

Connessione alla propria macchina

To access your instance:

1. Open an SSH client.
2. Locate your private key file (yourname-exam3.pem).
3. Your key must not be publicly viewable for SSH to work. Use this command if needed:
chmod 400 name-exam3.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 "Document/name-exam3.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

- Creare sotto il path **/exam/exercise1** le directory **exercise1_directoryX** con **X** compreso tra 1 e 5
- all'interno di **/exam/exercise1/exercise1_directoryX** creare i files **fileY_DATE.txt**
 - con **Y** compreso tra 1 e 10
 - **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
|   |-- file1_2020-04-15-09:40:51.txt
|   |-- file2_2020-04-15-09:40:51.txt
|   |-- file3_2020-04-15-09:40:51.txt
|   |-- file4_2020-04-15-09:40:51.txt
|   |-- file5_2020-04-15-09:40:51.txt
|   |-- file6_2020-04-15-09:40:51.txt
|   |-- file7_2020-04-15-09:40:51.txt
|   |-- file8_2020-04-15-09:40:51.txt
|   |-- file9_2020-04-15-09:40:51.txt
|   |-- file10_2020-04-15-09:40:51.txt

|-- exercisel_directory2
|   |-- file1_2020-04-15-09:40:51.txt
|   |-- file2_2020-04-15-09:40:51.txt
|   |-- file3_2020-04-15-09:40:51.txt
|   |-- file4_2020-04-15-09:40:51.txt
|   |-- file5_2020-04-15-09:40:51.txt
etc...
```

Exercise 2: Managing pipeline and regular expression

- Utilizzare il comando **find**, e l'utente **root**, per trovare sotto la directory **/usr** tutti i file che contengono almeno un numero. Redirigere lo standard output sul file **/exam/exercise2/find.txt**.

find example

```
[centos@ip-172-31-30-213 ~]$ find /usr
/usr/
/usr/share
/usr/share/awk
/usr/share/awk/getopt.awk
/usr/share/awk/noassign.awk
/usr/share/awk/readable.awk
etc...
```

find manual

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

Exercise 3: User and Group

- Create due nuovi gruppi **teachers** e **students**
 - **teachers** con GID 3000
 - **students** con GID 3001
- Creare l'utente appartenente al gruppo **teachers**: **cavatorta**
 - l'utente **cavatorta** avrà le seguenti caratteristiche:
 - **UID 3000**
 - home directory **/home/exercise3/teachers/cavatorta**
 - password **cavatorta**
 - dovrà cambiare password una volta ogni 6 mesi
 - dovrà poter accedere a file e directory di tutti gli utenti appartenenti al gruppo **students**
- Creare gli utenti appartenenti al gruppo **students**: **patelli**; **taverna**
 - L'utente **patelli** avrà le seguenti caratteristiche:
 - **UID 3010**
 - dovrà cambiare password al primo accesso e successivamente una volta ogni due mesi
 - l'account scadrà dopo un anno dalla sua creazione
 - l'utente **taverna** avrà le seguenti caratteristiche:
 - **UID 3020**
 - home directory **/home/exercise3/taverna**
 - password **taverna**
 - dovrà avere come gruppi secondari: **students** e **users**
 - dovrà utilizzare **/bin/sh** come login shell, al posto di **/bin/bash**

Exercise 4: alias command

- Creare i seguenti aliases commands disponibili a **tutti** gli utenti del sistema:
 - **"llc"**
 - comando: **ls -latr --color**
 - **"kernel"**
 - comando: **uname -r**
- Creare un alias command disponibile a tutti gli utenti del sistema chiamato **"psu"** il quale restituisca la lista di tutti i processi attivi del solo utente che lancia il comando.

Exercise 5: File permission

- Creare l'utente **bezos** appartenente al gruppo **amazon**.
- Fare in modo che nuovi files e directory creati dall'utente **bezos** alla creazione, di default non possano essere letti scritti o visti, da nessuno al di fuori dell'utente **bezos**.

Exercise 6: Bash script

- Create uno script bash sotto **/exam/exercise6** chiamato **numbers.sh** con le seguenti caratteristiche:
 - Accetti in ingresso due numeri come argomenti
 - Stampi "The larger of the two numbers is: NUMBER" NUMBER=il maggiore dei due numeri passati come argomento.
 - Se passati un numero di argomenti diversi da 2 stampi: "only two numbers!!" ed esca.
 - Non sono richiesti ulteriori controlli a quelli indicati sugli argomenti passati in ingresso

```
bash /exam/exercise6/numbers.sh 40 45
The larger of the two numbers is: 45
bash number.sh 40 45 50
only two numbers!!
```

Exercise 7: Bash script

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

Exercise 8: Firewallld

- Create una nuova regola firewall per il servizio http che vada ad aggiungere ai log delle connessioni, verso il servizio http il prefisso “**Exercise8_httpd**”
- Limitate il log ad un massimo di 2 nuovi messaggi per secondo

Exercise 9: Docker

- Installare Docker
- Renderlo attivo al boot della macchina

Exercise 10: Dockerfile

- Creare una nuova immagine docker chiamata **exam/exercise10:1.0** che esponga un server **HTTP** sulla porta **80** basato sulla ultima versione della immagine di base **centos**
- L'immagine dovrà essere creata a partire dal **Dockerfile** sotto **/exam/exercise10**
- Creare il file **index.html** sotto **/exam/exercise10** contenente la stringa “<h1>Hello Apache from Docker!!</h1>” da inserire nella DocumentRoot di Apache in fase di build della immagine.
- Fare in modo che una volta avviata l'immagine **exam/exercise10:1.0** sia possibile la connessione tramite IP pubblico del sistema AWS
 - <http://ec2-xx-xxx-xx-x.eu-central-1.compute.amazonaws.com:PORT>
 - **PORT**: si potranno utilizzare le porte 8080 e 8081.

example

<http://ec2-xx-xxx-xx-x.eu-central-1.compute.amazonaws.com:8080>

Hello Apache from Docker!!

Question 1

- Quali sono secondo voi i principali benefici e differenze, nella gestione di una applicazione basata sui soli container che interagiscono tra di loro, e la stessa all'interno di una soluzione in un cluster di container come Kubernetes?
 - Salvare la risposta sotto **/exam/question1/**