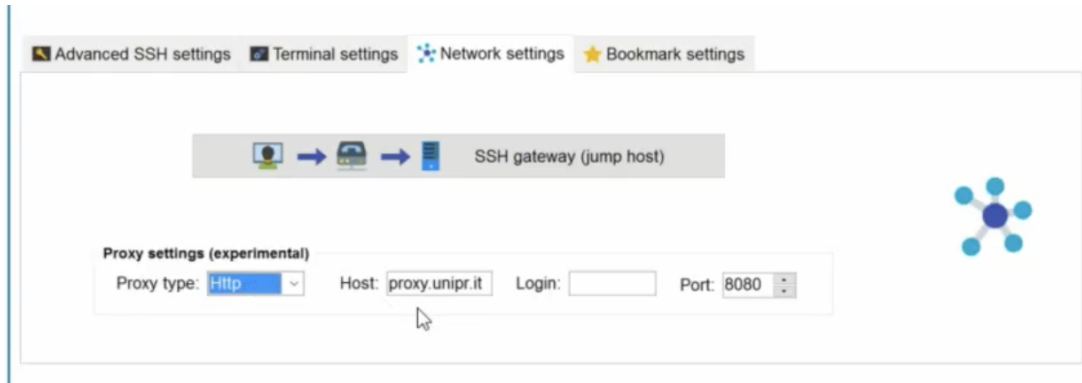
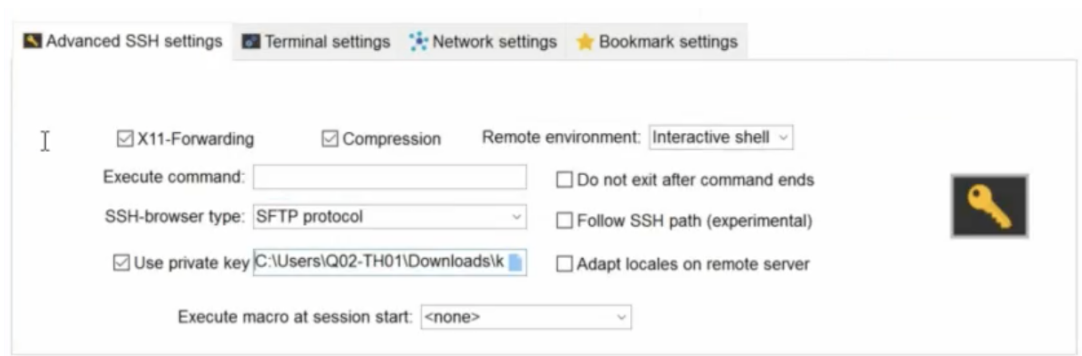


Connessione alla propria macchina

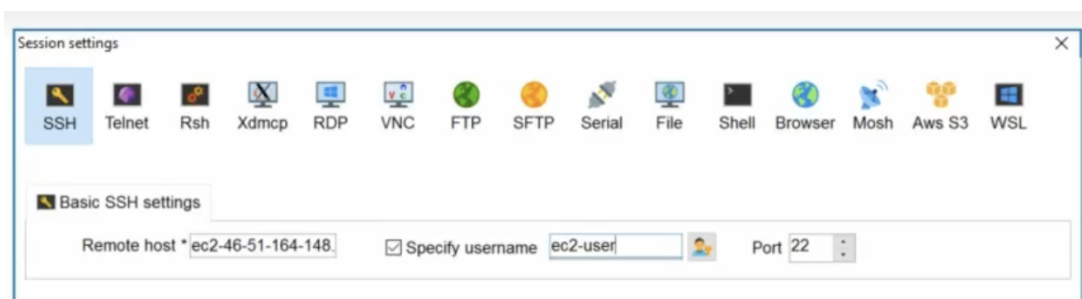
1. Aprire MobaXterm.
2. Tab session / SSH
3. Configurare il proxy dal tab Network Setting
 - a. Httpd
 - b. proxy.unipr.it
 - c. 8080



4. Passare la chiave SSH dal tab “Advanced SSH settings”



5. Inserire in Remote Hosts il dns della vostra istanza Linux
 - a. esempio: ec2-xx-xxx-xx-x.eu-central-1.compute.amazonaws.com
6. Username: **ec2-user**
7. port: **22**



Video accesso ai sistemi cloud dalle postazioni UNIPR:

[accesso_aws_pld_laboratori.mov](#)

Per passare all'utente **root** utilizzare il comando "**sudo su -**"
[ec2-user@ip-xxx-xx-xx-xxx ~]\$ **sudo su -**

Exercise 1: File permission and users

- Creare uno script bash che stampi in output “Hi! how are you?” chiamato **hi**. Il file dovrà appartenere all’utente e gruppo **root**.
- Assicuratevi possa essere lanciato da **qualunque** utente del sistema come comando

example:

```
[student@ip-172-31-35-174 ~]$ hello
hello world!!
[root@ip-172-31-35-174 exercise1]# hello
hello world!!
```

Exercise 2: Shell environment

- Creare due nuove variabili di ambiente (persistenti al login):
 - **HELLO_USER** per il solo utente **student** contenente la stringa “hello student”.
 - **HELLO_ALL_USERS** per tutti gli utenti di sistema contenente la stringa “hello all users”

Exercise 3: File system permission and user

- Creare l’utente **june** appartenente al gruppo primario **months**.
- La home directory dell’utente sarà **/home/months/june**, fare in modo che la directory sia presente sul sistema con i giusti permessi.

Exercise 4: HTTPD

- Installare sul sistema il servizio HTTP/Apache
- Questo server web dovrà visualizzare il contenuto "Hello exercise4" quando viene richiesto l'URL **http://www.exercise4.myexam.com/**
- il testo "Hello class" in ogni altro caso
- creare le corrette entry in **/etc/hosts** per risolvere **www.exercise4.myexam.com** sull'ip privato del sistema (ip a s)
- Potete verificare che il tutto funzioni tramite curl

```
[root@server ~]# curl http://www.exercise4.myexam.com
Hello exercise4!
[root@server ~]# curl http://localhost
Hello class
```

Exercise 5: Bash script

- Create uno script bash sotto **/exam/exercise5** chiamato **bash_wrapper.sh** con le seguenti caratteristiche:
 - accetti in ingresso due parametri
 - effettuare gli opportuni controlli in caso i parametri passati siano diversi da due stampando un errore a piacere e uscendo
 - il primo parametro sarà il comando bash da utilizzare
 - verificate che il comando esista altrimenti uscite con un errore a piacere
 - il secondo parametro sarà l'argomento del comando ed eventuali opzioni

example:

```
bash /exam/exercise5/bash_wrapper.sh ls /home
student    user1      user2
bash /exam/exercise5/bash_wrapper.sh ls "-l /home"
drwxr-xr-x  2 student  student  64 29 Giu 22:57 student
drwxr-xr-x  2 user1    user1    64 29 Giu 22:57 user1
drwxr-xr-x  2 user2    user2    64 29 Giu 22:57 user2
bash /exam/exercise5/bash_wrapper.sh echo "hello world!!"
hello world!!
```

Exercise 6: Docker

- Creare la propria immagine a partire da un'immagine di base a piacere
- La directory **/exam/exercise6** dovrà contenere:
 - **Dockerfile**
 - **logger.sh**
 - **docker-compose.yml**
 - **loggerdir** (questa è una directory)
- **logger.sh** sarà lo script di entrypoint il cui compito sarà quello di stampare all'interno di un file chiamato **logger.log** la stringa "hello docker" NUMBER volte e poi uscire
 - NUMBER sarà una variabile di ambiente contenente un numero di default a piacere.
- **docker-compose.yml** lo utilizzerete per:
 - gestire un bind locale con la directory **loggerdir** e il container, che conterrà il file **logger.log**
 - la build della immagine

Question :

le risposte andranno sotto la directory **/exam/question/**

- Q1: Cosa si intende per *service as a service* e perché non possiamo considerare kubernetes un servizio di questo tipo.
- Si dia una breve spiegazione di cosa è l'oggetto kubernetes chiamato **service**