

ssh -l root 127.0.0.1 -p 2222 --> Crea la connessione al root

BASH

#!/bin/bash --> Ogni file di bash parte così

IF

if [...]; then

...
elif [...]; then

...
fi

if ["\$#" -eq 2] --> Verificare i parametri lanciati in esecuzione siano 2 (-eq si usa con i numeri)

if [! -f "\$1"] --> Verificare il primo parametro non sia un file

if ["\$2" != "append"] --> Verifica che il secondo parametro non sia "append"

NEI BASH RICORDARSI CHE SERVONO TANTE PARENTESI QUANTI SONO GLI OPERATORI

if [\$? -eq 0] --> Verifica l'operazione precedente sia andata a buon fine

Es. variabile --> exists=\$(grep "\$line:/bin/bash" /etc/passwd)
grep funziona come gli includes di Java... cerca la parola specificata (\$line:/bin/bash) in un file (passwd)

if ! [["\$var" =~ ^[0-9]+\$]] --> Per verificare che una variabile sia un numero

LETTURA DI FILE

while read line

do
...
done < file.txt

chmod u+x exercise5index.sh --> Dare al file di bash i permessi d'esecuzione
mkdir -p "\$var" --> Per creare una cartella (solo se non già esistente) con il nome contenuto nella variabile var

CAMBIO PASSWORD

chage -d 0 mark --> Obbliga a cambiare password al prossimo accesso

chage -l mark --> Ti dice quali sono le caratteristiche dell'account mark

chage -m 30 mark --> Obbliga a cambiare password ogni 30 giorni

passwd --expire --> Obbliga a cambiare password al prossimo accesso

- PER L'ESAME:
- SELEZIONARE ESAMI E TEST CON SOFTWARE
 - APRO MOBAXTERM
 - APRO ELLY
 - PRENDO LA CHIAVE SUL PORTALE E ELLY, LA COPIO E LA INCOLLO IN UN NUOVO FILE CHE CHIAMERO' key.pem
 - PRENDO L'INDIRIZZO DELLA MACCHINA IN CLOUD E LO COPIO
 - PRENDO MOBAXTERM E CLICCO SU SESSION, POI SSH
 - INCOLLO IN REMOTE HOST L'INDIRIZZO DELLA MACCHINA IN CLOUD
 - SPECIFICO L'UTENTE: ec2-user
 - CLICCO ADVANCED SSH SETTINGS E FLAGGO «USE PRIVATE KEY», E SELEZIONO key.pem
 - POI CLICCO NETWORK SETTINGS E NELLA COMBOBOX PROXY TYPE, SCELGO HTTP
 - PER DIVENTARE ROOT: sudo su -

ALIAS

Aprire il file '/etc/profile' --> File contenente i vari Alias

In coda al file scrivere nomeAlias = 'echo "user: \$USER - working directory: \$(pwd) - home_directory: \$HOME"'
--> Questo alias stampa a schermo il nome dell'utente che lo manda in esecuzione, la working station e il percorso

Salvare e uscire dal file con :wq

source /etc/profile --> Aggiorni l'ambiente

unalias ciao --> Per eliminare l'alias ciao

FIND

find /usr -name '[0-9]*'
--> Trova tutti i file che hanno un numero nel nome

find /usr -name '[0-9]*' >> ~/Exam/exercise1/find.txt
--> Trova tutti i file che hanno un numero nel nome e ridirige l'output del file find.txt

find /var -name "[abc]*" -exec file {} \; > ~/Exam2/exercise1/find.txt 2> ~/Exam2/exercise1/error.txt
--> -exec file () \; esegue il comando file per ogni file trovato

find /usr -name "ab" -o -name "ac" -o -name "li"
--> Usare l'operatore OR nel find

UMASK ESEMPIO

es. umask 755

Proprietario: $rwx = 4 + 2 + 1 = 7$
Gruppo: $r-x = 4 + 0 + 1 = 5$
Altro: $r-x = 4 + 0 + 1 = 5$

DA SCRIVERE SEMPRE NELLA CARTELLA /home/nomeUser, NEL FILE .bashrc
PER VERIFICARE LO STATO ATTUALE DELL'umask... SCRIVERE umask

GRUPPI

Cos'è un gruppo? E' un insieme di utenti che condividono gli stessi permessi su una serie di risorse, file e directory

groupadd collaboration --> Crea un gruppo "collaboration"

chgrp -R collaboration /exam/exercise2 --> Associa la directory /exam/exercise2 e suoi contenuti (-R) al gruppo "collaboration"

chmod g+s /home/students/collaboration --> Associa i futuri file creati in /home/students, al gruppo collaboration

ls -l --> Per scoprire a quale gruppo appartiene il file

chmod -R g=rwx --> Imposta i permessi del gruppo (Lettura = si, Scrittura = si, Esecuzione = si) / chmod -R g+rwx

chmod g+s /exam/exercise2 --> Associa i futuri file creati in /exam/exercise2, al gruppo associato alla directory

UTENTI

GLI UTENTI CHE CREO LI VEDO NELLA CARTELLA /home

Per verificare se un utente esiste → id nomeUser

rm -r /exercise3 --> Rimuovere una directory utente

userdel -r nomeUser --> Rimuove utente e sottocartelle

usermod -d /new/home/directory john --> Per cambiare la home directory dell'utente john

usermod -g nomeGruppo nomeUser --> Per aggiungere l'utente al gruppo principale

usermod -a -G nomeGruppo nomeUser --> Per aggiungere l'utente al gruppo
//L'opzione "-a" aggiunge l'utente al gruppo senza rimuoverlo da altri gruppi a cui appartiene, mentre "-G" specifica il nome del gruppo.

setfacl -d -m o::--- /home/exercise3 --> Imposta le ACL predefinite (-d) per la directory home dell'utente exercise3 (-m) in modo che gli utenti others non abbiano alcun permesso (o::---).

umask 077 --> I nuovi file e directory creati dall'utente exercise3 non avranno permessi associati agli utenti others

useradd -m nomeUser --> crea un nuovo utente (-m) con la directory home predefinita

useradd -s /bin/bash nomeUser --> creo un utente e gli attribuisco la bash di default così da poter eseguire qualche comando

SISTEMA E INSTALLAZIONI

dnf install vsftpd --> Per installare vsftpd sul sistema
yum install vsftpd --> Per installare vsftpd sul sistema

systemctl enable vsftpd --> Abilita il servizio vsftpd sul sistema (al boot della macchina)

systemctl start vsftpd --> Avvia il servizio sul sistema <-- Per sopprimere l'output: systemctl start vsftpd > /dev/null

systemctl status vsftpd --> Verifica lo status del servizio

systemctl restart vsftpd --> Restarta il servizio

Configurare il server FTP per consentire solo l'accesso agli utenti locali e impedire l'accesso anonimo:
Aprire il file etc/vsftpd/vsftpd.conf e verificare siano presenti le seguenti linee:

- anonymous_enable=NO
- local_enable=YES
- write_enable=YES
- chroot_local_user=YES

ID/ATTRIBUTI A GRUPPI E UTENTI

groupadd -g 3005 students --> Crea un gruppo con GID 3005

useradd -u 3008 -g students -d /home/students/erossi -m -p erossi -e \$(date -d "+30 days" +"%Y-%m-%d") erossi
--> Crea un utente erossi con:

- u specifica l'UID dell'utente
- g specifica il GID del gruppo primario dell'utente
- d specifica la directory home dell'utente
- m crea la directory home dell'utente se non esiste
- p specifica la password dell'utente
- e specifica la data di scadenza dell'account (in questo caso 30 giorni dopo la data di creazione dell'account)

sudo useradd -u 3010 -g students -G collaboration -d /home/students/gbianchi -m -s /bin/sh -p '*' -e \$(date -d "+1 year" +"%Y-%m-%d") gbianchi
--> Crea l'utente gbianchi con:

- G specifica il gruppo supplementare dell'utente (in questo caso il gruppo collaboration)
- p '*' forza l'utente a cambiare la password al primo accesso
- e specifica la data di scadenza dell'account (in questo caso 1 anno dopo la data di creazione dell'account)
- s utilizza come shell di default sh al posto di bash

FIREWALL

firewall-cmd --state

--> Questo comando controlla lo stato del firewall sul tuo sistema Linux. Se il firewall è attivo, il comando restituirà lo stato attuale del firewall. In caso contrario, il comando restituirà un messaggio di errore.

firewall-cmd --permanent --add-port=8081/tcp

--> Questo comando apre la porta 8081/TCP per l'accesso al sistema. La regola ha esito permanente. Se si vuole rimuovere una regola basta sostituire "add" con "remove"

firewall-cmd --reload --> Riavvia il Firewall, per consentire le modifiche

firewall-cmd --list-all

--> Elenca tutte le regole del firewall attive. Se la porta 8081/TCP è elencata, allora la regola è stata applicata correttamente.

ALTRI COMANDI

ps -u \$USER --> Per la lista dei processi relativi all'utente corrente

touch ~/Exam2/exercise2/directory{1..15}/exam2_file{1..9}_\$(date +%y-%m-%d).txt

--> Crea nel percorso " ~/Exam2/exercise2/directory{1..15} ": 10 file .txt per ogni directory

touch season{1..9}_episode{1..12}.ogg

--> Crea 108 file: season da 1 a 9 e _episode da 1 a 12.

rm -rf exercise1

--> Elimina tutti file dentro exercise1 anche non vuoti.

mkdir directory{1..15}

--> Crea delle directory chiamate: directory1, directory2, directory3,... directory15.

chage -E \$(date -d +365days +%Y-%m-%d) wolverine

--> Scadenza dell'utente.

chmod g+s nomecartella/

--> Ogni cosa creata all'interno di nomecartella viene assegnata al gr

envsubst < ~/directory/template.conf > /etc/httpd/conf/httpd.conf

--> Popola il template di template.conf con i valori contenuti in httpd.conf



TEMPLATE

FUNZIONI

<pre>function saluta_persona() { nome=\$1 echo "Ciao, \$nome!" }</pre>	←	ESEMPIO FUNZIONE	→	<pre>function somma() { num1=\$1 num2=\$2 risultato=\$((num1 + num2)) echo \$risultato }</pre>
<pre>saluta_persona "Mario"</pre>	←	ESEMPIO CHIAMATA A FUNZIONE	→	<pre>risultato=\$(somma 3 5)</pre>

LOOP

<pre>for var in "\$@" do ... done</pre>	←	Ciclo for sui parametri dati alla chiamata del bash
	←	Sintassi del FOR
<pre>if ((var % 2 == 0)) then ((even += var)) else ((odd += var)) fi</pre>	←	Classico contenuto di un ciclo per la somma dei numeri pari / dei numeri dispari

REGULAR EXPRESSION

<pre>grep '^([c m f])an\$' regexpr.txt</pre>	←	Esempio di come usare un grep con una regular expression che seleziona del parole che iniziano o per ‘c’ o per ‘m’ o per ‘f’ e sono seguite da ‘an’
--	---	---