    exit 1
fi

echo "Archive moved to '{BACKUP_DIR}',"
echo ""

echo "Compressing '${BACKUP_DIR}/${filename}' using gzip ... "
gzip "${BACKUP_DIR}/${filename}"

if [ $? -ne 0 ]; then
    echo "Error: Failed to compress archive, Exiting script."
    exit 1
fi

echo "Backup completed successfully!"
echo "The compressed backup file is located at: ${BACKUP_DIR}/${COMPRESSED_FILENAME}."
echo ""

-- INSERT --                                     66,8                                     Bot
```

Step 3

```
PS> sziegler@kali: /home/sziegler
File Actions Edit View Help
GNU nano 3.4 /tmp/crontab.grtkif/crontab
(sziegler@kali)-[/home/sziegler]
└─# PS> sudo crontab -e
no crontab for root - using an empty one
crontab: installing new crontab

(sziegler@kali)-[/home/sziegler]
└─# PS> sudo crontab -e
No modification made
```

```
PS> sziegler@kali: /home/sziegler
File Actions Edit View Help
No modification made /tmp/crontab.grTk1f/crontab *
(sziegler@kali)-[/home/sziegler]
PS> sudo crontab -l
# 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/sziegler/alice_input.sh
```

```
PS> sziegler@kali: /home/sziegler
File Actions Edit View Help
GNU nano 3.4 /tmp/crontab.grTk1f/crontab
(sziegler@kali)-[/home/sziegler]
└─# ps> ls -lh /var/backups/
total 3.1M
-rw-r--r-- 1 root root 170K Sep  4 18:06 alternatives.tar.0
-rw-r--r-- 1 root root 162K Aug 25 18:09 apt.extended_states.0
-rw-r--r-- 1 root root   0 Sep  4 18:06 dpkg.arch.0
-rw-r--r-- 1 root root 8.3K Aug 25 18:09 dpkg.diversions.0
-rw-r--r-- 1 root root  683 Aug 25 18:07 dpkg.statoverride.0
-rw-r--r-- 1 root root 2.7M Aug 25 18:08 dpkg.status.0

(sziegler@kali)-[/home/sziegler]
└─# ps>
# For more information see the manual pages of crontab(1) and cron(8)
# See also the cron(8) command
*/3 * * * * /home/sziegler/alice_input.sh
write to File: /tmp/crontab.grTk1f/crontab
⌘ Help      ⌘-D DOS Format  ⌘-A Append    ⌘-B Backup File
⌘ Cancel    ⌘-M Mac Format  ⌘-P Prepend   ⌘ Browse
```

Step 4

```
PS> sziegler@kali: /home/sziegler
File Actions Edit View Help
$ pwsh
PowerShell 7.5.1

(sziegler@kali)-[/home/sziegler]
└─$ sudo crontab -r
[sudo] password for sziegler:

(sziegler@kali)-[/home/sziegler]
└─$ sudo crontab -l
no crontab for root

(sziegler@kali)-[/home/sziegler]
└─$ crontab -l
no crontab for sziegler

(sziegler@kali)-[/home/sziegler]
└─$ PS> █
```

Task B

```
PS> sziegler@kali: /home/sziegler
File Actions Edit View Help
GNU nano 8.4 cleanup_backups.sh *
#!/bin/bash

threshold=3

backup_count=$(ls /var/backups/*.tar.gz 2>/dev/null | wc -l)

if [ "$backup_count" -gt "$threshold" ]; then
    echo "Number of backups exceeds threshold ($threshold)." Cleaning up b>

    to_delete=$((backup_count - threshold))

    ls -t /var/backups/*.tar.gz | tail -n $to_delete | xargs sudo rm -f
    echo "Deleted $to_delete old backup(s)."
else
    echo "Number of backups is under the threshold ($threshold)." No cleaup n>
fi
```

^G Help **^O** Write Out **^F** Where Is **^K** Cut **^T** Execute **^C** Location
^X Exit **^R** Read File **^N** Replace **^U** Paste **^J** Justify **^/** Go To Line

```
PS> sziegler@kali: /home/sziegler
File Actions Edit View Help

(sziegler@kali)-[/home/sziegler]
└─┬─ PS> sudo nano cleanup_backups.sh

(sziegler@kali)-[/home/sziegler]
└─┬─ PS> chmod +x cleanup_backups.sh
    │ chmod: changing permissions of 'cleanup_backups.sh': Operation not permitted
    └─

(sziegler@kali)-[/home/sziegler]
└─┬─ PS> sudo ./cleanup_backups.sh
    │ sudo: ./cleanup_backups.sh: command not found
    └─

(sziegler@kali)-[/home/sziegler]
└─┬─ PS> sudo chown sziegler:sziegler alice_input.sh

(sziegler@kali)-[/home/sziegler]
└─┬─ PS> sudo chown sziegler:sziegler cleanup_backups.sh

(sziegler@kali)-[/home/sziegler]
└─┬─ PS> █
```