

Indice

[**Estudio de Roles del Servidor:** 3](#_Toc177228680)

[**Relevamiento y Justificación:** 3](#_Toc177228681)

[**Instalación de Máquina Virtual:** 4](#_Toc177228682)

[**Instalación de Sistema Operativo (Rocky Linux):** 8](#_Toc177228683)

[**Instalación de Servidor mySQL Local:** 16](#_Toc177228684)

[**Prototipo de Script:** 17](#_Toc177228685)

# **Estudio de Roles del Servidor:**

**Usuario cliente**: El recibirá un reporte impreso mensual.

Asimismo, puede acceder a su evolución y al plan que el entrenador le asignó. Ingresará su CI o identificación para registrar el cumplimiento de lo agendado.

**Usuario administrativo**: El usuario administrativo se encargará de llevar la agenda de los clientes.

También estará a cargo del cobro de cuotas debiendo llevar control de los pagos por cuponera, mensuales y anuales.

**Usuario avanzado**: Ingresará usuarios básicos, ejercicios y deportes a nivel nacional.

**Entrenador**: Ingresará programas (planes de ejercicios) para cada usuario, armará combos, registrará evolución y calificación. Podrá consultar la evolución de cualquier deportista, cliente de fisioterapia o cliente de Pilates, de su club correspondiente.

**Usuario Seleccionador**: Podrá consultar la evolución de cualquier deportista, cliente de fisioterapia o cliente de Pilates, de cualquier club y podrá armar equipos con ellos, indicando el deporte en el cual actuarán.

Podrá ingresar clubes y entrenadores.

**Usuario Administrador TI**: Es el súper usuario de la aplicación. (todo lo anterior y más)

# **Relevamiento y Justificación:**

Las ventajas que tiene Rocky Linux con respecto al relevamiento son las siguientes:

Estabilidad y Compatibilidad Empresarial: comparable a Red Hat Enterprise Linux (RHEL), lo que lo hace adecuado para entornos empresariales y servidores.

Comunidad activa y colaborativa que brinda soporte y comparte conocimientos: lo que brinda soporte y comparte conocimientos.

Actualizaciones y parches de seguridad consistentes: Lo que es fundamental para mantener la estabilidad y seguridad del sistema

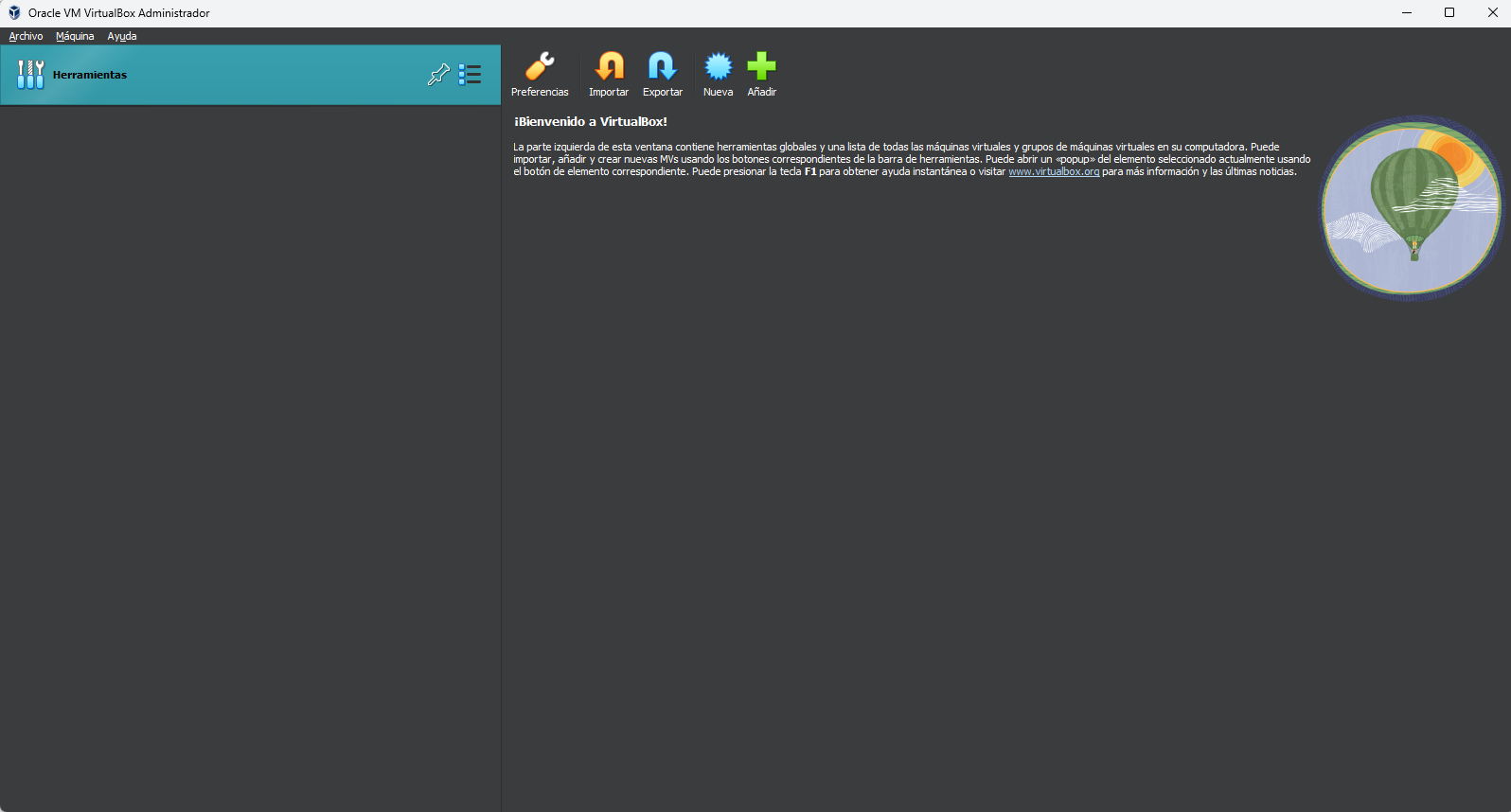
Licencia de código abierto: Al ser una distribución de código abierto, Rocky Linux permite la libertad de acceso y personalización.

**Manual de Instalación de Rocky Linux:**

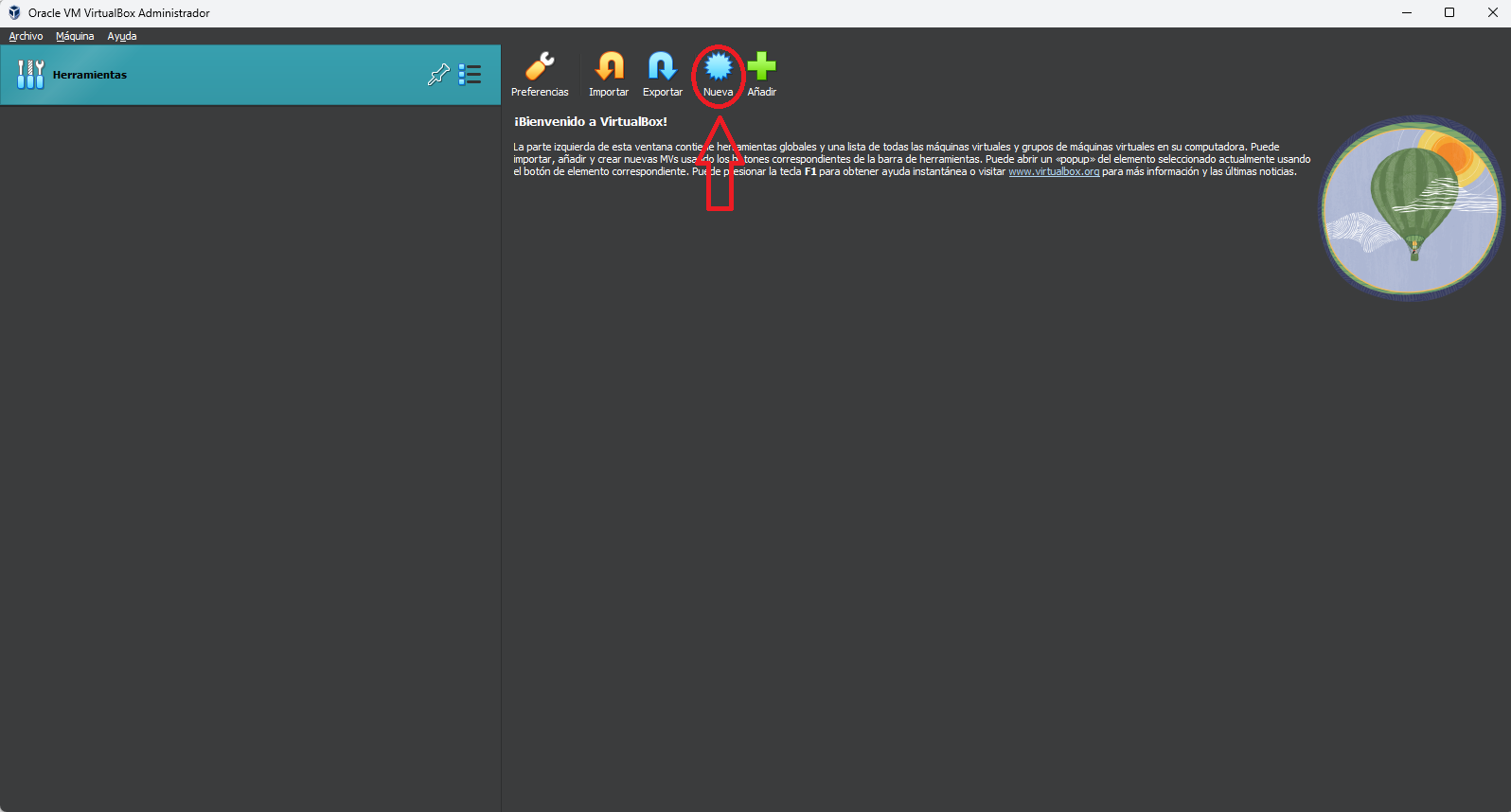
# **Instalación de Máquina Virtual:**

Programa que usaremos para la Instalación: Oracle VM Virtual Box

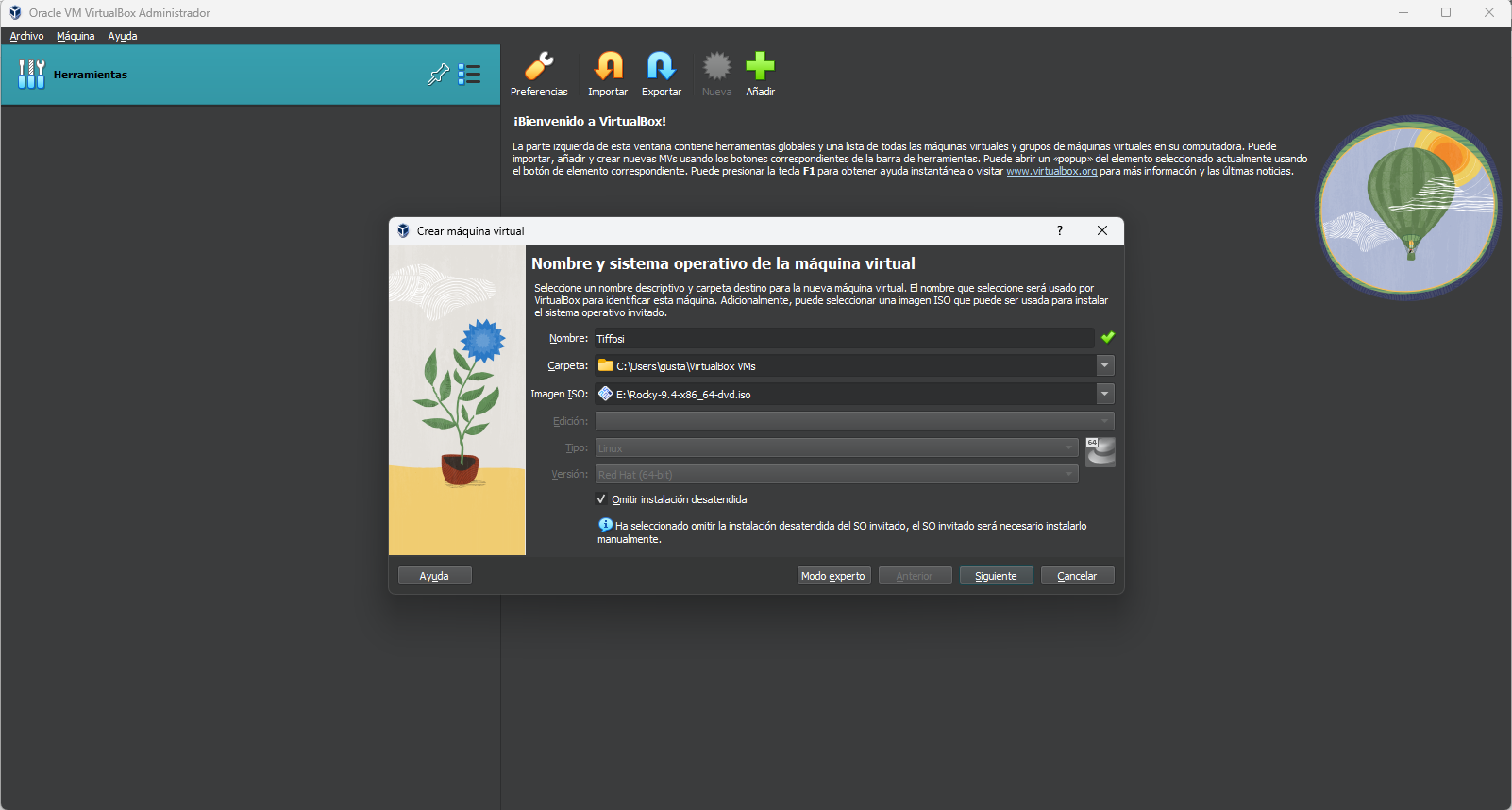
Primer Paso: Abrir Oracle VM Virtual Box.

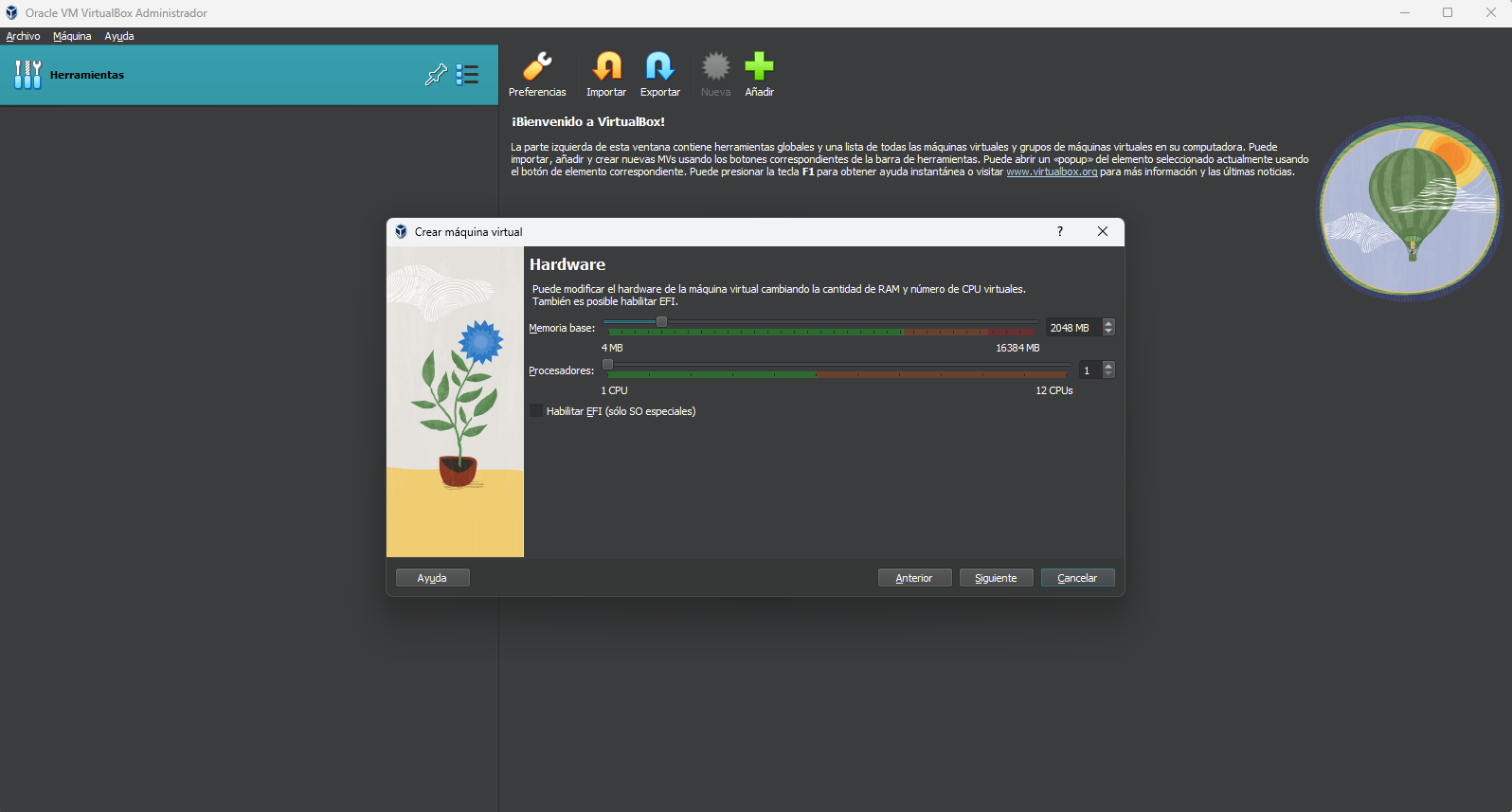


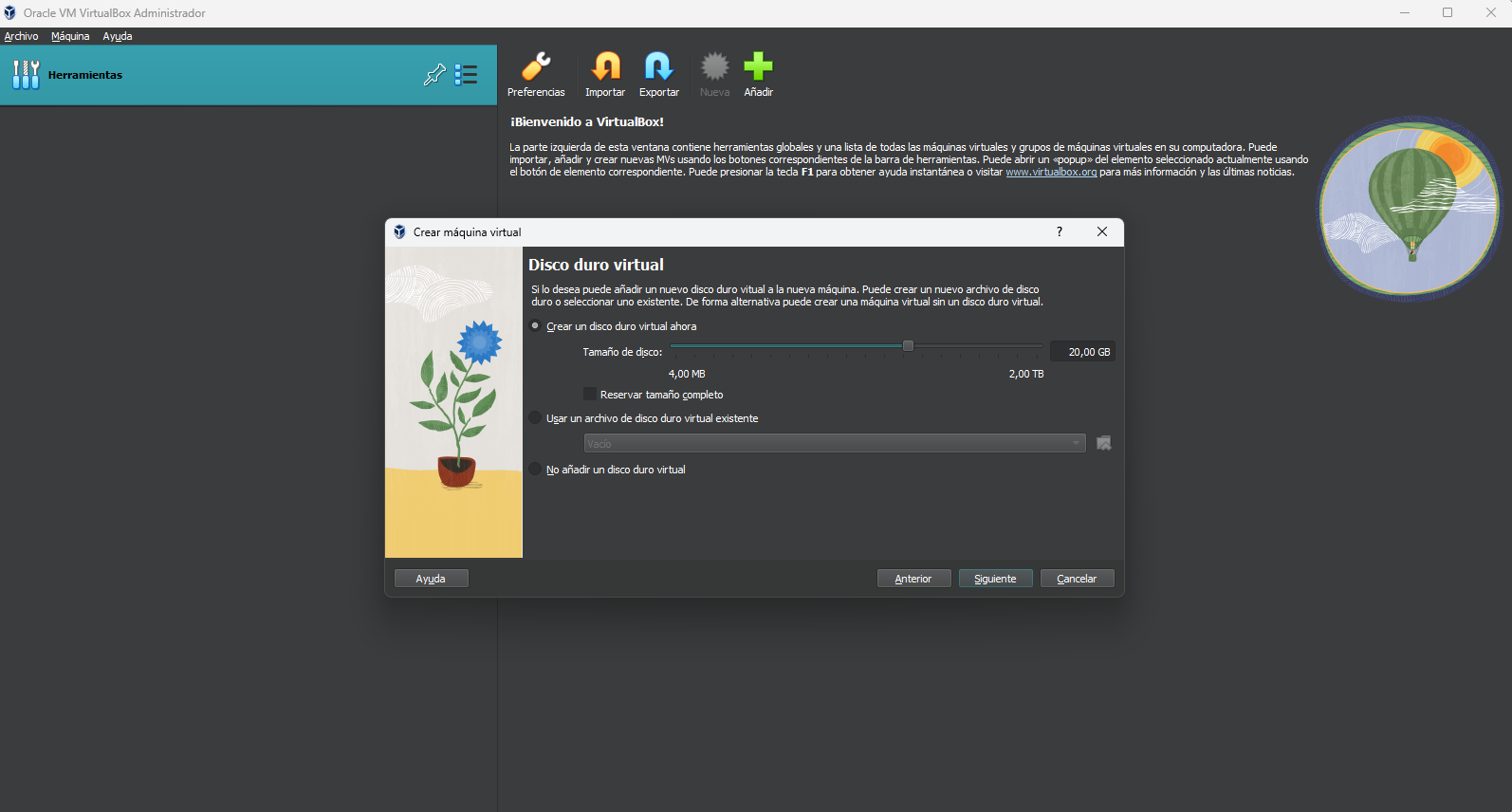
Segundo Paso: Damos en la opción de Nueva.



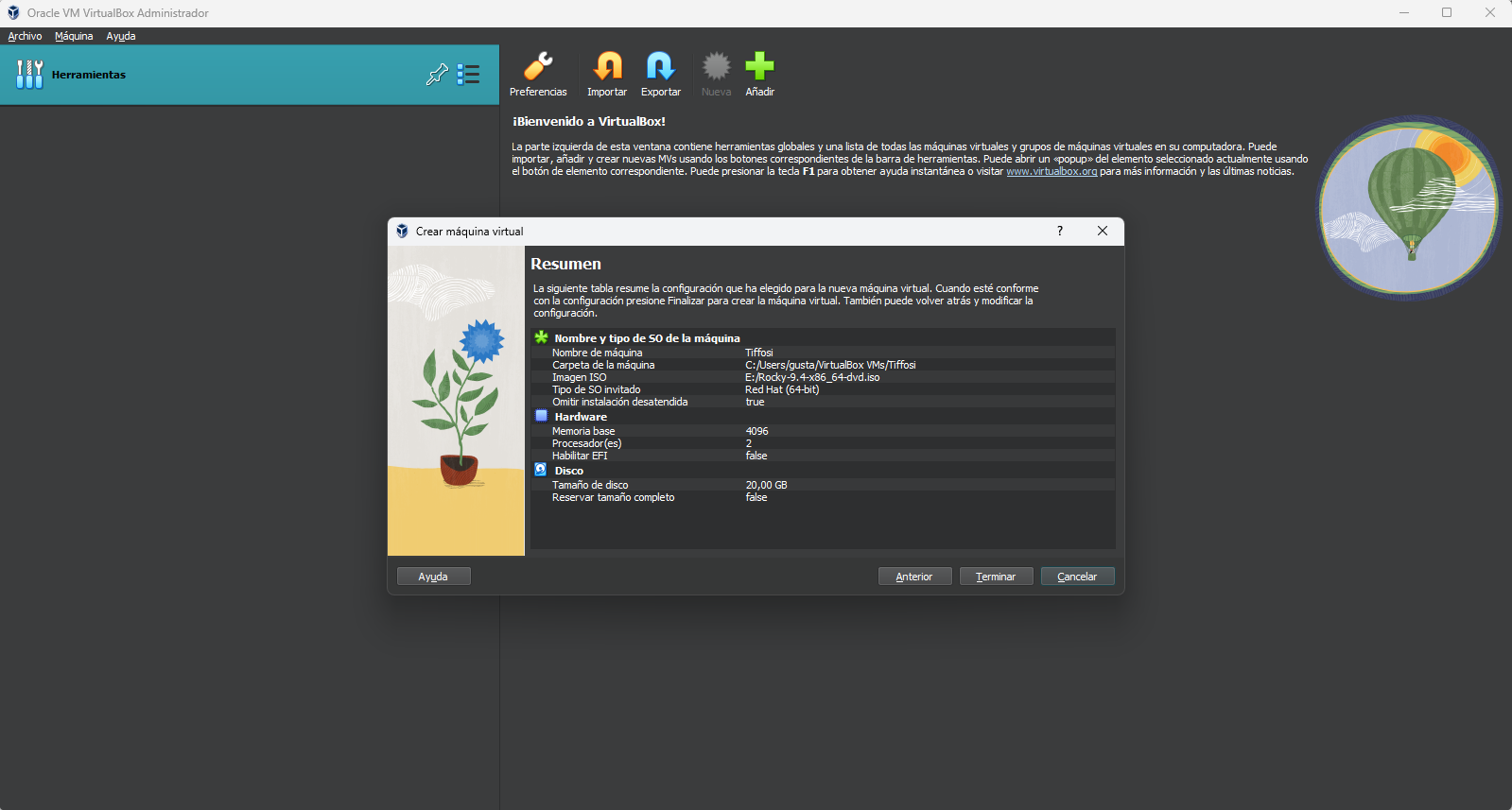
Tercer Paso: Al darle al botón de nuevo se despliega un nuevo menú para poder colocar un nombre a la máquina virtual:

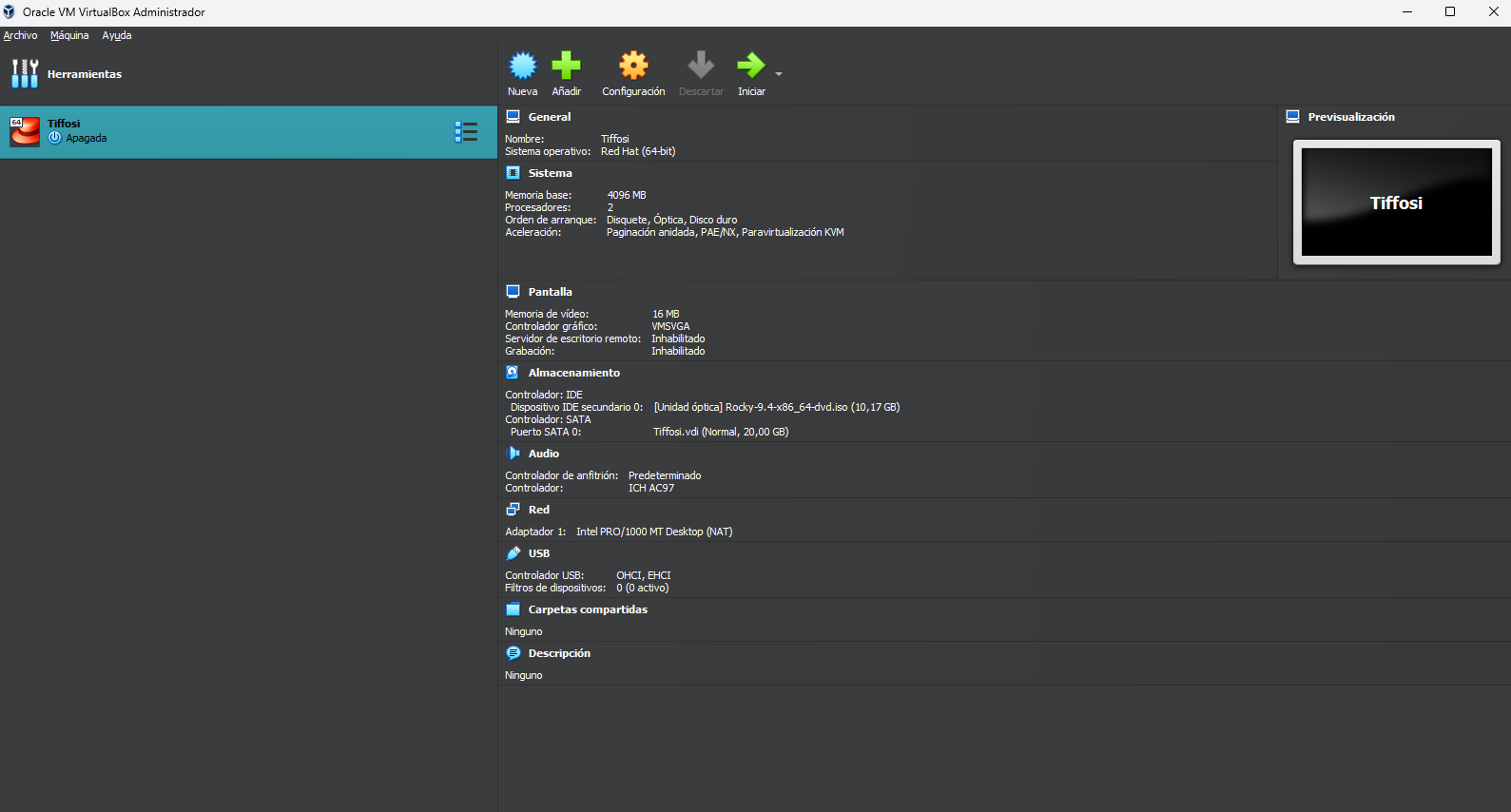


Cuarto Paso: Después de darle al botón de siguiente del menú anterior, nos aparecerá el siguiente menú, en el cual asignaremos la memoria RAM que queremos que tenga la máquina y la cantidad de núcleos del procesador de la misma. La **recomendación** es 2gb y 1 núcleo para que funcione correctamente.

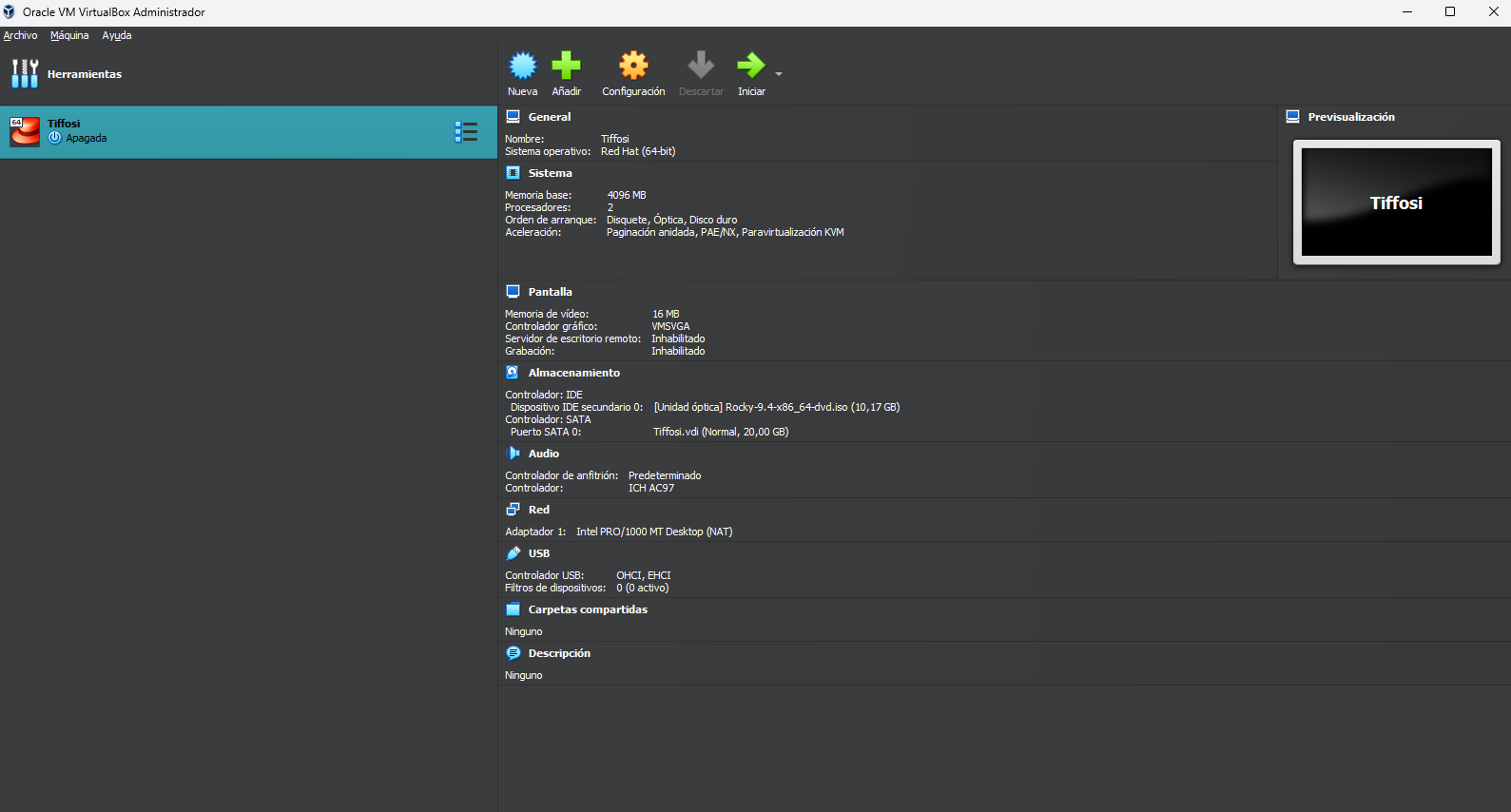
Quinto Paso: Después del menú anterior, nos aparecerá un menú para asignarle el almacenamiento (tamaño de disco) virtual de la máquina. La recomendación sería de 20gb mínimo.

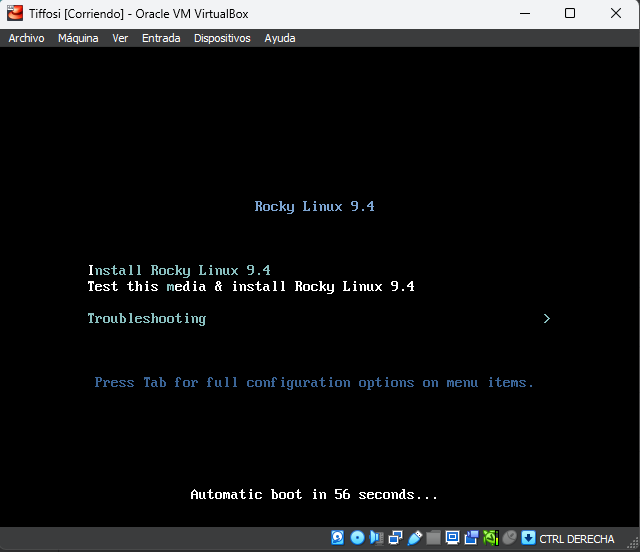
Sexto Paso: Tendremos el menú de configuración de la máquina virtual, el cual te permite verificar los datos puestos anteriormente, en caso de que algún dato esté erróneo, nos permite ir hacia los pasos anteriores con el botón “Anterior” y en caso de que todo esté correcto, le podemos dar al botón “Terminar”.

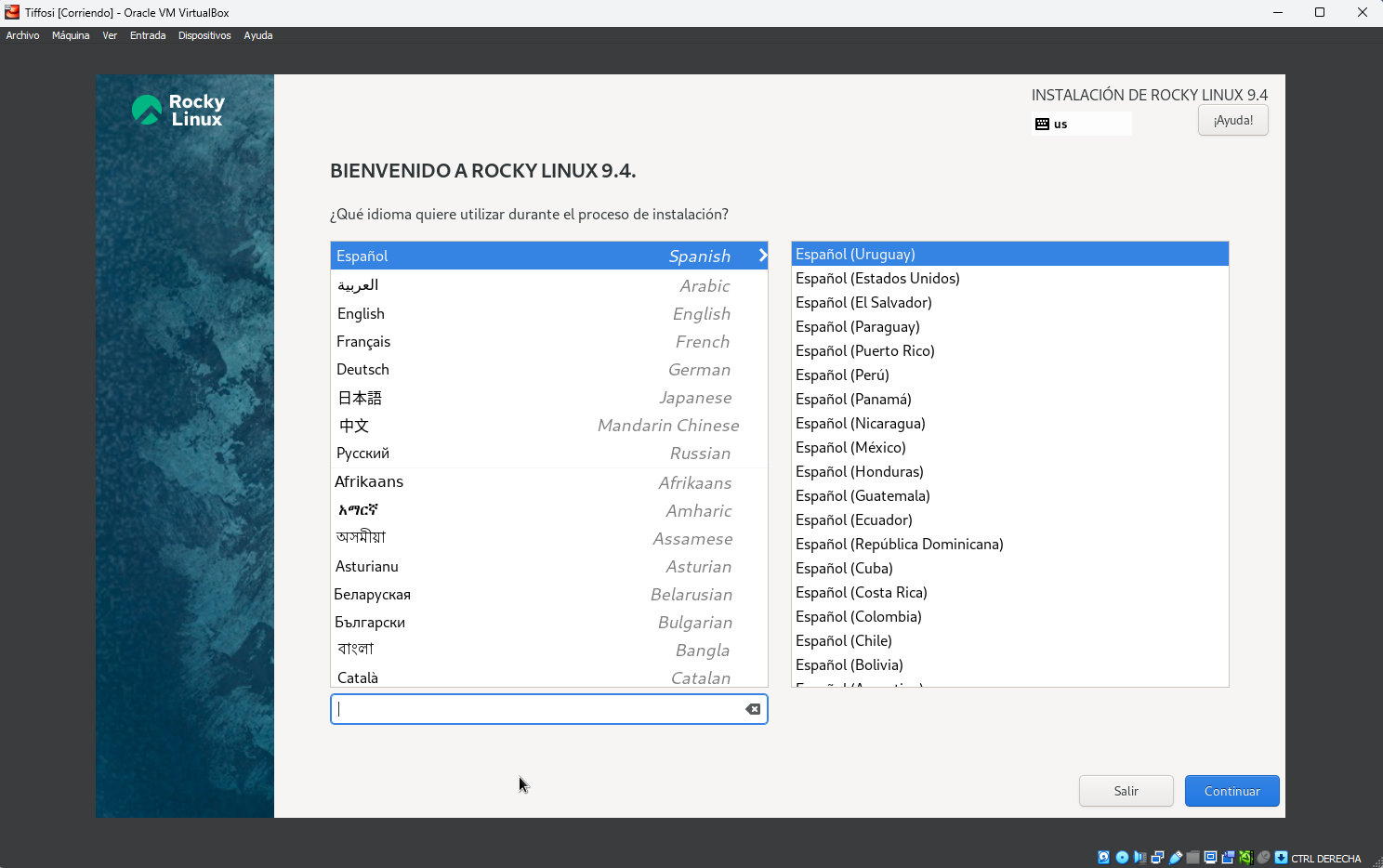


Séptimo y último paso: luego de darle al botón “Terminar”, podemos verificar la creación de la máquina virtual en la parte izquierda del programa con su correspondiente configuración.

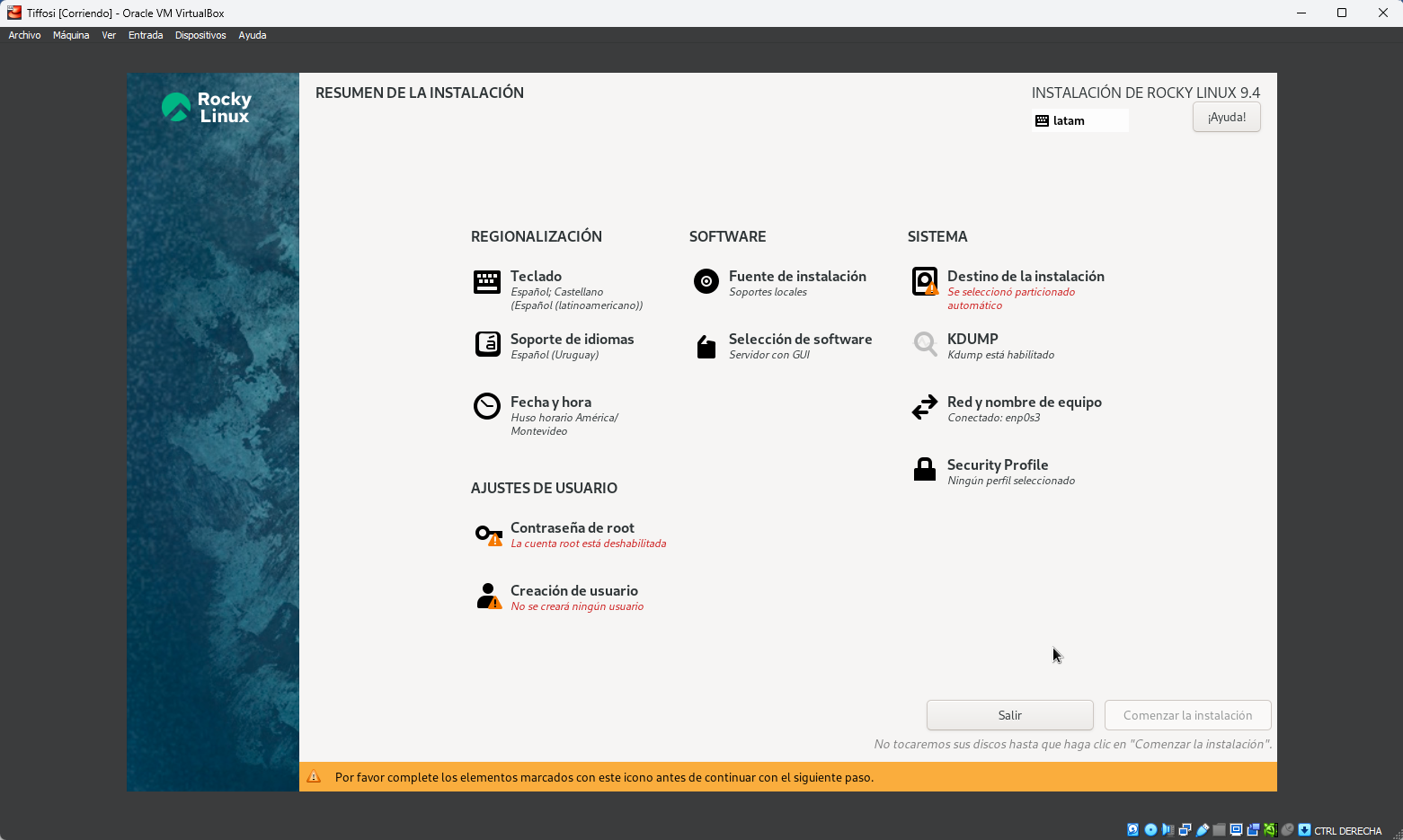
# **Instalación de Sistema Operativo (Rocky Linux):**

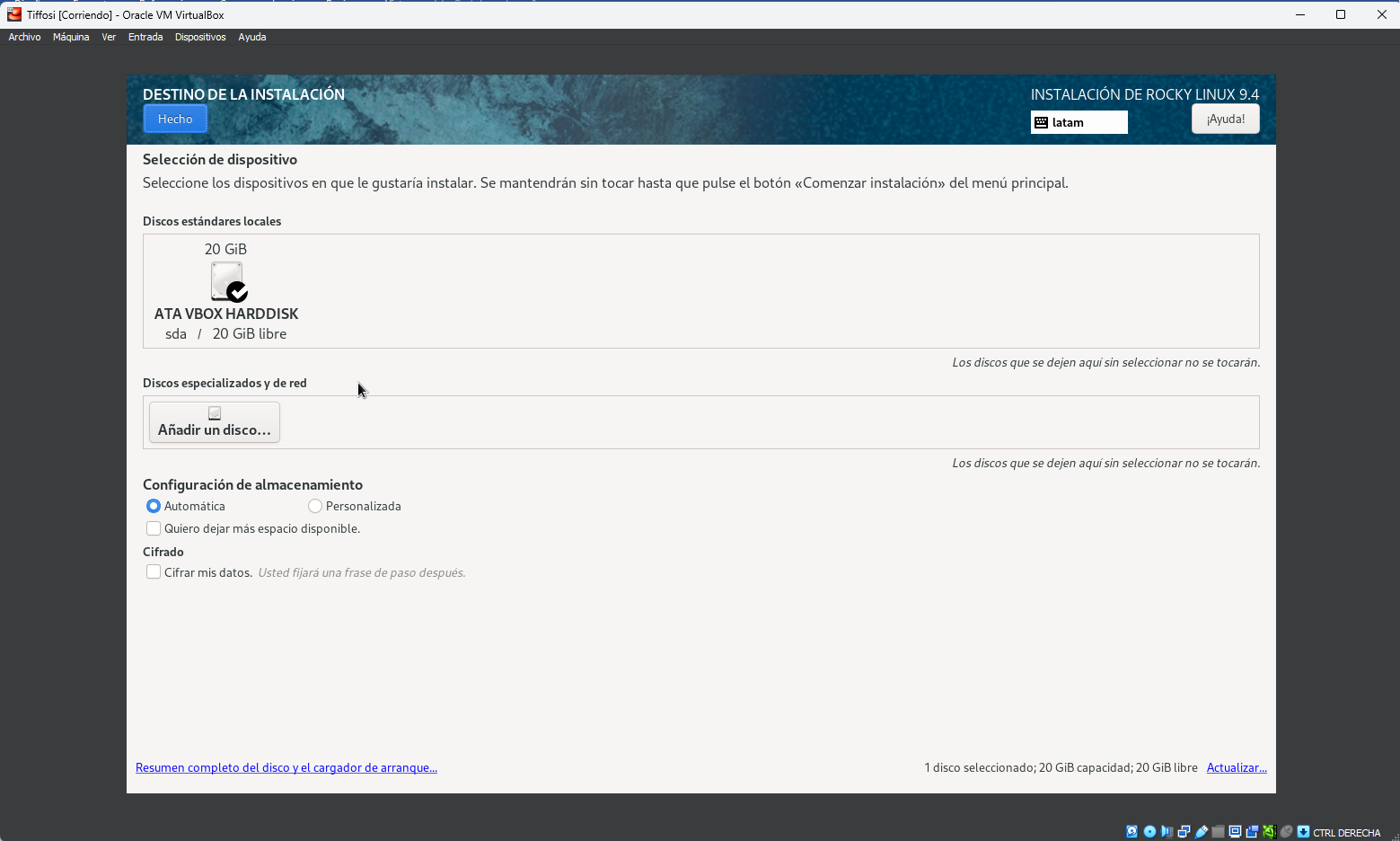
Primer Paso: Iniciar la máquina virtual.

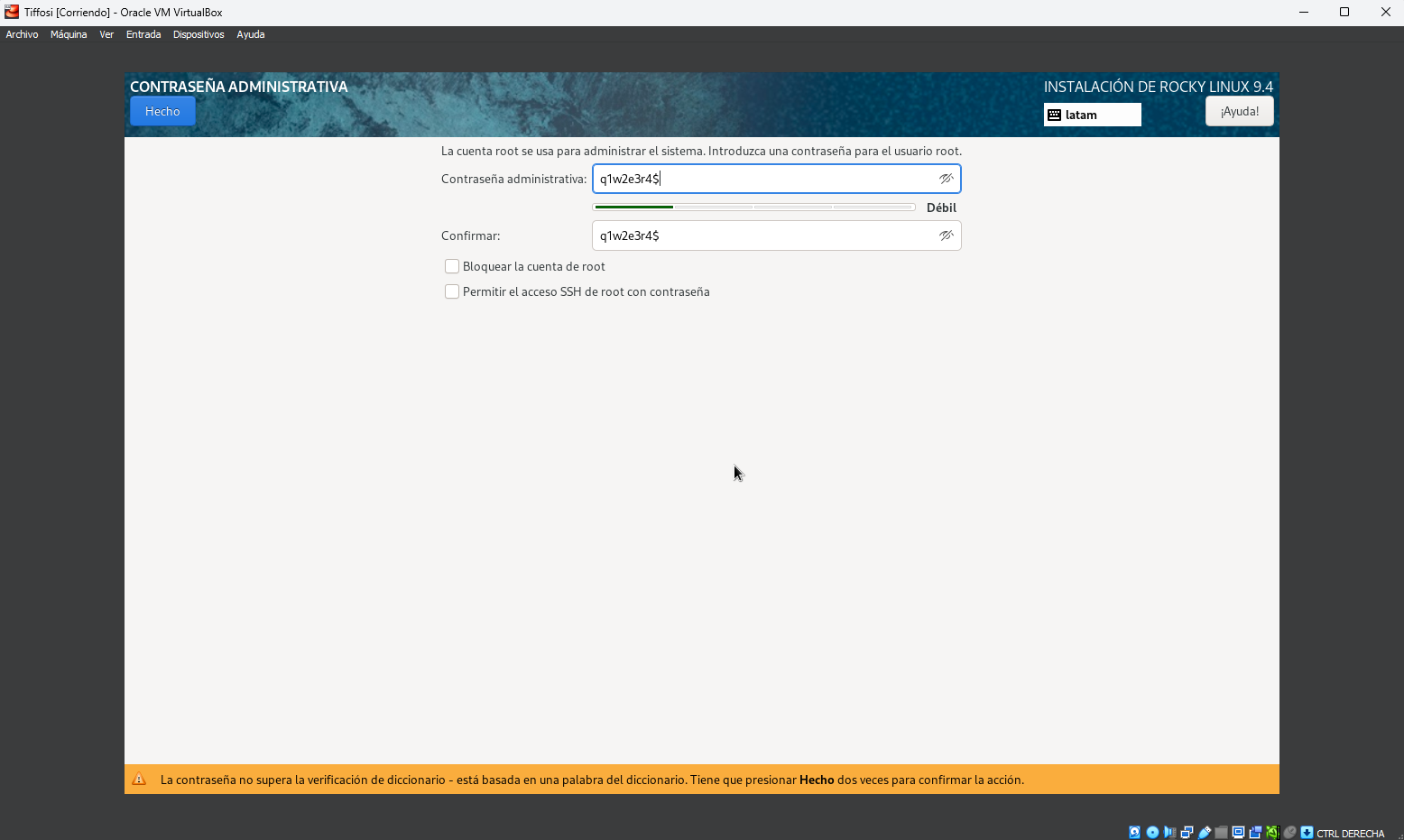
Segundo Paso: Darle a la opción “Install Rocky Linux 9.4” con el botón Enter.

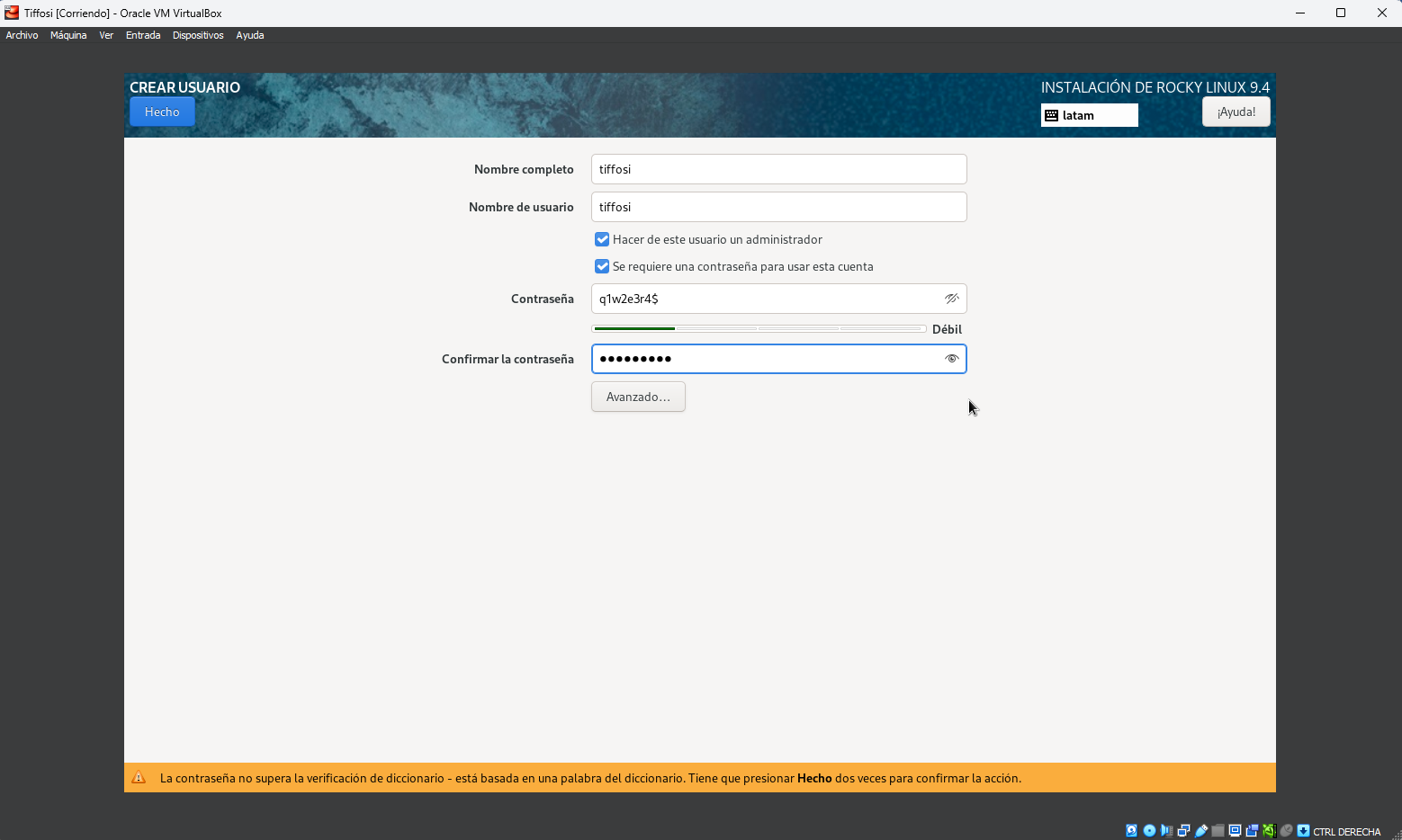
Tercer Paso: Al darle a instalar nos saldrá el menú de selección de idioma para el sistema operativo, el cual seleccionaremos y daremos al botón de “Continuar” para poder acceder a lo siguiente.

Cuarto Paso: Una vez seleccionado el idioma, nos aparecerá un menú con diversas opciones, los que nos interesan principalmente son: Selección de Software, Destino de la instalación, Contraseña de root y Creación de Usuario.



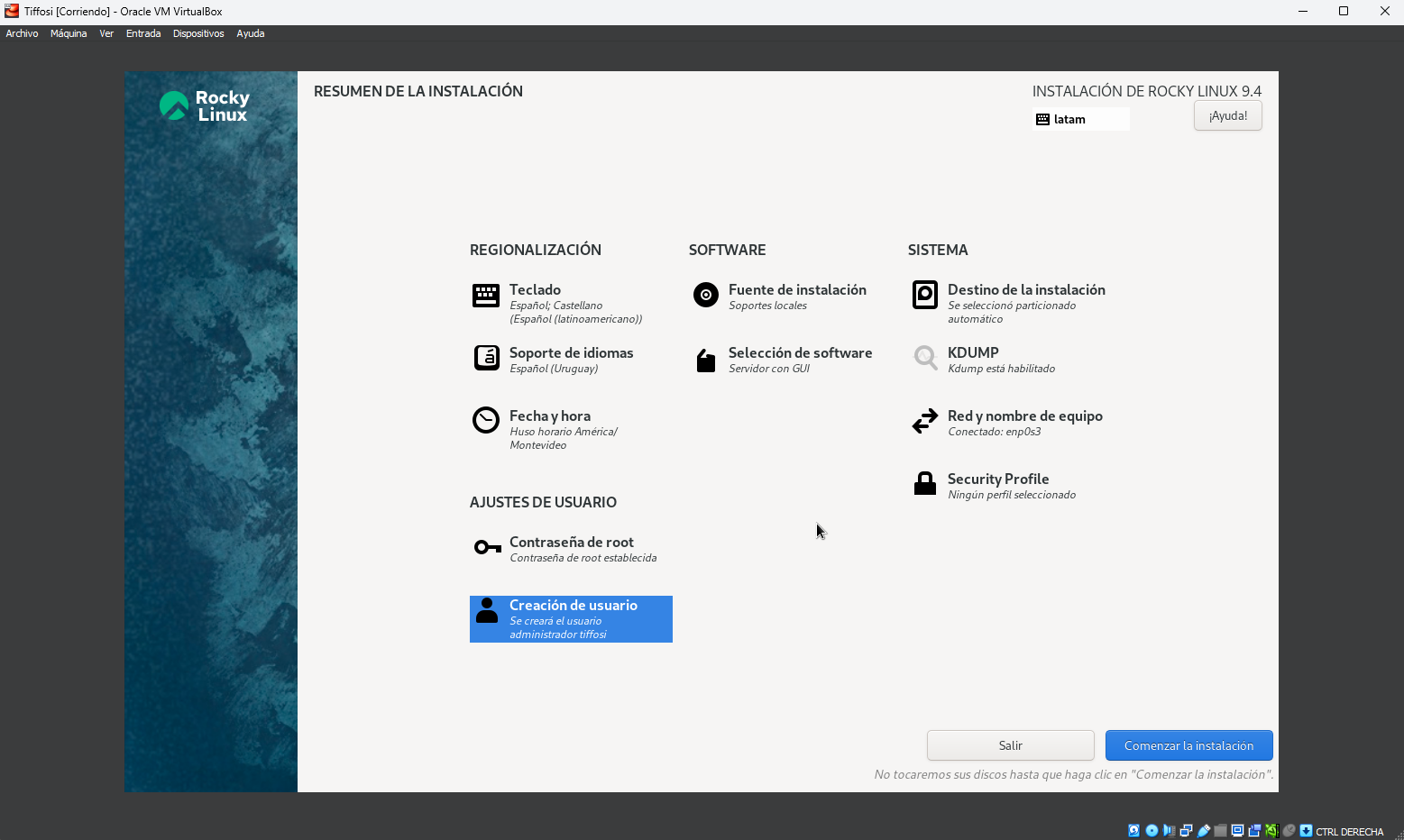
Quinto Paso: Daremos clic encima en Destino de la Instalación y se nos abrirá el siguiente menú, del cual tendremos que apretar, en la parte inferior de “Discos estándares locales”, en donde nos aparecerán los discos detectados por el sistema, seleccionamos uno y le damos a el botón “Hecho” en la parte superior izquierda de la pantalla.

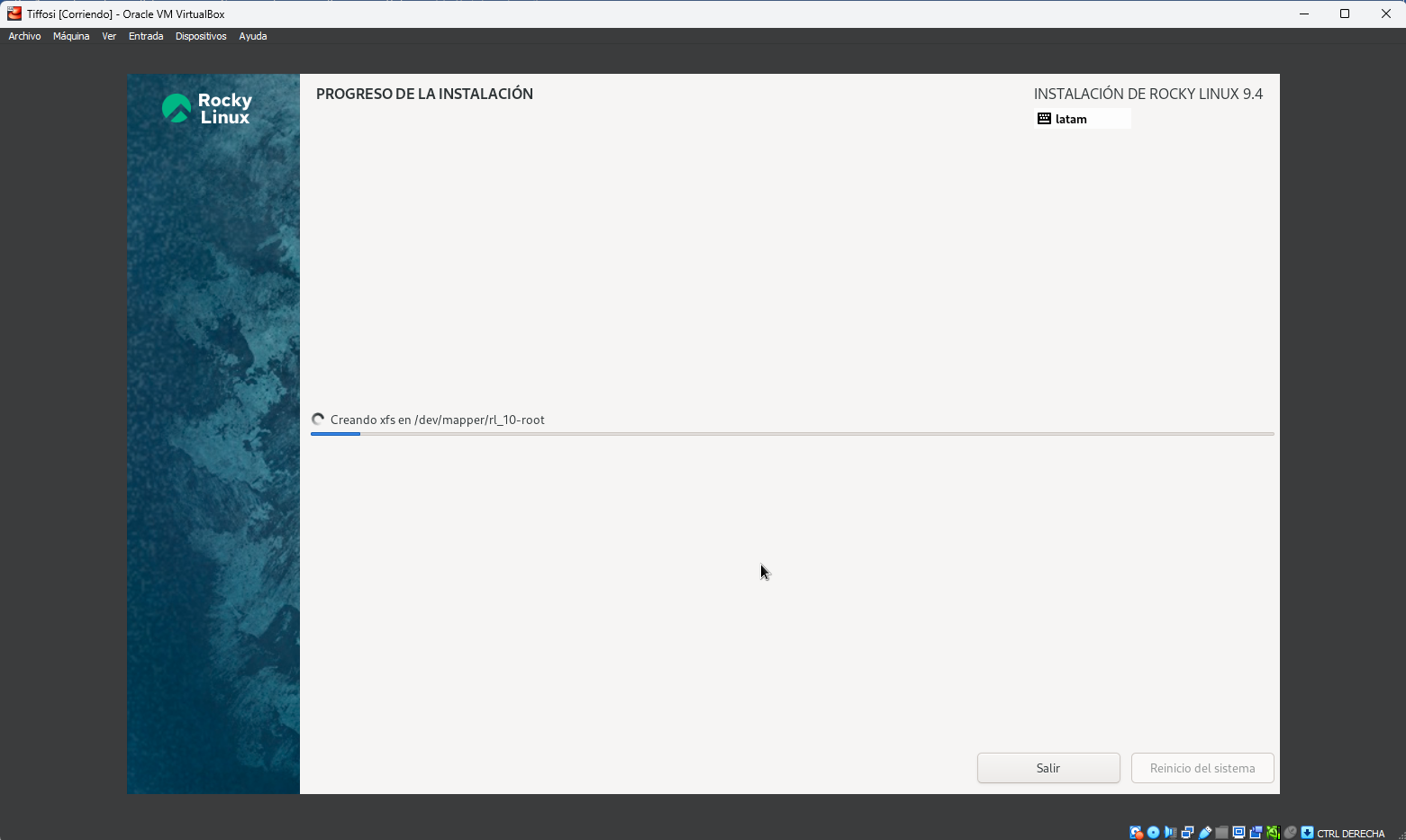
Sexto Paso: Luego de haber salido del menú anterior, entraremos en la parte de ajustes de usuario, precisamente en la opción de contraseña de root, con el cual nos dará acceso al usuario root, que tiene permiso sobre la mayoría de cosas dentro del sistema operativo, le damos al botón de hecho y nos mandará al menú con las otras opciones.

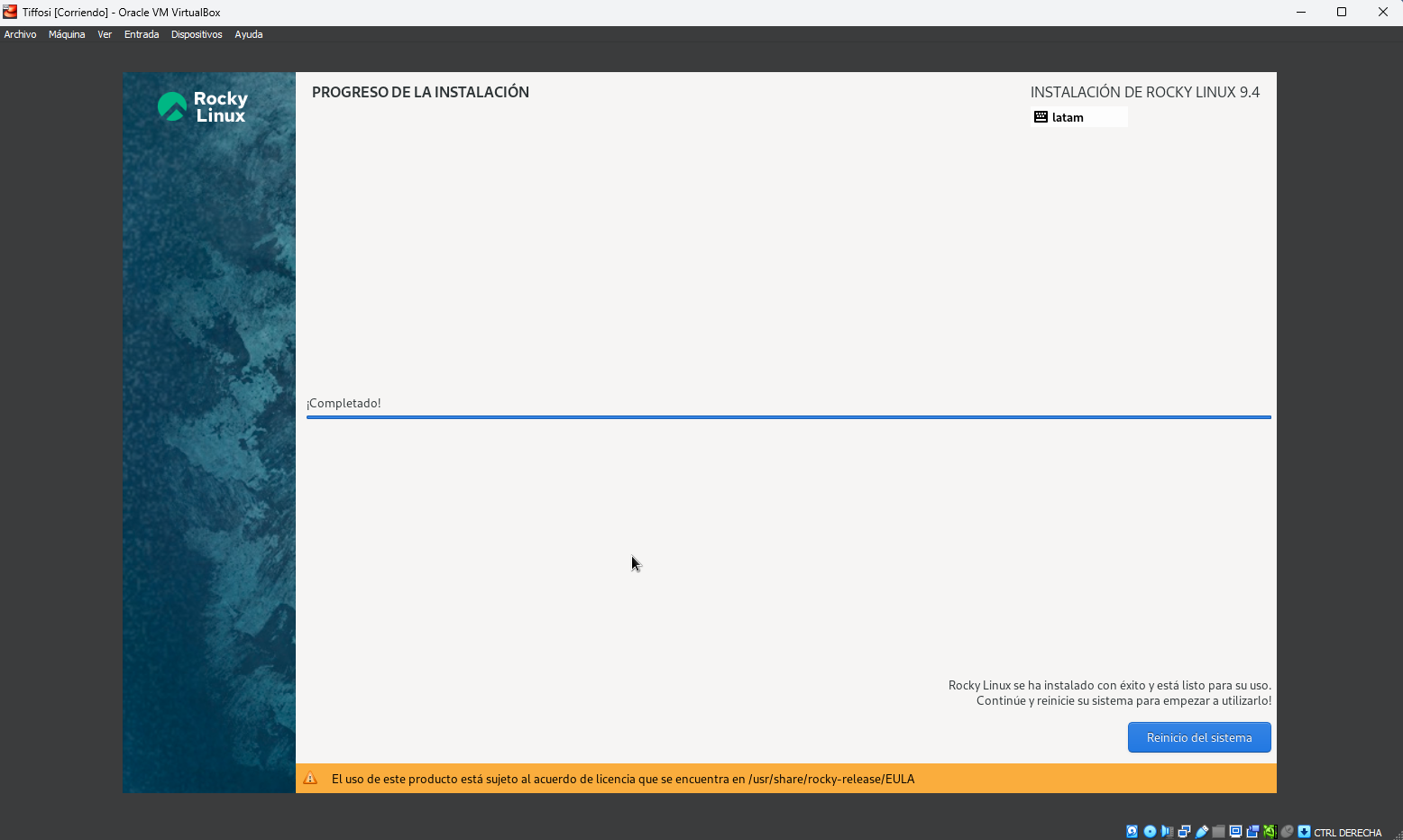
Septimo Paso: Luego de haber salido del menú anterior, entraremos en la parte de ajustes de usuario, precisamente en la opción de Creación de Usuario, en el cual crearemos un nuevo usuario dentro del sistema con una contraseña, y activamos la opción de “Hacer de este usuario un administrador”, para que el mismo tenga acceso a casi todo el sistema al igual que el usuario root, le damos al botón de hecho y nos mandará al menú con las otras opciones.

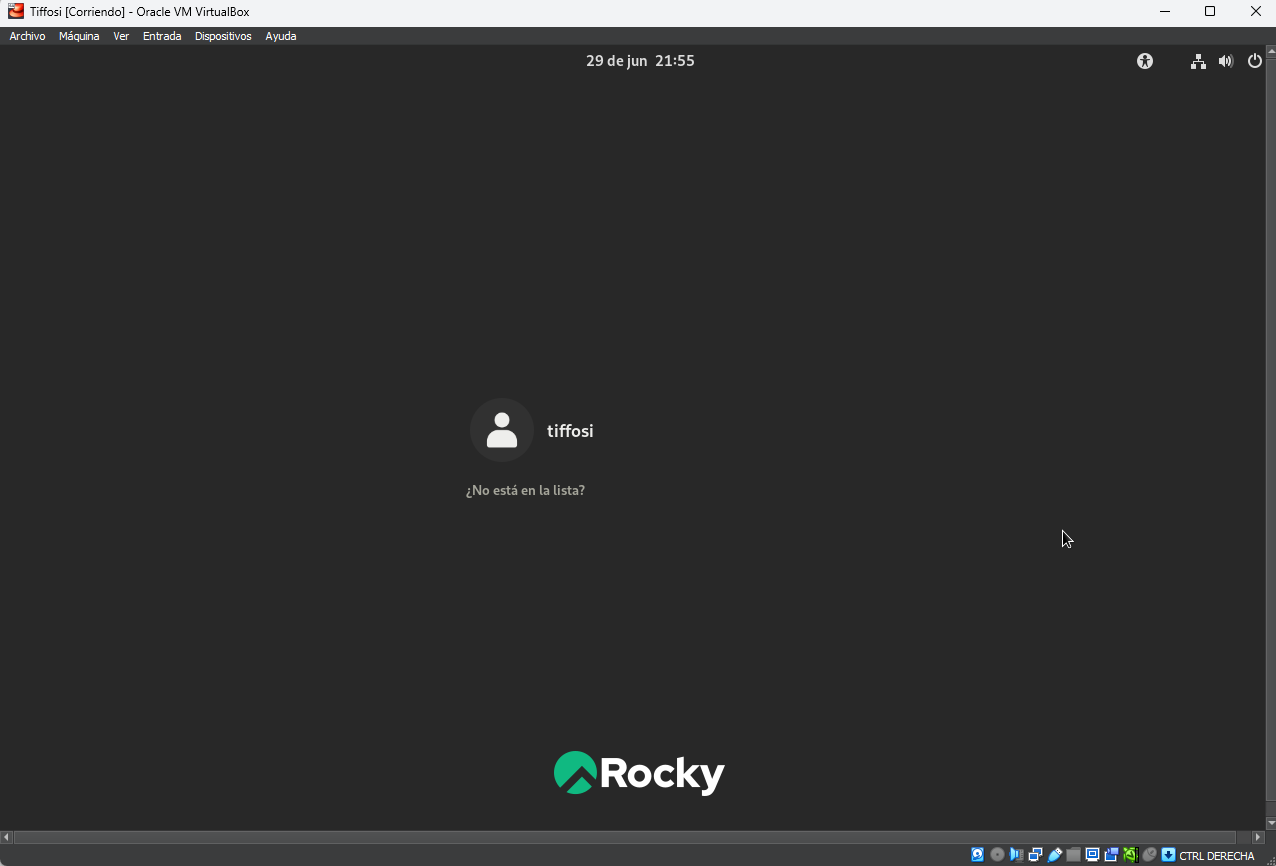
Octavo Paso: Una vez configurado todo lo anterior, nos fijaremos que la opción de “Selección de

Software” esté en GUI, para que nos permita tener una interfaz gráfica en el sistema operativo.

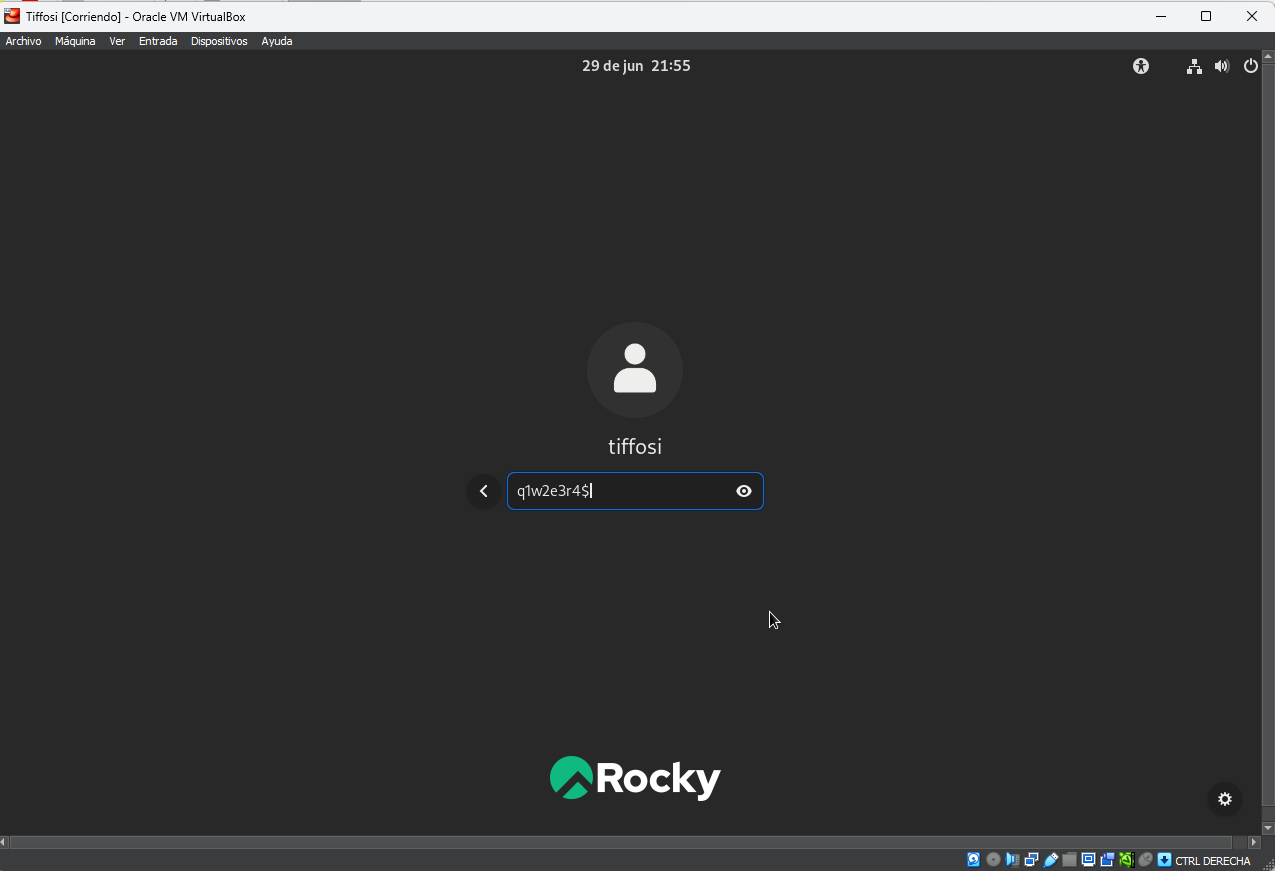


Noveno Paso: Luego de haber verificado, le damos al botón de “Comenzar la Instalación” y este mismo iniciará la instalación como su nombre lo indica.

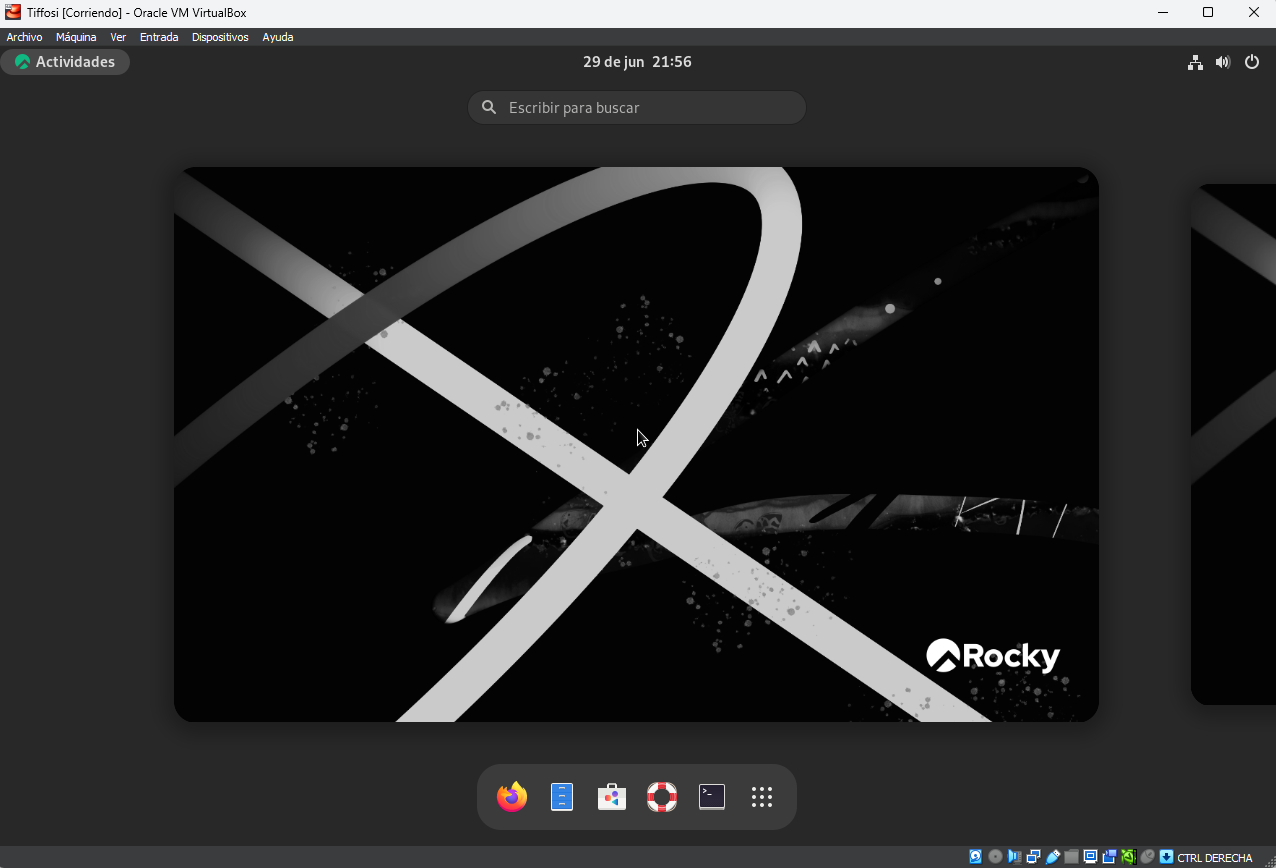
Décimo Paso: Una vez que finalice el proceso de instalación, se nos habilitará el botón de “Reinicio del Sistema”, el cual reinicia la máquina virtual y nos mostrará el sistema operativo ya instalado.

Onceavo Paso: Al iniciar la máquina virtual, nos aparecerá una lista con los usuarios que hallamos creado, en este caso nosotros hemos creado el usuario “tiffosi”, el cual tiene rol como administrador, seleccionamos ese usuario.

Doceavo Paso: Una vez seleccionado el usuario, ponemos la contraseña del mismo que configuramos antes.



Treceavo Paso: Una vez iniciada la sesión con el usuario nos muestra satisfactoriamente el sistema operativo “Rocky Linux”, instalado correctamente y funcional.



**Comandos Utilizados**(Rocky Linux)(Versión de Fedora)

sudo dnf install zenity

# **Instalación de Servidor mySQL Local:**

Para instalar y configurar MySQL en Linux, primero cambias a usuario root con su -, luego actualizas los paquetes con dnf update, instalas el servidor MySQL con dnf install mysql-server, inicias el servicio con systemctl start mysqld y lo habilitas al inicio con systemctl enable mysqld. Después, aseguras la instalación ejecutando mysql\_secure\_installation y verificas el estado del servicio con systemctl status mysqld. Para acceder a MySQL, usas mysql -u root -p e introduces la contraseña. También puedes verificar las conexiones de red con netstat -punta o filtrar por MySQL con netstat -punta | grep mysql, y si necesitas comprobar el puerto 3306, lo haces con nmap -p 3306 localhost.

Comandos:

su -

dnf update

dnf install mysql-server

systemctl start mysqld

systemctl enable mysqld

mysql\_secure\_installation

systemctl status mysqld

// PARA INGRESAR AL SISTEMA

mysql -u root -p

Q1w2e3r4$

netstat -punta

netstat -punta | grep mysql

di nmap y me ofreció instalarlo

nmap -p 3306 localhost

# **Prototipo de Script:**

#!/bin/bash

while true; do

opcion=$(zenity --list --title="Menú Principal" --text="Seleccione una opción:" \

--column="Número" --column="Opción" \

--height=400 --width=300 \

1 "Menú de Usuarios" \

2 "Menú de Grupos" \

3 "Menú de Red" \

4 "Menú de Servicios" \

5 "Menú de Firewall" \

6 "Hacer Respaldo Local" \

7 "Salir")

case $opcion in

1)

bash tiffosiMenuUsuarios.sh

;;

2)

bash tiffosiMenuUsuarios.sh

;;

3)

bash red.sh

;;

4)

bash servicios.sh

;;

5)

bash firewall.sh

;;

6)

Bash respaldo.sh

;;

7)

zenity --info --text="Saliendo del programa"

break

;;

\*)

zenity --error --text="Opción no válida"

;;

esac

done

**BASH tiffosiMenuUsuarios**

#!/bin/bash

# codigo principal (todo lo que no esta dentro de una function), primer parte

# Script Adm. Usuarios linux -

clear

echo "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"

echo "\* \*"

echo "\* Script para gestionar usuarios a traves de un GUI >> GESTUSERS << \*"

echo "\* \*"

echo "\* Fecha y hora de la ejecución del Script \*"

echo "\* date +%d-%m-%Y - date +%H:%M:%S \*"

echo "\* \*"

echo "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"

#sleep 1

#exit

# sigue y se encuentra con la invocacion de una funcion, que se llama de casualidad main !

# (no ejecuta funciones si no son invocadas)

# puede haber funciones antes y despues del codigo principal

###################################################################################

function seleccion(){

# Obtener la lista de usuarios del sistema que pueden iniciar sesión

lsusr=$(awk -F: '($3 >= 1000 && $3 < 60000) || $1 == "root" {print $1}' /etc/passwd | sort)

# Mostrar la lista de usuarios en un cuadro de diálogo Zenity

usr=$(zenity --list --height=300 --title="Elija un usuario" --column="USUARIOS" $lsusr)

cod=$?

case $1 in

1)

while true

do

# https://askubuntu.com/questions/50988/wheres-documented-combo-box-usage-with-zenity

# zenity --help-forms

# zenity --forms --title "Window title" --text "Combo name" --add-combo "Insert your choice." --combo-values "a|b|c|d|e"

alta\_usuario=$(zenity --forms \

--title 'ABM usuarios y grupos' \

--text ' \* Alta de usuario \* ' \

--add-entry "Nombre de usuario (ej. Juan Perez): " \

--add-entry "HOME dir (vacío=default) : " \

--add-entry "Comentario : " \

--add-combo 'Shell' \

--combo-values '/bin/bash|/bin/sh' \

--add-password "Contraseña : " \

--add-password "Repita contraseña : ")

cod=echo $?

if [ $cod != 0 ] #Pulso aceptar

then

# pulsó tecla Esc, botón Cancelar, o cerró la ventana con la cruz (X)

echo cancelar

return # vuelvo al menu principal

fi

# echo 'aceptar' # pulso aceptar

# debug

echo '$alta\_usuario : >'$alta\_usuario'<'

usuario=$(echo $alta\_usuario | cut -f1 -d'|')

home\_dir=$(echo $alta\_usuario | cut -f2 -d'|')

comentario=$(echo $alta\_usuario | cut -f3 -d'|')

interprete=$(echo $alta\_usuario | cut -f4 -d'|')

contr1=$(echo $alta\_usuario | cut -f5 -d'|')

contr2=$(echo $alta\_usuario | cut -f6 -d'|')

echo '$usuario = >'$usuario'<'; echo '$home\_dir = >'$home\_dir'<'; echo '$comentario = >'$comentario'<'; echo '$interprete = >'$interprete'<'; echo '$contr1 = >'$contr1'<'; echo '$contr2 = >'$contr2'<'

if test $(echo $usuario | wc -w) -ne 2 # pruebo si tengo algo del estilo Juan Perez

then

zenity --info --text 'Debe ser de formato primer-nombre primer-apellido (Ej.: Juan Perez).'

continue # vuelvo a la cabeza del loop a pedir los datos de nuevo

#echo largo distinto de 2

#else

# echo largo es 2 # largo correcto

fi

#echo 'seguir puliendo esto, si tiene una sola parte capaz tomarla como el usuario, ejemplo jperez.

# normalizo el nombre Juan Perez -> jperez

#https://linuxcenter.es/aprende/guias-tutoriales/item/136-como-pasar-todas-las-letras-a-minusculas-a-mayusculas-o-la-primera-letra-en-mayuscula-y-el-resto-de-letras-en-minusculas-en-bash

nom="$usuario"

echo $nom

n=$(echo $nom | cut -c1 | tr '[:upper:]' '[:lower:]')

echo $n

a=$(echo $nom | cut -f2 -d" " | tr '[:upper:]' '[:lower:]')

echo $a

nc=$n$a

echo $nc

usr=$nc

# https://www.delftstack.com/es/howto/linux/check-if-variable-is-empty-in-bash/

if [ "$home\_dir" == "" ]

then

# echo home dir vacio

home\_dir=/home/$usr # directorio de inicio por defecto en /home/ mas nombre de usuario normalizado (Juan Perez -> jperez)

zenity --info --text "Directorio de inicio vacio, se ajusta automaticamente a $home\_dir"

fi

if [ "$comentario" == "" ]

then

zenity --info --text "Debe ingresar un coen la linea 301mentario"

continue

fi

# https://linuxsimply.com/bash-scripting-tutorial/conditional-statements/if/or-and-not-in-if/

if [ "$contr1" == "" ] || [ "$contr2" == "" ] || [ "$contr1" != "$contr2" ]

then

zenity --info --text "Las contraseñas no pueden ser vacias o no coinciden"

continue

fi

echo validar robustez de la password

if [ "$interprete" == " " ]

then

zenity --info --text "Interprete ajustado automaticamente a /bin/bash"

interprete=/bin/bash

fi

# ensamblo en variable el comando useradd

vusradd="useradd -d $home\_dir -m -c '$comentario' -s $interprete $usr"

echo '$vusradd = ' $vusradd

echo $vusradd >scrusradd

chmod u+x scrusradd

./scrusradd

# rm scrusradd

# ejecuto la variable

# "$vusradd"

# ($vusradd)

# cambiamos el password

echo $usr:$contr1 | chpasswd

exit

echo tambien ver del mail el buscar contrasena que sea robusta

user=zenity --entry --width=300 --text "Nombre del usuario" --title "$2"

cod=echo $?

if [ $cod == 0 ] #Pulso aceptar o cerrar

then

sudo adduser $user

if [ "$?" == 1 ] # si la salida de adduser es 1, es que el usuario existe

then

zenity --warning --title="AddUser" --text="El usuario $user ya existe"

else

echo "creo el usuario $user" >> /tmp/log #Almaceno las operaciones del usuario

progress "Creando usuario" $user "$2"

fi

elif [ $cod == -1 ] #Error

then

zenity --error --no-wrap --title="Error" --text="Ocurrio un error"

fi

done

;;

2)

group=zenity --entry --width=300 --text "Nombre del grupo" --title "$2" #Recogo el nombre del nuevo grupo

cod=echo $?

if [ $cod == 0 ] #Pulso aceptar o cerrar

then

creargrupo "$group" #llamo a mi funcion crear grupo

if [ "$?" == 0 ] # si creargrupo devuevle 0, todo correcto

then

progress "Creando grupo" $group "$2"

echo "creo el grupo $group" >> /tmp/log

fi

elif [ $cod == -1 ] #Error

then

zenity --error --no-wrap --title="Error" --text="Ocurrio un error"

fi

;;

3)

listaUser=cat /etc/passwd | cut -d: -f1 | sort #extraigo del archivo passwd una lista de los usuarios

ordenada

user=zenity --list --height=300 --title="Elija un usuario" --column="USER" $listaUser #Meto los

usuarios en el list

cod=echo $?

if [ $cod == 0 ] #Pulso Aceptar

then

lsgrp=cat /etc/group | cut -d: -f1 | sort

grp=zenity --list --height=300 --title="Elija un grupo" --column="GRUPOS" $lsgrp

sudo usermod -aG $grp $user # -a de append, que no lo quite de su anterior grupo, y -G para

agregarlo a otro grupo como secundario

zenity --info --no-wrap --title="$2" --text="Añadió a \'$user\' al grupo $grp"

echo "Añadió a $user al grupo $grp" >> /tmp/log

elif [ $cod == -1 ] #Error

then

zenity --error --no-wrap --title="Error" --text="Ocurrio un error"

fi

;;

4)

listaUser=cat /etc/passwd | cut -d: -f1 | sort

user=zenity --list --height=300 --title="Elija una opcion" --column="USER" $listaUser

cod=echo $?

if [ $cod == 0 ] #Pulso aceptar o cerrar

then

zenity --question --title="deluser" --text="Borrar home de $user?"

if [ $? == 0 ]

then

sudo deluser $user --remove-home

echo "Elimino el usuario $user con su home" >> /tmp/log

progress "Eliminando Usuario" $user "$2"

else

sudo deluser $user

echo "Elimino el usuario $user y no su home" >> /tmp/log

progress "Eliminando Usuario" $user "$2"

fi

elif [ $cod == -1 ] #Error

then

zenity --error --no-wrap --title="Error" --text="Ocurrio un error"

fi

;;

5)

listaUser=cat /etc/passwd | cut -d: -f1 | sort

user=zenity --list --height=300 --title="Elija un usuario" --column="USER" $listaUser

cod=echo $?

if [ $cod == 0 ] #Pulso Aceptar

then

lsgrp=cat /etc/group | cut -d: -f1 | sort

grp=zenity --list --height=300 --title="Elija un grupo" --column="GRUPOS" $lsgrp

sudo deluser $user $grp # pasandole a deluser un usuario y un grupo, borra al usuario del grupo

zenity --info --no-wrap --title="$2" --text="Eliminó a \'$user\' del grupo $grp"

echo "Eliminó a $user del grupo $grp" >> /tmp/log

elif [ $cod == -1 ] #Error

then

zenity --error --no-wrap --title="Error" --text="Ocurrio un error"

fi

;;

6)

lsgrp=cat /etc/group | cut -d: -f1 | sort

grp=zenity --list --height=300 --title="Elija un grupo" --column="GRUPOS" $lsgrp

cod=echo $?

if [ $cod == 0 ] #Pulso Aceptar

then

gksudo delgroup $grp

zenity --info --no-wrap --title="$2" --text="El grupo $grp ha sido eliminado"

echo "Elimino el grupo $grp" >> /tmp/log

elif [ $cod == -1 ] #Error

then

zenity --error --no-wrap --title="Error" --text="Ocurrio un error"

fi

;;

7)

lsgrp=cat /etc/group | cut -d: -f1 | sort

grp=zenity --list --height=300 --title="Elija un grupo" --column="GRUPOS" $lsgrp

cod=echo $?

if [ $cod == 0 ] #Pulso Aceptar

then

member=cat /etc/group | grep $grp: | cut -d: -f4 | sort #Localizo el nombre del grupo con grep

$grp: y corto los miembros

zenity --info --no-wrap --title="$2" --text="El grupo $grp tiene de miembros: $member"

echo "Listó los miembros de $grp" >> /tmp/log

elif [ $cod == -1 ] #Error

then

zenity --error --no-wrap --title="Error" --text="Ocurrio un error"

fi

;;

8)

# Obtener la lista de usuarios del sistema que pueden iniciar sesión

lsusr=$(awk -F: '($3 >= 1000 && $3 < 60000) || $1 == "root" || $1 == "tiffosi" {print $1}' /etc/passwd | sort)

# Mostrar la lista de usuarios en un cuadro de diálogo Zenity

usr=$(zenity --list --height=300 --title="Elija un usuario" --column="USUARIOS" $lsusr)

cod=$?

if [ $cod -eq 0 ]; then # Si se pulsó Aceptar

# Obtener información del usuario seleccionado

user\_info=$(getent passwd "$usr")

if [ -n "$user\_info" ]; then

# Extraer el nombre completo y el directorio home

full\_name=$(echo "$user\_info" | cut -d: -f5)

home\_dir=$(echo "$user\_info" | cut -d: -f6)

# Mostrar la información del usuario

zenity --info --no-wrap --title="$usr" --text="Usuario: $usr\nNombre completo: $full\_name\nDirectorio home: $home\_dir"

echo "Mostró información del usuario $usr" >> /tmp/log

else

zenity --error --no-wrap --title="Error" --text="No se encontró información para el usuario $usr"

fi

elif [ $cod -eq 1 ]; then # Si se cerró la ventana

zenity --info --no-wrap --title="Información" --text="Operación cancelada."

else # Error

zenity --error --no-wrap --title="Error" --text="Ocurrió un error"

fi

;;

9)

zenity --info --text="Saliendo del programa." --title="Salir"

exit 0

;;

esac

}

# Gestion de usuarios grafica

function main(){

#seleccion 1 # shortcut para debug rapido

#exit

while [ true ]

do

opcion=zenity --list --height=400 --width=400 --radiolist \

--title="Elija una opcion" \

--column="" --column="Opcion" \

"FALSE" "1 Alta Usuario" \

"FALSE" "2 Crear grupo" \

"FALSE" "3 Añadir usuario/Grupo" \

"FALSE" "4 Eliminar Usuario" \

"FALSE" "5 Eliminar Usuario/Grupo" \

"FALSE" "6 Eliminar grupo" \

"FALSE" "7 Mostrar usuarios de un grupo" \

"FALSE" "8 Listado de Usuarios" \

"TRUE" "9 Salir"

res=$?

if [ $res -ne 0 ]; then

zenity --info --text="Operación cancelada."

break

fi

seleccion "$opcion"

done

}

# codigo principal (todo lo que no esta dentro de una function), segunda parte

# Luego de pasar por la parte inicial de todo este programa, pasa por aca e invoca a una funcion main (no es un nombre especial)

main

echo 'punto no alcanzado nunca (hay exit en la funcion seleccion)' # debo proteger con '' este echo porque tiene ()

# puede haber funciones antes y despues del codigo principal

###################################################################################

#Funcion para usar la barra de progreso

function progress(){

(

echo "0" ; sleep 1

echo "# $1" ; sleep 1

echo "25" ; sleep 1

echo "75" ; sleep 1

echo "100" ; sleep 1

echo "# Finalizado"

) |

zenity --progress \

--title="$3" \

--text="" \

--percentage=0 \

--width=300

if [ "$?" = -1 ] ; then

zenity --error \

--text="Suma Abortada."

fi

}

###################################################################################

function creargrupo(){

gksudo groupadd $1

if [ "$?" == 9 ]

then

zenity --warning --title="Groupadd" --text="El grupo \'$1\' ya existe"

return 1

else

return 0

fi

}

###################################################################################

Codigo de base de datos utilizada:

CREATE DATABASE sigenDB;

USE sigenDB;

create table clientes(

nombre varchar (50) not null,

apellido varchar (50) not null,

ci int not null,

Fecha\_de\_nacimiento DATE,

Departamento VARCHAR(50),

Telefono VARCHAR(13) NOT NULL,

primary key(ci)

);

CREATE TABLE grupos (

id INT AUTO\_INCREMENT PRIMARY KEY,

nombre\_grupo VARCHAR(50) NOT NULL UNIQUE

);

DAR PERMISOS AL ARCHIVO

chmod +x ./(nombre del archivo).format

# Configuración SSH

netstat -punta | grep ssh

nmap -p 22 localhost

service sshd status

ssh localhost

cp /etc/ssh/sshd\_config /etc/ssh/sshd\_config\_backup

vi /etc/ssh/sshd\_config Port 2222

yum provides /usr/sbin/semanage

yum install policycoreutils-python-utils

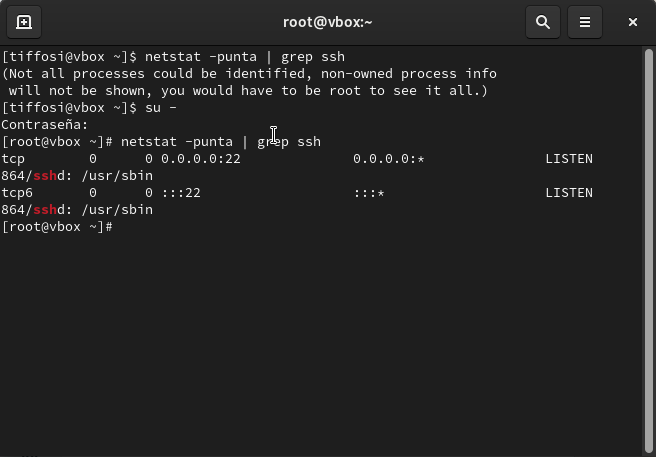
semanage port -a -t ssh\_port\_t -p tcp 2222

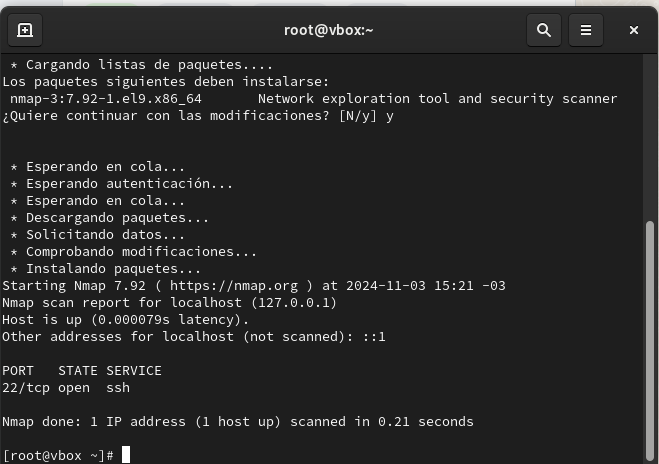
firewall-cmd --zone=public --add-port=2222/tcp --permanent

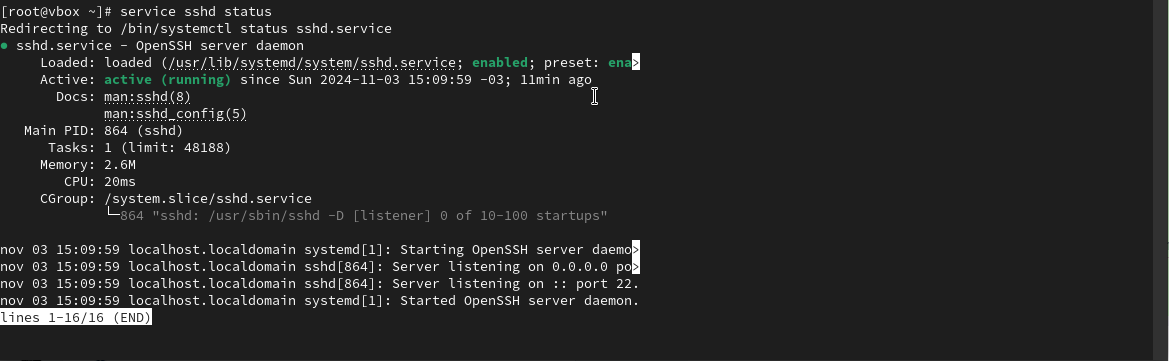
firewall-cmd --reload

systemctl restart sshd

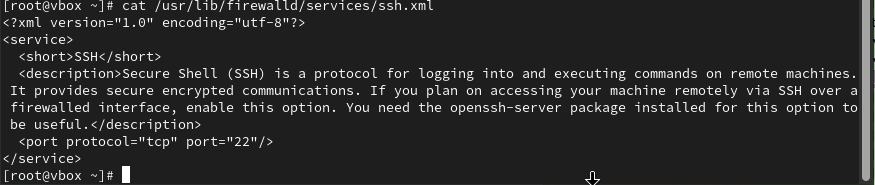
# Capturas de Configuración SSH

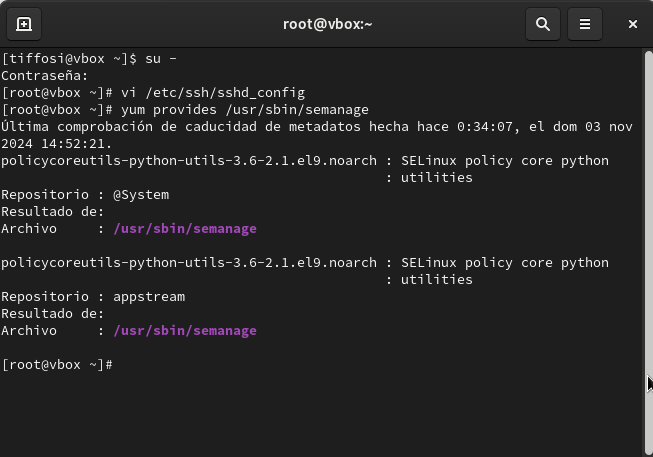


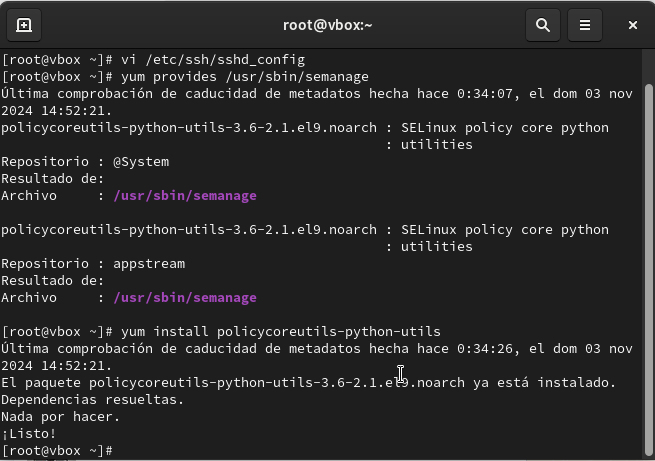












# Configuración del Firewall:

systemctl status firewalld

sudo dnf install firewalld -y

sudo systemctl enable firewalld

sudo systemctl start firewalld

sudo firewall-cmd --state

firewall-cmd --get-default-zone

firewall-cmd --get-active-zones

sudo firewall-cmd --list-all

firewall-cmd --get-zones

sudo firewall-cmd --zone=home --list-all

firewall-cmd --list-all-zones

sudo firewall-cmd --zone=home --change-interface=eth0

firewall-cmd --get-active-zones

sudo firewall-cmd --set-default-zone=home

firewall-cmd --get-services

ls -lrt /usr/lib/firewalld/services

ls -lrt /usr/lib/firewalld/services | less

cat /usr/lib/firewalld/services/ssh.xml

sudo firewall-cmd --zone=public --add-service=http

sudo firewall-cmd --zone=public --list-services

sudo firewall-cmd --zone=public --add-service=http --permanent

sudo firewall-cmd --runtime-to-permanent

sudo firewall-cmd --zone=public --list-services --permanent

sudo firewall-cmd --zone=public --add-service=https

sudo firewall-cmd --zone=public --list-services

sudo firewall-cmd --zone=public --add-service=https --permanent

sudo firewall-cmd --zone=public --list-services --permanent

sudo firewall-cmd --zone=public --add-port=5000/tcp

sudo firewall-cmd --zone=public --list-ports

sudo firewall-cmd --zone=public --add-port=4990-4999/udp

sudo cp /usr/lib/firewalld/services/ssh.xml /etc/firewalld/services/example.xml

ls -lrt /etc/firewalld/services

cat /etc/firewalld/services/example.xml

sudo vi /etc/firewalld/services/example.xml

<?xml version="1.0" encoding="utf-8"?>

<service>

<short>Example Service</short>

<description>This is just an example service. It probably shouldn't be used on a real system.</description>

<port protocol="tcp" port="7777"/>

<port protocol="udp" port="8888"/>

</service>

sudo cat /etc/firewalld/services/example.xml

sudo firewall-cmd --reload

firewall-cmd --get-services

sudo firewall-cmd --permanent --new-zone=publicweb

sudo firewall-cmd --permanent --new-zone=privateDNS

sudo firewall-cmd --permanent --get-zones

sudo firewall-cmd --reload

firewall-cmd --get-zones

sudo firewall-cmd --zone=publicweb --add-service=ssh

sudo firewall-cmd --zone=publicweb --add-service=http

sudo firewall-cmd --zone=publicweb --add-service=https

sudo firewall-cmd --zone=publicweb --list-all

sudo firewall-cmd --zone=privateDNS --add-service=dns

sudo firewall-cmd --zone=privateDNS --list-all

sudo firewall-cmd --zone=publicweb --change-interface=eth0

sudo firewall-cmd --zone=privateDNS --change-interface=eth1

sudo firewall-cmd --runtime-to-permanent

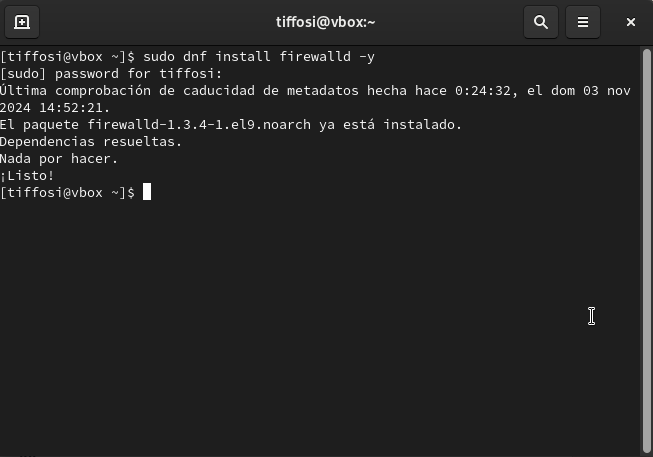
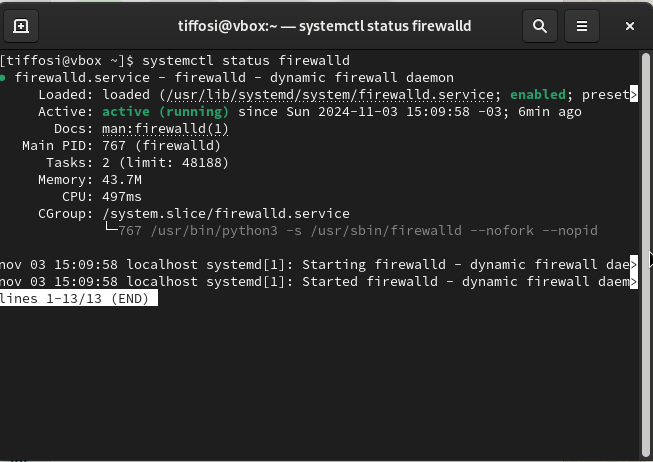
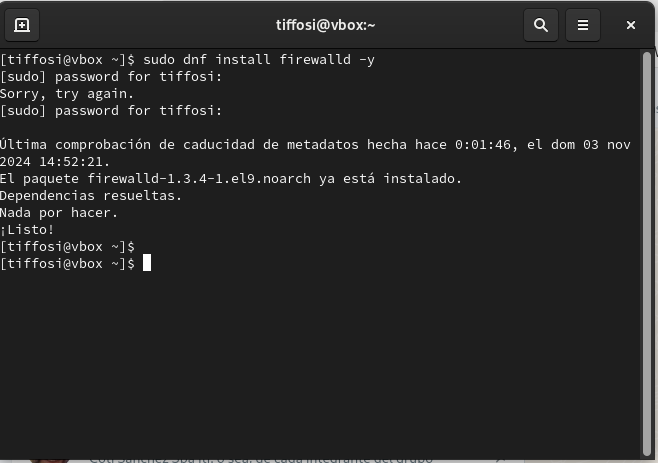
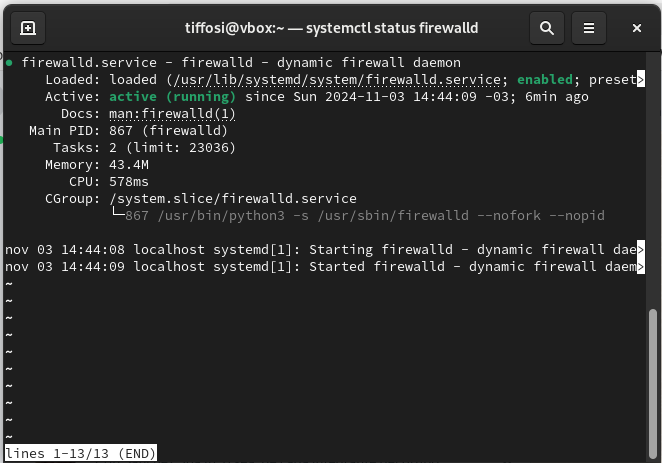
sudo firewall-cmd --reload

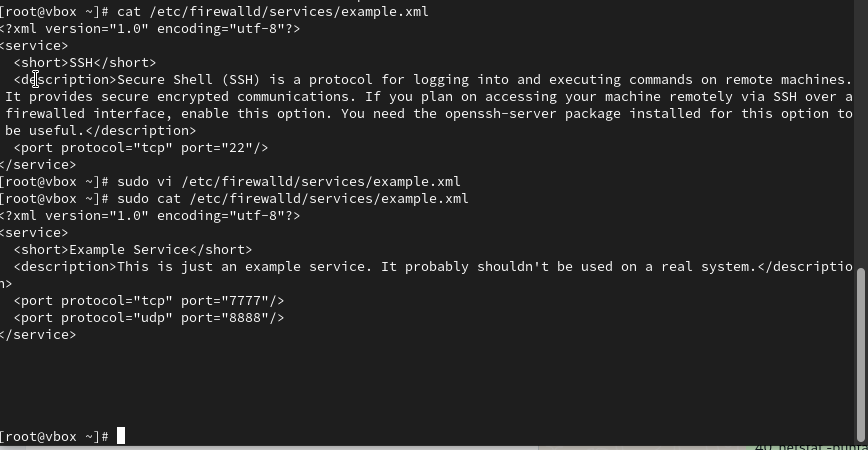
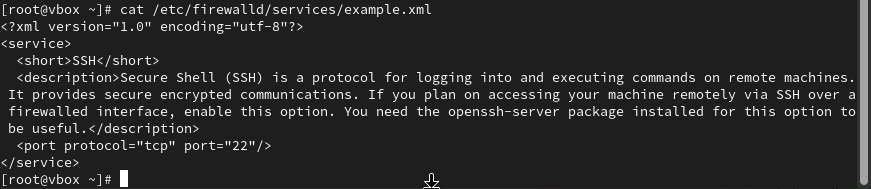
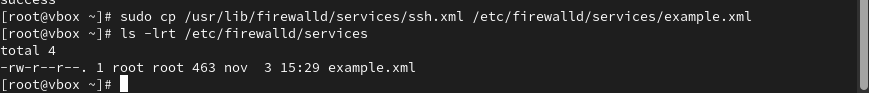
firewall-cmd --get-active-zones

sudo firewall-cmd --zone=publicweb --list-services

sudo firewall-cmd --zone=privateDNS --list-services

sudo firewall-cmd --set-default-zone=publicweb

Capturas del Firewall:



# Capturas del Programa Funcionando:

