

Connessione alla propria macchina

To access your instance:

1. Open an SSH client.
2. Locate your private key file (name-exam.pem).
3. Your key must not be publicly viewable for SSH to work. Use this command if needed:
`chmod 400 name-exam.pem`
4. Connect to your instance using its Public DNS:
`ec2-35-158-97-6.eu-central-1.compute.amazonaws.com`

Example:

```
ssh -i "name-exam.pem"  
centos@ec2-35-158-97-6.eu-central-1.compute.amazonaws.com
```

Switch to root user

```
[centos@ip-172-31-30-213 ~]$ sudo su -
```

Exercise 1: Managing Files with Shell Expansion

ITA

- Creare **108** nuovi files dal nome **seasonX_episodeY.ogg** all'interno della Directory **/exam/exercise1**. **X** è il numero della stagione e **Y** dell'episodio per **nove** stagioni e **dodici** episodi per ogni stagione.

ENG

- Create a total of **108** new files with names **seasonX_episodeY.ogg** under the path **/exam/exercise1**. Replace **X** with the season number and **Y** with that season's episode, for **nine** seasons of **twelve** episodes each.

Exercise 2: Managing stdout; stderr

ITA

- utilizzando il comando **find** e l'utente **student**, fare una ricerca sul path **/** reindirigendo l'output delle occorrenze trovate sul file **find.txt** sotto la directory **/exam/exercise2**; reindirigere gli eventuali errori sul file **/exam/exercise2/finderr.txt**. Non reindirigere lo standard error sul file **find.txt**.

ENG

- using **find** command, and **student** user, find all the occurrence (file directory etc..) under path **/** redirecting all the finding occurrence to the file **find.txt** under the path **/exam/exercise2**; redirect also all the eventually error to the file **/exam/exercise2/finderr.txt**. Do not redirect standard error into **find.txt** file.

find example

```
[centos@ip-172-31-30-213 ~]$ find ./
./
./bash_logout
./bash_profile
./bashrc
./ssh
./ssh/authorized_keys
./bash_history
./file1
./file2
```

find manual

```
[centos@ip-172-31-30-213 ~]$ man find
```

Exercise 3: User and Group

ITA

- Creare due nuovi gruppi chiamati Marvel e Disney
 - marvel con GID 2000
 - disney con GID 3000
- Creare gli utenti appartenenti al gruppo Marvel **thor**, **wolverine** e **hulk**
 - l'utente **thor** avrà le seguenti caratteristiche:
 - UID 2001
 - home directory /home/marvel/thor
 - password thor
 - dovrà cambiare password una volta al mese
 - l'utente **wolverine** avrà le seguenti caratteristiche:
 - UID 2002
 - potrà accedere e gestire i file e directory appartenenti al gruppo disney
 - password wolverine
 - dovrà cambiare password al primo accesso e successivamente una volta ogni 3 mesi
 - l'account scadrà dopo un anno dalla sua creazione
 - l'utente **hulk** avrà le seguenti caratteristiche:
 - UID 2010
 - home directory /home/marvel/hulk
 - password hulk
 - l'utente non potrà fare accesso al sistema
- Creare gli utenti **bb-8** e **yoda** appartenenti al gruppo **disney** entrambi con password centos, gruppo primario disney e gruppo secondario marvel. Entrambi gli utenti dovranno cambiare password ogni 6 mesi e l'account scadrà dopo due anni dalla sua creazione, dovranno cambiare password al primo accesso.

ENG

- Create two groups called Marvel and Disney
 - **marvel** with GID 2000
 - **disney** with GID 3000
- Create users for the group Marvel **thor**, **wolverine** and **hulk**
 - the **thor** user need to be configured with:
 - UID 2001
 - home directory /home/marvel/thor
 - password thor
 - the user need to change their password once a month
 - the **wolverine** user need to be configured with:
 - UID 2001
 - will be able to access and manage the files and directories belonging to the disney group
 - wolverine password

- the user need to change the password at the first access, and every 3 months
 - the account will expire a year after its creation
- the **hulk** user need to be configured with:
 - UID 2010
 - home directory /home/marvel/hulk
 - hulk password
 - the user won't able to access to the system
- Create users for **disney** group **bb-8** and **yoda**, both with password **centos**, primary group disney, and secondary group marvel. Both users will have to change their password every 6 months, and the account will expire two years after its creation, they will have to change their password at the first log on.

Exercise 4: File permission

ITA

- Creare una directory sotto **/home/collaboration** dove gli utenti del gruppo **collaboration**, **mark nadine** e **martin** potranno condividere files
- Solamente gli utenti del gruppo collaboration potranno creare e cancellare i file sotto **/home/collaboration**
- Tutti i file creati sotto la directory **/home/collaboration** dovranno essere assegnati automaticamente al gruppo **collaboration**

ENG

- Create a new directory **/home/collaboration** where users of the **collaboration** group, **mark nadine** and **martin** will be able to share files
- Only users of the **collaboration** group will be able to create and delete files under **/home/collaboration**
- All files created in **/home/collaboration** directory must be automatically assigned to the **collaboration** group

Exercise 5: File permission

ITA

- Creare l'utente **murdoch** appartenente al gruppo **a-team** con password uguale alla username.
- Fare in modo che nuovi files e directory creati dall'utente **murdoch** alla creazione possano essere letti e scritti dall'utente e gruppo associato ma che nessun altro utente possa accedervi.

ENG

- Create the **murdoch** user belonging to the **a-team** group with a password equal to the username.
- Make sure that new directories and files created by the user murdoch can automatically be read and written by the user and associated group but that no other user can access them.

Exercise 6: Alias

ITA

- Creare un comando alias **per tutti gli utenti** chiamato **mycommand** che stampi il nome dell'utente che lancerà il comando e gli utenti collegati al sistema
 - Si possono utilizzare i comandi **"w"** e **"whoami"** provare a lanciali singolarmente per vederne l'output
 - Il risultato dovrà essere :

```
mycommand
hello <username>, users connected to the system:
10:48 up 1 day, 41 mins, 2 users, load averages:
1,66 1,73 1,67
USER      TTY      FROM            LOGIN@   IDLE
WHAT
luca      console  -                Lun10    24:41 -
```

ENG

- Create an alias command **for all users** called **mycommand** that prints the username of the user who will launch the command, and a list of users connected to the system
- You can use the **"w"** and **"whoami"** commands, launch them individually to see the output
- The result should be:

```
mycommand
hello <username>, users connected to the system:
10:48 up 1 day, 41 mins, 2 users, load averages:
1,66 1,73 1,67
USER      TTY      FROM            LOGIN@   IDLE
WHAT
luca      console  -                Lun10    24:41 -
```

Exercise 7: Bash script

ITA

- Create uno script bash chiamato **/exam/exercise7/random** con i seguenti dettagli:
 - Se lanciato con **/exam/exercise7/random postconf**, dovrà restituire l'output **"postroll"**
 - Se lanciato con **/exam/exercise7/random postroll**, dovrà restituire l'output **"postconf"**
 - Se lanciato con qualsiasi altro argomento dovrà restituire l'output **"/exam/exercise7/random postconf|postroll"**

ENG

- Create a bash script called **/exam/exercise7/random** with following details.
 - When run as **/exam/exercise7/random postconf**, should bring the output as **"postroll"**
 - When run as **/exam/exercise7/random postroll**, should bring the output as **"postconf"**
 - When run with any other argument or without argument, should bring the output as **"/exam/exercise7/random postconf|postroll"**

Exercise 8: Bash script

ITA

- Create uno script chiamato **/exam/exercise8/createusers**
 - Lo script accetterà come argomento un file con una lista di utenti, e dovrà creare gli utenti presenti nel file
 - Utilizzate il file presente sotto **/exam/exercise8/testuser**
 - Tutti gli utenti dovranno avere la login shell settata a **/bin/false** (man useradd), la password non è necessaria
 - Se lo script viene chiamato con qualunque altro argomento dovrà tornare il messaggio **"Input File Not Found"**
 - Se lo script viene chiamato senza argomenti dovrà tornare il messaggio **"Usage: /exam/exercise8/createusers"**
 - Per creare gli utenti utilizzate una funzione

ENG

- Create a script called **/exam/exercise8/createusers**
 - When this script is called with the argument, it should add all the users from the file
 - Use the file under **/exam/exercise8/testuser**
 - All users should have the login shell as **/bin/false**, password not required.
 - When this script is called with any other argument, it should print the message as **"Input File Not Found"**
 - When this script is run without any argument, it should display **"Usage: /exam/exercise8/createusers"**
 - To create the user use a function

Exercise 9: Firewallld

ITA

- Configurazione di port forwarding.
 - Create una regola di port forwarding sul vostro sistema in ingresso sulla porta 8081/tcp verso la porta 22/tcp.
 - Potete testare la nuova regola connettendovi via ssh al vostro sistema sulla porta 8081 con l'opzione -p 8081 del comando ssh.

ENG

- Configure port forwarding.
 - Configure your server to forward traffic incoming on port 8081/tcp from source to port on 22/tcp.
 - You can test the new firewall rule trying to connect on your system in SSH with port 8081. Option -p 8081 from your ssh client

Exercise 10: Systemd

ITA

- Installare sul vostro sistema il servizio vsftpd tramite yum
 - Il servizio deve essere attivato al boot della macchina
 - Il servizio deve essere raggiungibile dall'esterno (firewall rule)

ENG

- Install on your system vsftpd server using yum
 - The vsftpd must enable at boot
 - The vsftpd must be available from external (firewall rule)

Exercise 11: Docker

Install docker e docker-compose

- yum install -y docker
- enable docker to start at boot (systemctl enable docker)
- avviare docker (systemctl start docker)
- installare docker-compose
 - curl -L "https://github.com/docker/compose/releases/download/1.25.1/docker-compose-\$(uname -s)-\$(uname -m)" -o /usr/local/bin/docker-compose
 - chmod +x /usr/local/bin/docker-compose

ITA

- Creare una nuova immagine docker chiamata **exercise11** con tag "1.0" che esponga un server **HTTP** sulla porta **80**
- L'immagine Docker può essere qualunque ma il servizio esposto dovrà essere obbligatoriamente **Apache**

- Il contenuto della **index.html** riportato sul browser dovrà essere il vostro “**nome cognome**”
- Potete verificare che funzioni via browser collegandovi all’ip pubblico del server sulla porta **80** o **8080** (aperte in ingresso).
- L’immagine non serve rimanga attiva, l’importante è che il comando “docker run [OPTION] exercise11:1.0” attivi un apache server sulla vostra macchina raggiungibile dall’esterno.
- Salvare l’immagine Docker in un file chiamato **exercise11.tar** sotto **/exam/exercise11/**

ENG

- Create a new docker image called **exercise11** with tag “**1.0**” that exposes an **HTTP** server on **80** port.
- The Docker image can be any, but the httpd service must be **Apache**
- The content of the **index.html** must be your “**name surname**”
- You can verify that it works on browser connecting to the server's public IP on **80** or **8080** port (open on your server).
- The image doesn't needs remains active, but starting it with "docker [OPTION] exercise11:1.0" an apache server must be reachable from the outside.
- Save the Docker image created in a file called **exercise11.tar** under **/exam/exercise11**

Exercise 12: Docker compose

ITA

- Creare una applicazione wordpress mysql tramite docker-compose con le seguenti caratteristiche
 - porta locale wordpress 8080
 - DB_USER=student
 - DB_PASSWORD=stUdEnE2020
 - DB_NAME=exercise12
 - Utilizzate un bind locale sotto
 - /exam/exercise12/wordpress per wordpress
 - /exam/exercise12/mysql per mysql
- Utilizzate le immagini ufficiali dal Docker Hub (hub.docker.com)
- Il file docker-compose.yml dovrà essere messo sotto /exam/exercise12

ENG

- Create a wordpress with mysql application via docker-compose following this features
 - local port wordpress 8080
 - DB_USER=student
 - DB_PASSWORD = stUdEnt2020
 - DB_NAME = exercise12
 - Use a local binding below
 - /exam/exercise12/wordpress for wordpress

■ /exam/exercise12/mysql for mysql

- Use the official images from the Docker Hub (hub.docker.com)
- The docker-compose.yml file should be placed under /exam/exercise12

Question 1

ITA

- Da quanto visto a lezione secondo voi valgono i principali benefici di una infrastruttura virtuale in termini di scalabilità, alta affidabilità e distribuzione delle risorse.

ENG

- In your opinion, what are the main benefits of a virtual infrastructure in terms of scalability, high reliability and resource distribution.

Question 2

ITA

- In una infrastruttura (virtuale o fisica che sia) basata su tecnologia a container gestiti all'interno di un orchestratore quale è secondo voi il principale beneficio dal punto di vista funzionale.

ENG

- Inside an infrastructure (virtual or physical it does not matter) based on container technology, managed inside an orchestrator, which is in your opinion the main benefit in functionality.