

## Connessione alla propria macchina

1. Aprire il client SSH .
2. Posizionarsi nella directory di salvataggio del file .pem allegato alla mail.
3. Tramite il client SSH, utilizza il seguente comando al fine di impostare le autorizzazioni del file della chiave privata in **read only** per l'utente owner.  
**chmod 400 keyPairName.pem**
4. Nella finestra del terminale, utilizzare il comando ssh per connettersi all'istanza. Specificare il percorso e il nome del file della chiave privata (.pem), il nome utente per l'istanza e il nome DNS pubblico per l'istanza.
5. Utilizzare la username **centos** per la connessione al sistema

Esempio:

```
ssh -i "keyPairName.pem"
```

```
centos@ec2-xx-xxx-xx-x.eu-central-1.compute.amazonaws.com
```

Per passare all'utente **root** utilizzare il comando "**sudo su -**"

```
[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)

## Exercise 2: Special file permission

- Creare una directory sotto `/exam/exercise2` dove gli utenti che possono accedere al gruppo **collaboration** potranno condividere files
- Tutti i file creati sotto la directory `/exam/exercise2` dovranno essere assegnati automaticamente al gruppo **collaboration**

## Exercise 3: User and Group

- Create due nuovi gruppi **students** e **exam**
  - **students** con GID 2100
  - **exam** con GID 2101
- Creare gli utenti appartenenti al gruppo **exam**: **rossi** e **morini**
  - l'utente **rossi** avrà le seguenti caratteristiche:
    - **UID** 3100
    - home directory `/home/exam/rossi`
    - dovrà cambiare password al primo accesso
    - dovrà poter accedere a file e directory appartenenti al gruppo **students**
  - l'utente **morini** avrà le seguenti caratteristiche:
    - **UID** 3101
    - dovrà cambiare password al primo accesso e successivamente una volta al mese
    - gruppi secondari **users gamers** e **wheel**
    - utilizzare come shell di default **sh** al posto di **bash**

## Exercise 4: alias command

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

## Exercise 5: Systemd

- Installare sul vostro sistema il servizio vsftpd
- Il servizio deve essere attivo al boot della macchina

## Exercise 6: Bash script

- Create uno script bash sotto **/exam/exercise6** chiamato **check\_arguments.sh** con le seguenti caratteristiche:
  - accetti in ingresso una lista di argomenti contenenti una lista di path verso file e/o directory
  - lo script dovrà analizzare la lista di TUTTI gli argomenti passati e se l'argomento passato è:
    - un **file** stampi la stringa: **ArgName is a regular file**
    - una **directory** stampi la stringa: **ArgName is a directory**
    - un puntamento a qualcosa di **non esistente**: **ArgName File or Directory not found**
    - **altrimenti** dovrà stampare la stringa: **is another type of file**
    - se non verrà passato nulla in ingresso allo script stamperà un generico messaggio di errore

### example:

```
# bash /exam/exercise6/check_arguments.sh /etc/passwd /etc
/usr/tmp /foo
```

```
/etc/passwd is a regular file
/etc is a directory
/usr/tmp is another type of file
/foo File or Directory not found
```

## Exercise 7: Docker

- La directory **/exam/exercise7** dovrà contenere i files:
  - **Dockerfile-sidecar**
  - **Dockerfile-httpd**
  - **sidecar.sh**
  - **docker-compose.yml**
- **Dockerfile-sidecar** il compito del container sarà quello di scaricare ad intervalli di 1 ora il contenuto del repository GitHub: <https://github.com/lcavator/exam3.git>
  - l'immagine di base da utilizzare sarà: **cavatortaluca/exam:centos8**
  - il contenuto sarà scaricato su directory a piacere all'interno del container in bind con il sistema host sul path: **/exam/exercise7/httpd/data**
  - dovrete installare il software **git** come unico prerequisito alla immagine sidecar (**yum install -y git**)
- **sidecar.sh** sarà lo script bash che avrà in carico l'aggiornamento del repository git ogni ora.
  - il comando per scaricare dal repository GitHub è il seguente:
    - Fase di inizializzazione **solo la prima volta**:

```
cd /path/inside/container
git init
git remote add origin https://github.com/lcavator/exam3.git
git fetch
```

- Comando per aggiornare il contenuto:

```
git pull origin master
```

- **Dockerfile-httpd** il compito del container sarà quello di esporre un server **httpd** la cui DocumentRoot monterà in bind dal sistema host la directory **/exam/exercise7/httpd/data** in **read only** mode (il container non avrà possibilità di scrivere su **/exam/exercise7/httpd/data**)
  - potete utilizzare l'immagine che volete (se centos8 utilizzate **cavatortaluca/exam:centos8**)
- **docker-compose.yml** verrà utilizzato per avviare entrambi i servizi in container, ed eventualmente ma non obbligatorio gestirne la build
  - Il container httpd dovrà essere raggiunto dal sistema host su porta **8080**
  - N.B per montare in read only mode una directory, passarla al container nel seguente modo

```
volumes:
  - /path/on/host:/path/inside/container:ro
```

### esempio:

```
# curl localhost:8080
Hello exam3!!!!!!!!!!
```

## Question :

- Doveste avviare i container descritti nell'esercizio 7 all'interno di un cluster Kubernetes. Di quali oggetti avreste bisogno sulla base di quanto visto durante il

corso? Se ne dia una breve descrizione e motivazione in questo specifico caso a parole.