

ssh node1.domain6.example.com
roxicant

=====

1)

Create users

On node1 create user accounts with the following characteristics:

- ☐ A user named chris with a user id of 2345
- ☐ A user named eric who also belongs to a secondary group named devops
- ☐ A user named joe who also belongs to a secondary group named devops
- ☐ The password for all three user accounts is roxicant

Back

Provide feedback on this item

useradd -u 2345 chris

useradd eric

useradd joe

groupadd devops

usermod -G devops eric

usermod -G devops joe

passwd chris roxicant

passwd eric roxicant

passwd joe roxicant

=====

2)

Create files and directories

On node1 create the following directories according to the following specifications:

- ☐ /penguin directory is owned by the user `chris` and belongs to the group `chris`
- ☐ /shared directory is owned by the user `root` and belongs to the group `root`
- ☐ Other users are able to read from and write to /shared
- ☐ Users who create files in /shared are able to delete the files they create but they cannot delete files created by other users

Back

Provide feedback on this item

```
mkdir /penguin
chown chris:chris /penguin
```

```
mkdir /shared
chown root:root /shared
chmod o+rw /shared
chmod o+t /shared
```

```
ls -l
```

=====

3)

Locate files

Use the `find` command to locate all files under `/usr` which are larger than 25M and smaller than 50M

- ☐ Ensure that you use the command in such a way that all files are listed with their full path name.
- ☐ Redirect the list of files that satisfy this criterion in the file `/root/output.txt`

[Back](#)

[Provide feedback on this item](#)

```
find /usr -type f -size +25M -size -50M > /root/output.txt
cat /root/output.txt
```

=====

4)

Create a user with account restrictions

On node1 create a user account according to the following specifications:

- ☐ The username for the account is `jerry`
- ☐ The password for the account is `roxicant`
- ☐ The user is prompted to change their password every 3 days

```
useradd jerry
passwd roxicant
chage -M 3 jerry
chage -d 3 jerry
chage -l jerry
```

=====

5)

Configure administrative privileges

Configure sudo for user olive such that:

- ☐ User olive is able to install, remove and update packages
- ☐ User olive is able to start, stop, restart, enable and disable a service
- ☐ User olive can run the commands required to perform the tasks above without providing a password

[Back](#)

[Provide feedback on this item](#)

Which yum

Which systemctl

vim /etc/sudoers.d/olive

```
olive ALL=(ALL) /usr/bin/yum install *
olive ALL=(ALL) /usr/bin/yum remove *
olive ALL=(ALL) /usr/bin/yum update *
olive ALL=(ALL) /usr/bin/systemctl start *
olive ALL=(ALL) /usr/bin/systemctl stop *
olive ALL=(ALL) /usr/bin/systemctl enable *
olive ALL=(ALL) /usr/bin/systemctl disable *
olive ALL=(ALL) /usr/bin/systemctl restart *
olive ALL=(ALL) NOPASSWD:ALL
```

=====

6)

Configure logging

☐ Configure logging so that all messages with facility authpriv and priority info are logged to the file /var/log/new-log

Back

Provide feedback on this item

```
vim /etc/rsyslog.d/authpriv.conf
    authpriv.info /var/log/new-log
systemctl restart rsyslog
logger -p authpriv.info "test"
tail /var/log/new-log
```

=====

7)

Install and configure a service

☐ Install and enable the vsftpd service on node1.

☐ Ensure that the vsftpd service is available across system reboots.

Back

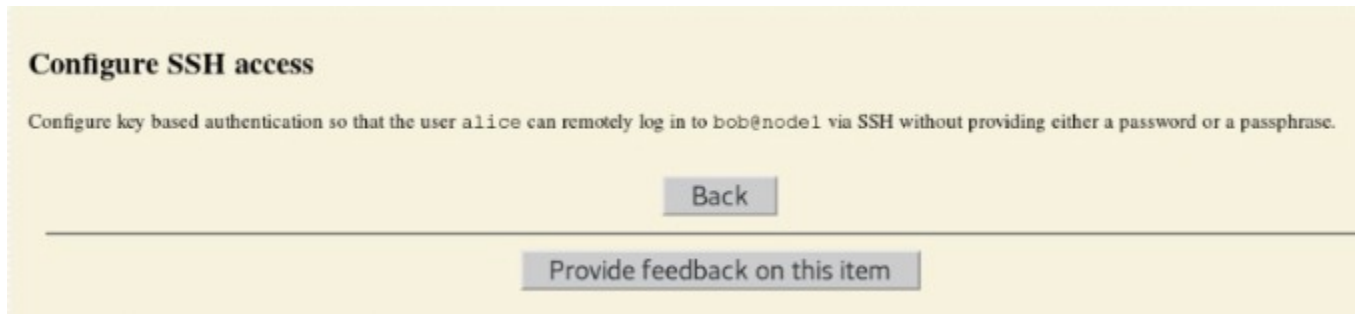
Provide feedback on this item

```
yum install vsftpd
systemctl start vsftpd
systemctl enable vsftpd
systemctl is-active vsftpd
```

systemctl is-enabled vsftpd

=====

8)



Configure SSH access

Configure key based authentication so that the user `alice` can remotely log in to `bob@node1` via SSH without providing either a password or a passphrase.

[Back](#)

[Provide feedback on this item](#)

su - alice

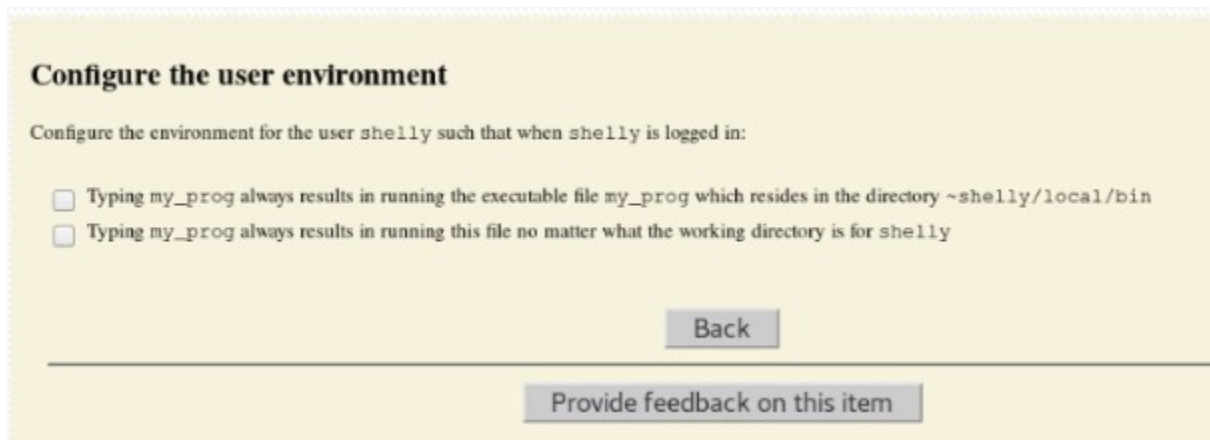
ssh-keygen

ssh-copy-id bob@node1

ssh bob@node1

=====

9)



Configure the user environment

Configure the environment for the user `shelly` such that when `shelly` is logged in:

- ☐ Typing `my_prog` always results in running the executable file `my_prog` which resides in the directory `~shelly/local/bin`
- ☐ Typing `my_prog` always results in running this file no matter what the working directory is for `shelly`

[Back](#)

[Provide feedback on this item](#)

Su - shelly

Export PATH=\${PATH}:~shelly/local/bin

my_prog

=====

10)

Configure an application

Configure the application pe124 on node1 such that when pe124 is run as user user_pe124 it displays the message
borough lively excursion nephew scarce

[Back](#)

[Provide feedback on this item](#)

su - user_pe124

vim ~/.bashrc

```
PE124="borough lively excursion nephew scarce"
```

```
echo $PE124
```

```
export PE124
```

Export pe124

=====

11)

Copy a file

The block device /dev/vda3 contains an ext4 filesystem. Copy the file secret.txt that is contained within this filesystem to the /root directory. The filesystem does not need to be permanently accessible.

[Back](#)

[Provide feedback on this item](#)

mount /dev/vda3 /mnt

ls /mnt

cp /mnt/secret.txt /root

unmount /mnt