

## Connessione alla propria macchina

To access your instance:

1. Open an SSH client.
2. Locate your private key file (yourname-exam6.pem).
3. Your key must not be publicly viewable for SSH to work. Use this command if needed:  
**chmod 400 yourname-exam6.pem**
4. Connect to your instance using its Public DNS:  
ec2-xx-xxx-xx-x.eu-central-1.compute.amazonaws.com
5. Use **centos** user to connect

Example:

```
ssh -i "yourname-exam6.pem"  
centos@ec2-xx-xxx-xx-x.eu-central-1.compute.amazonaws.com
```

**Switch to root user**

**[centos@ip-xxx-xx-xx-xxx ~]\$ sudo su -**

## Exercise 1: Managing Files with Shell Expansion and Command substitution

- Utilizzare l'account **student** (se non presente crearlo)
- Creare sotto il path **/exam/exercise1/** le directory **exercise\_directoryX** con **X** compreso tra 1 e 100
- all'interno di ogni directory sotto **/exam/exercise1/exercise\_directoryX** creare i files
  - **file(a..z)\_HOSTNAME\_DATE.txt**
    - con **(a..z)**= tutte le lettere dalla a alla z
    - con **HOSTNAME** = hostname del sistema (**hostname -s**)
    - **DATE** la data di creazione file nel formato Y-m-d-H:M:S (**date +%Y-%m-%d-%H:%M:%S**)

Il risultato sarà il seguente:

```
|-- exercisel_directory1
|   |-- filea_desktop_2020-08-28-00:36:47
|   |-- fileb_desktop_2020-08-28-00:36:47
|   |-- filec_desktop_2020-08-28-00:36:47
etc...
|-- exercisel_directory2
|   |-- filea_desktop_2020-08-28-00:36:47
|   |-- fileb_desktop_2020-08-28-00:36:47
|   |-- filec_desktop_2020-08-28-00:36:47
etc...
```

## Exercise 2: Managing stdout; stderr

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

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

- Create due nuovi gruppi **student** e **exam**
  - **student** con GID 4000
  - **exam** con GID 4001
- Creare gli utenti appartenenti al gruppo **exam**: **thomas** e **lucy**
  - l'utente **thomas** avrà le seguenti caratteristiche:
    - **UID 3000**
    - home directory **/home/exam/thomas**
    - password **thomas**
    - dovrà cambiare password una volta ogni 5 mesi
    - dovrà poter accedere a file e directory appartenenti al gruppo **student**
    - l'account scadrà dopo due anni dalla sua creazione
  - l'utente **lucy** avrà le seguenti caratteristiche:
    - **UID 3002**
    - home directory **/home/exam/lucy**
    - password **lucy**
    - dovrà avere come gruppo primario: **exam**
    - dovrà avere come gruppi secondari: **student** e **users**
    - dovrà avere **/bin/sh** come login shell al posto di **/bin/bash**

### Exercise 4: File permission

- 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**

## Exercise 5: alias command

- Creare i seguenti aliases commands disponibili **a tutti** gli utenti del sistema:
  - **"llc"**
    - comando: **ls -latr --color**
  - **"kernel"**
    - comando: **uname -r**

## Exercise 6: Bash script

- Create uno script chiamato **/exam/exercise6/exercise6.sh**
  - Lo script accetterà come unico argomento un numero
  - Se il numero passato è il numero 28 restituirà in output **"num is actually equal to 28"**
  - Se il numero passato è diverso da 28 restituirà in output **"num is not 28"**
  - Se non viene passato nessun argomento dovrà restituire il messaggio **"Usage:/exam/exercise6/exercise6.sh <number>"**

## Exercise 7: Bash script and yum

- Create uno script chiamato **/exam/exercise7/manageftpservice.sh**
  - Lo script accetterà in ingresso un solo argomento corrispondente alla azione da intraprendere **start;stop;restart** sul server **vsftpd**
  - Qualunque altro argomento passato dovrà restituire il messaggio **"Usage: /exam/exercise7/manageftpservice.sh start;stop;restart"**
  - ad ogni comando passato dovrà corrispondere una azione sul servizio **vsftpd** di start stop o restart.
  - installare **vsftpd** se non presente sul sistema
  - **NB. si utilizzi il case statement per la gestione dello script.**

## Exercise 8: Firewall: Configure 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. Attenzione alle regole firewall se necessarie.

## Exercise 9: Docker

Install docker e docker-compose

- Creare una nuova immagine docker chiamata **exercise9** con tag "**exam**" che esponga un server **HTTP** sulla porta **80**
- L'immagine Docker di base può essere qualunque ma il servizio esposto dovrà essere obbligatoriamente **Apache**
- Il contenuto della **index.html** riportato sul browser dovrà essere "**Hello Docker!!**"
- Lanciando l'immagine Docker con un map sulla porta locale dell'host (porta **80** oppure porta **8080**) dovrà essere possibile accedere al server apache via browser dall'esterno (Ip pubblico della vostra macchina AWS). Se serve dovrete abilitare le connessioni in ingresso per le porte 80 e 8080 in maniera permanente.
- L'immagine non serve rimanga attiva
- Salvare l'immagine Docker in un file chiamato **exercise9.tar** sotto **/exam/exercise9/**

## Exercise 10: Docker compose

- Creare un mariadb service via docker-compose con le seguenti caratteristiche.
  - db root password: "exam6"
  - db password: student
  - db name: exercise10
  - Utilizzate un bind locale per i volumi del mariadb service
    - /exam/exercise10/mariadb
- Utilizzate le immagini ufficiali dal Docker Hub ([hub.docker.com](https://hub.docker.com))
- Il file **docker-compose.yml** dovrà essere messo sotto **/exam/exercise10**

## Question 1

- Quali sono secondo voi i principali benefici di una applicazione in container rispetto ad una applicazione installate sul sistema operativo.
  - Salvare la risposta sotto **/exam/question1**

## Question 2

- Quali sono le funzioni presenti all'interno di un orchestratore di container come Kubernetes più vantaggiose rispetto ad una o più applicazioni in container gestite tramite ad esempio il tool Docker Compose
  - Salvare la risposta sotto **/exam/question2**