Born2beroot:

- Esta guía esta diseñada para 42Cursus-Born2beroot.
- Esta guía supone que ya tienes instalada una máquina virtual debian.

Por favor, lee la sección de notas antes de empezar.

Configuración:

Sudo:

Usamos sudo para ejecutar comandos de superusuario sin ser root. Esto es, en general, más seguro que ser siempre root.

- Instalación de sudo:

```
su -
apt install sudo
```

- Add user to sudo group:

  Este paso nos hará pertenecer al grupo sudo.

  Este paso nos permitirá ejecutar commandos con SSH en el futuro cercano.

```
su -
usermod -aG sudo USER
```

Si hecho de manera correcta, deberíamos ver que el usuario pertenece al grupo con el comando:

```
getent group sudo
```

Dar al usuario privilegios de superusuario:

- Abre el archivo `/etc/sudoers` y agrega la siguiente línea:

```
su -
visudo
```

- Añade esta línea si no está ya:

```
%sudo ALL=(ALL) ALL
```

(un buen sitio es justo debajo de esta:)

```
root  ALL=(ALL) ALL
```

- Guarda y sal del archivo. Si está hecho de manera correcta, puedes logearte de nuevo con tu cuenta para verificar si ha funcionado.
- Por ejemplo, ahora deberías ser capaz de ejecutar este comando sin ser root:

```
apt update # Este no funcionará
sudo apt update
```

Instalación de herramientas:

Necesitamos instalar algunas herramientas que son esenciales:

Actualizando los paquetes/sistema actual:

```
sudo apt update && sudo apt upgrade -y
```

Instalar las herramientas:

- git:

```
sudo apt install git -y
```

- wget o curl:
    - Ambas herramientas nos permiten descargar archivos de internet a través de su URL.
    - No son 100% necesarias para el proyecto pero son útiles.
    - En mi caso, he usado wget usando:

      sudo apt install wget -y

- Herramientas de personalización:
    - Este paso es opcional.
    - En mi caso, para trabajar de una manera más cómoda y rápida:
        - He instalado:
            - man:

              ```
              sudo apt install man -y
              ```

            - vim:

              ```
              sudo apt install vim -y
              ```

            - zsh:

              ```
              sudo apt install zsh -y
              ```

            - Oh my zsh
        - He editado tanto **~/.zshrc** y **~/.vimrc** con la configuración básica que necesito para trabajar de una manera rápida e inteligente.
        - Ten en cuenta que estas herramientas son ligeras y fáciles de quitar si fuera necesario, con lo que puedes quitarlo en cualquier momento.
    - Recuerda que instalar una interfaz gráfica está prohibido.

Configurando el servicio SSH:

Este paso nos permitirá conectarnos a la máquina virtual a través de un terminal de nuestro ordenador. Esto es muy bueno para poder copiar y pegar contenido entre ambas máquinas.

Instalación de SSH:

```
sudo apt update && sudo apt install openssh-server -y
```

Comandos útiles de SSH:

Nombre	Comando	Descripción
Ver estado ssh	`sudo systemctl status ssh`	Muestra el estado actual del servicio SSH.
Reiniciar servicio SSH	`sudo service ssh restart`	Reinicia el servicio SSH.
Check port settings	`sudo grep Port /etc/ssh/sshd_config`	Nos permite ver los puertos configurados en la configuración (NO DEL SERVICIO).

Configuración:

- Ver estado ssh
- Reinicia el servicio SSH
- Cambia el puerto por defecto (22) al 4242:
    - Abre con sudo el archivo de configuración:

      sudo vim /etc/ssh/sshd_config
    - Encuentra la línea:

      ```
      #Port 22
      ```
    - Cambia el valor por:

      ```
      Port 4242
      ```
    - Guarda y sal del archivo (verifica que se ha editado correctamente).

- Reinicia el servicio:
  - Si usas Ver estado ssh otra vez, verás que nada ha cambiado. Esto es porque los cambios no tendrán efecto hasta que el servicio se reinicie. Por tanto, Reinicia el servicio SSH.
  - Si todo ha ido bien, es posible ver en el resultado de Ver estado ssh que el servidor está ahora escuchando por el puerto 4242.
  - También puedes ver que el ID ha cambiado de manera esperada.
  - Ejemplo:

```
sudo systemctl status ssh | grep port
```

```
  DATE MACHINE_NAME sshd[ID]: Server listening on 0.0.0.0 port
22.
  DATE MACHINE_NAME sshd[ID]: Server listening on :: port 22.
```

```
sudo service ssh restart
sudo systemctl status ssh | grep port
```

```
  DATE MACHINE_NAME sshd[ID]: Server listening on 0.0.0.0 port
4242.
  DATE MACHINE_NAME sshd[ID]: Server listening on :: port 4242.
```

Configuración del firewall:

Instalación del firewall:

```
sudo apt update && sudo apt install ufw -y
```

Comandos útiles de UFW:

Nombre	Comando	Descripción
Activar UFW	`sudo ufw enable`	Activa UFW y lo configura para que se active cada vez que se inicie el servidor.
Ver estado UFW	`sudo ufw status numbered`	Muestra el estado actual y las reglas de UFW. El parámetro *numbered* nos muestra el índice de cada una (PORT_ID)
Permite SSH	`sudo ufw allow ssh`	Permite el uso de SSH
Abre puerto	`sudo ufw allow` PORT	Abre el puerto dado (ej: 4242)
Quita puerto	`sudo ufw delete` PORT_ID	Quita el puerto seleccionado (usando el índice que nos da el comando `sudo ufw status numbered`)

Configuración UFW:

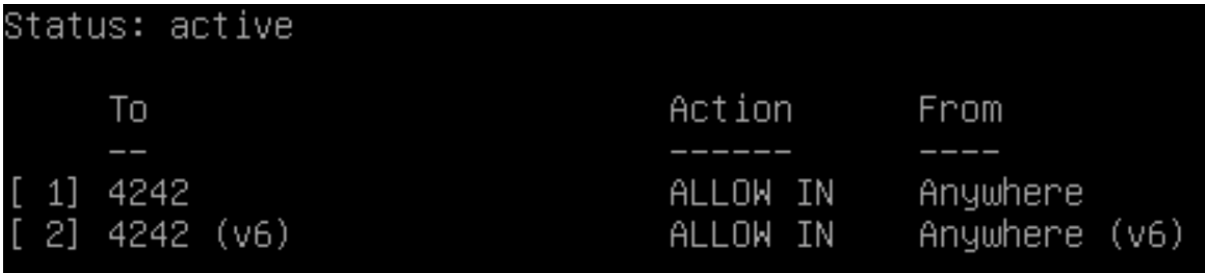
- Activar UFW
- Ver estado UFW
- Permite SSH

  ```
  sudo ufw allow ssh
  ```

- Configura las reglas de los puertos:
  - Abre el puerto 4242:

    sudo ufw allow 4242

- Elimina todos los otros puertos. Si hecho correctamente, debería quedar:



Permitir la conexión SSH usando Virtualbox:

- Ve a Virtualbox -> Choose the VM -> Select Settings
- Elige "Network" -> "Adapter 1" -> "Avanced" -> "Port forwarding"
- Add a new one with the following rules:

Name	Protocol	Host IP	Host Port	Guest IP	Guest Port
SSH	TCP		4242		4242

- Para que se apliquen los cambios, Reinicia el servicio SSH.

```
sudo systemctl restart ssh
```

- Ya está! Ahora podemos conectarnos a la máquina virtual desde nuestro ordenador. Desde ahora, podemos usar SSH para copiar y pegar contenido entre ambas máquinas.

```
ssh USER@localhost -p 4242
```

o

```
ssh USER@127.0.0.1 -p 4242
```

Configurar política de contraseñas:

Este paso nos permite requerir ciertas condiciones a las contraseñas que se generen a partir de ahora.

- Instala la librería que verifica la integridad de las contraseñas:

```
sudo apt-get install libpam-pwquality
```

- Cambia las reglas de la calidad de las contraseñas:
  - Abre el archivo:

```
sudo vi /etc/pam.d/common-password
```

  - Encuentra la línea:

```
password [success=1 default=ignore] pam_unix.so obscure sha512
```

  - Añade lo siguiente:

```
password [success=1 default=ignore] pam_unix.so obscure
use_authtok try_first_pass sha512 minlen=10
```

Elemento	Descripción
`obscure`	Realiza algunos test a la contraseña: palíndromo, diferenciar mayúsculas de minúsculas...
`use_authtok`	Si hay alguna contraseña pendiente, usa esa contraseña antes de usar la nueva.
`try_first_pass`	Antes de cambiar la contraseña, verifica que las anteriores contraseñas cumplen la norma también.
`sha512`	Usa este tipo de encriptación.
`minlen=`N	La longitud mínima de la contraseña es N.

  - Configura el resto de los ajustes. Encuentra la línea:

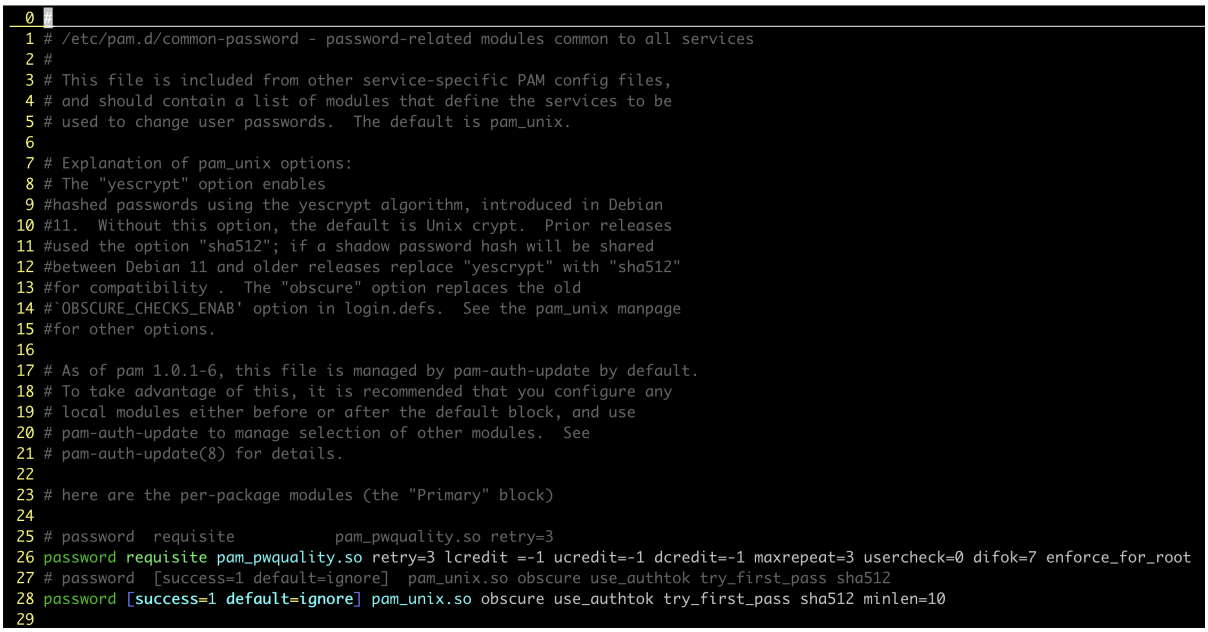
```
password requisite pam_pwquality.so retry=3
```

- Añade lo siguiente:

```
  password requisite pam_pwquality.so retry=3 lcredit =-1
ucredit=-1 dcredit=-1 maxrepeat=3 usercheck=0 difok=7
enforce_for_root
```

Elemento	Descripción
`lcredit=`N	Minimum number of *lower-case* characters.
`ucredit=`N	Minimum number of *upper-case* characters.
`dcredit=`N	Minimum number of *digit* characters.
`maxrepeat=`N	Maximun character repetition.
`usercheck=`N	If the password can contain the user name in some form (1: ON, 0: OFF).
`difok=`N	Minimum number of chararters that must be different from the previous password.
`enforce_for_root`	This rules also apply for root users.

  - Deberías terminar con algo parecido a esto:



```
# /etc/pam.d/common-password - password-related modules common to all services
#
# This file is included from other service-specific PAM config files,
# and should contain a list of modules that define the services to be
# used to change user passwords.  The default is pam_unix.

# Explanation of pam_unix options:
# The "yescrypt" option enables
#hashed passwords using the yescrypt algorithm, introduced in Debian
#11.  Without this option, the default is Unix crypt.  Prior releases
#used the option "sha512"; if a shadow password hash will be shared
#between Debian 11 and older releases replace "yescrypt" with "sha512"
#for compatibility .  The "obscure" option replaces the old
#`OBSCURE_CHECKS_ENAB' option in login.defs.  See the pam_unix manpage
#for other options.

# As of pam 1.0.1-6, this file is managed by pam-auth-update by default.
# To take advantage of this, it is recommended that you configure any
# local modules either before or after the default block, and use
# pam-auth-update to manage selection of other modules.  See
# pam-auth-update(8) for details.

# here are the per-package modules (the "Primary" block)

# password  requisite          pam_pwquality.so retry=3
password requisite pam_pwquality.so retry=3 lcredit =-1 ucredit=-1 dcredit=-1 maxrepeat=3 usercheck=0 difok=7 enforce_for_root
# password  [success=1 default=ignore]  pam_unix.so obscure use_authtok try_first_pass sha512
password [success=1 default=ignore] pam_unix.so obscure use_authtok try_first_pass sha512 minlen=10

```

- Cambia las reglas de expiración/caducidad:

  - Abre el archivo:

```
sudo vi /etc/login.defs
```

- Modifica las siguientes líneas:

```
PASS_MAX_DAYS 9999
PASS_MIN_DAYS 0
PASS_WARN_AGE 7
```

Elemento	Descripción
PASS_MAX_DAYS N	Vida máxima de una contraseña en días.
PASS_MIN_DAYS N	Mínima vida de una contraseña (0 to disable).
PASS_WARN_AGE N	Recibe una notificación N días antes de cambiar la contraseña.

- En mi caso, terminé con:

```
PASS_MAX_DAYS 30
PASS_MIN_DAYS 2
PASS_WARN_AGE 7
```

- Ya está! Reinicia la máquina virtual para aplicar los cambios.

```
sudo reboot
```

Desde ahora, cada usuario que **creemos** seguirá estas normas.

- Si ejecutamos:

```
chage -l USER
```

y

```
sudo chage -l root
```

Verás que la configuración de la caducidad de las contraseñas de ambos usuarios no ha cambiado. Para cambiarla:

```
sudo chage USER
```

y

```
sudo chage root
```

- Ejemplo de ejecución:

```
➜  ~ sudo chage jre-gonz42
Changing the aging information for jre-gonz42
Enter the new value, or press ENTER for the default

        Minimum Password Age [0]: 2
        Maximum Password Age [99999]: 30
        Last Password Change (YYYY-MM-DD) [2022-02-26]:
        Password Expiration Warning [7]:
        Password Inactive [-1]:
        Account Expiration Date (YYYY-MM-DD) [-1]:
```

```
➜  ~ chage -l jre-gonz42
Last password change                                    : Feb 26, 2022
Password expires                                        : Mar 28, 2022
Password inactive                                       : never
Account expires                                         : never
Minimum number of days between password change          : 2
Maximum number of days between password change          : 30
Number of days of warning before password expires       : 7
```

```
➜  ~ sudo chage root
Changing the aging information for root
Enter the new value, or press ENTER for the default

        Minimum Password Age [0]: 2
        Maximum Password Age [99999]: 30
        Last Password Change (YYYY-MM-DD) [2022-02-26]:
        Password Expiration Warning [7]:
        Password Inactive [-1]:
        Account Expiration Date (YYYY-MM-DD) [-1]:
```

```
➜  ~ sudo chage -l root
Last password change                                    : Feb 26, 2022
Password expires                                        : Mar 28, 2022
Password inactive                                       : never
Account expires                                         : never
Minimum number of days between password change          : 2
Maximum number of days between password change          : 30
Number of days of warning before password expires       : 7
```

- Cambia las passwords de USER y root para forzar que sigan las nuevas reglas:

```
passwd USER
sudo passwd root
```

Configuración de grupos del usuario:

En ocasiones, puede que nos interese dar algunos privilegios/permisos a ciertos usuarios. Por ejemplo, puede que queramos que los usuarios "administradores" tengan la habilidad de realizar tareas de mantenimiento, o puede que queramos que los usuarios "usuarios" puede que puedan hacer/tener cosas que los "administradores" no puedan.

En nuestro caso, nos piden definir dos grupos: sudo y user42. El primero es el que ya hemos configurado para permitir a los usuarios que pertenezcan a este grupo ejecutar comandos como root. El segundo nos permitirá definir que USER es un usuario de 42.

Comandos útiles:

Elemento	Descripción
`cut -d: -f1 /etc/passwd`	Ver todos los usuarios
`sudo adduser` USER	Crea un nuevo usuario con el nombre USER
`sudo usermod -l` USER_NEW USER_OLD	Renombra el usuario USER_OLD a USER_NEW.
`sudo userdel` USER	Elimina el usuario dado. Usa `-r` para eliminar también su directorio en /home.
`getent group`	Ver todos los grupos.
`groups`	Ver todos los usuarios en los que está el usuario que estamos usando (usa `groups USER` para hacer lo mismo con USER).
`getent group` GROUP	Verica qué usuarios están en el grupo GROUP.
`sudo groupadd` GROUP	Crea el grupo GROUP.
`sudo groupdel` GROUP	Borra el grupo GROUP.
`sudo usermod -aG` GROUP USER	Añade el usuario USER al grupo GROUP.

Configuración:

- Crea el grupo `user42`

```
sudo groupadd user42
```

  - Verifica que se ha creado con:

```
getent group
```

- Añade al usuario a los grupos requeridos:

```
sudo usermod -aG user42 USER
```

  - Verifica que el usuario está en los grupos `sudo` y `user42` con:

```
getent group
```

  - **Nota**: Recuerda que esta guía ya ha añadido el usuario al grupo sudo. Si no fuese así, añádele ahora.

Configuración de sudoers:

- Edita el archivo `/etc/sudoers`

```
sudo visudo
```

Modifica el archivo para tener:

```
  Defaults  env_reset
  Defaults  mail_badpass
  Defaults  badpass_message="Ups! Password is wrong. Let's try again."
  Defaults
secure_path="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/
bin:/snap/bin"
  Defaults  passwd_tries=3
  Defaults  logfile="/var/log/sudo/sudo.log"
  Defaults  log_input,log_output
  Defaults  requiretty
```

Elemento	Descripción
`env_reset`	Reinicia la variable de entorno (para sólo mostrar los comandos adecuados a los usuarios adecuados)
`mail_badpass`	Manda un mensaje si falla la autenticación.
`badpass_message="`MESSAGE`"`	Define el mensaje que imprime cuando falla la contraseña.
`secure_path="`PATHS`"`	Define el valor de la variable PATH.
`passwd_tries=`N	Números de intentos para iniciar sesión.
`logfile="`PATH`"`	Dirección donde guardar los registros de los usuarios que usan estos comandos.

Elemento	Descripción
`log_input, log_output`	Registros que guardar.
`requiretty`	Se pone para evitar un fallo de seguridad donde puedes iniciar sesión como root directamente.

- Ejecuta este comando para asegurarnos de que existe el directorio `/var/log/sudo`:

```
sudo mkdir -p /var/log/sudo
```

Configuración de Crontab:

Estos pasos nos permitirán ejecutar comandos en una fecha y/o hora determinada.

- Instalación:

```
sudo apt update -y
sudo apt install net-tools -y
```

- Introduce el script que quieras ejecutar de manera periódica (monitoring.sh) en el directorio `/usr/local/bin/`.

```
sudo vi /usr/local/bin/monitoring.sh
```

  - Verifica que se ha añadido de manera correcta:

```
sudo ls -l /usr/local/bin/monitoring.sh
```

    Debería aparecer:

```
-rw-r--r-- 1 root root 3582 Feb 27 05:53
/usr/local/bin/monitoring.sh
```

- Modifica de nuevo `sudoers` para permitir que el archivo se ejecute como súper-usuario sin password.

  - Abre el archivo:

```
sudo visudo
```

    - Añade la siguiente línea (un buen sitio para hacerlo es debajo de `%sudo ALL=(ALL:ALL) ALL`)

      ```
      %sudo ALL=(ALL) NOPASSWD: /usr/local/bin/monitoring.sh
      ```

- Reinicia:

  ```
  sudo reboot
  ```

- Verifica que funciona:

  ```
  sudo bash /usr/local/bin/monitoring.sh
  ```

- Abre crontab

  ```
  sudo crontab -u root -e
  ```

    - Este nos preguntará qué editor usar. Selecciona el que quieras.
    - Añade la siguiente línea al final del archivo para ejecutarlo cada 10min:

      ```
      */10 * * * * bash /usr/local/bin/monitoring.sh
      ```

Defensa:

Obtener la firma de LVM:

- Dirígete a la localización donde esté instalada la máquina virtual.
- Encuentra el archivo `.dvi`:

```
find . -name "*.dvi"
```

- Ve al directorio donde está el archivo.
- Ejecuta este comando:

Sistema operativo	Comando
Linux	`sha1sum *.vdi`
MacOS	`shasum *.vdi`

Nombre de la máquina:

Comando	Descripción
`hostnamectl`	Ver nombre actual.
`sudo hostnamectl set-hostname` HOSTNAME	Cambia el nombre de la máquina. Recuerda cambiarlo también en el archivo `/etc/hosts`. Necesita **reiniciar** para aplicarse.

Preguntas de teoría:

- Baigalaa's blog

Qué verificar:

Comando	Descripción
`lsblk`	Ver particiones.
`sudo aa-status`	Ver estado AppArmor.
`getent group` sudo	Ver usuarios en el grupo sudo.
`getent group` user42	Ver usuarios en el grupo user42.
`sudo service ssh status`	Sí, ver estado de SSH.
`sudo ufw status`	Estado de UFW.
`ssh` USER@IP -p 4242```	Conectar desde el ordenador a la máquina virtual por el puerto 4242.

Comando	Descripción
`sudo visudo`	Abrir el archivo de configuración de sudoers.
`vi /etc/login.defs`	Política de contraseñas.
`vi /etc/pam.d/common-password`	Política de contraseñas.
`sudo crontab -l`	Ver horario de cron.

Archivos de log:

Los archivos de registro/log se guardan en el directorio `/var/log/sudo`.

Ejecutar monitoring.sh cada 30s:

- Ejecuta:

```
sudo crontab -u root -e
```

- Modifica el archivo para que aparezca estas líneas:

```
*/1 * * * * /usr/local/bin/monitoring.sh
*/1 * * * * sleep 30s && /path/to/monitoring.sh
```

en vez de

```
*/10 * * * * /usr/local/bin/monitoring.sh
```

- ¿Cómo funciona? Ejecuta dos veces cada minuto el script. Sin embargo, el segundo se retrasa 30s para que entre ambos se ejecuten cada 30s.

Crear un nuevo usuario:

- Crea el nuevo usuario USER:

```
sudo adduser USER
```

- Verifica que la información de la expiración de la contraseña de este:

```
sudo chage -l USER
```

- Añade el usuario a los grupos sudo y user42:

```
sudo usermod -aG sudo USER
sudo usermod -aG user42 USER
```

Notas:

- Cuando el comando *su -* es mostrado, la intención es que se ejecute siendo root. Por tanto, todas las secciones que no usen ese comando están pensadas para ser ejecutadas no siendo root (USER).
- Antes the avanzar a la siguiente sección, verifica que lo hecho hasta ahora ha funcionado correctamente. El orden elegido con un motivo específico.
- Cuando sea necesario editar un archivo, el comando utilizado será **Vim**. Siéntete libre de usar el que prefieras.
- Si en algún momento encuentra alguna de estas palabras, estas dependerán de su máquina y deberán ser cambiadas por la que corresponda.
    - DATE
    - MACHINE_NAME
    - ID
    - PORT
    - USER
    - N
    - GROUP
    - MESSAGE
    - FILE