

# Obligatorio

## Administración de Servidores Linux

**Profesores:**

Roberto Carlos Wagner Fort

Enrique Manuel Verdes Míguez

**Alumnos:**

Gabriel Somma Nº202801

Pablo Alexis Rivadavia Cawen Nº284705

# Índice

---

<b>Índice.....</b>	<b>2</b>
<b>Declaración de Autoría.....</b>	<b>3</b>
<b>Resumen Ejecutivo.....</b>	<b>3</b>
<b>Aporte individual.....</b>	<b>3</b>
<b>Escenario planteado.....</b>	<b>4</b>
<b>Topología.....</b>	<b>5</b>
<b>A - Ubuntu Controller .....</b>	<b>6</b>
Configuración de Red:.....	6
GIT:.....	6
<b>A - Rocky Linux.....</b>	<b>7</b>
Configuración de red:.....	7
Particiones Rocky Linux: .....	8
Creación usuario Ansible: .....	8
<b>A - Ubuntu.....</b>	<b>9</b>
Configuración de red:.....	9
Particiones Ubuntu: .....	9
Creación usuario Ansible: .....	10
<b>A - Pares de clave .....</b>	<b>11</b>
Creación:.....	11
<b>B - Ansible .....</b>	<b>12</b>
update:.....	12
dbserver: .....	12
webserver:.....	13
tomcat:.....	13
<b>C - Demostración:.....</b>	<b>18</b>
<b>Referencias bibliográficas.....</b>	<b>19</b>

# Declaración de Autoría

---

Pablo Rivadavia y Gabriel Somma estudiantes en la Universidad ORT, en relación con el Obligatorio para la materia Taller de Servidores Linux presentado para su defensa y evaluación en el curso, declaramos que asumimos la originalidad de dicho trabajo, entendida en el sentido de que no ha utilizado fuentes sin citarlas debidamente.

## Resumen Ejecutivo

---

El siguiente documento presenta la solución planteada en resolución a los objetivos y a los requisitos solicitados en la letra del trabajo obligatorio en la materia Taller Linux.

La solución contiene lo necesario para la implementación como también una explicación técnica y detallada sobre los componentes de la misma.

## Aporte individual

---

La implementación de la solución global, en conjunto con las pruebas de rendimiento, evaluación de los requerimientos y documentación fueron repartidas equitativamente entre los dos integrantes del equipo de proyecto, lo que hace complejo discriminar que tareas realizó cada integrante.

Debido a la dinámica aplicada en el grupo de trabajo, se llevaron varias sesiones de trabajo en conjunto a través de Teams, realizando todas las tareas entre los dos miembros del grupo.

En todo momento se mantuvo informado al resto del grupo los progresos en la solución mediante distintos medios entre ellos github, para así mantenernos todos informados de las modificaciones/avances tanto a nivel de documentación como de implementación de la maqueta.

# Escenario planteado

---

Prerrequisitos: Debe tener un equipo bastión (o controlador) con ansible instalado, un usuario para ejecutar los playbooks (recomendamos el usuario ansible) con su clave pública generada.

A) 30% Instalar un servidor Rocky y otro Ubuntu con 1 GB de memoria RAM y 25 GB de Disco (o más),

Configure la instalación con el siguiente esquema de particionamiento

Una partición de 1G para /boot

usando LVM: un volumen lógico de 5GB para /, 5GB para /var, 1GB para /home, 5GB /opt y 2 GB de Swap.

Debe tener 2 interfaces de red, 1 conectada a NAT y la otra a una red Interna o “Solo Anfitrión” que le permita conectarse al equipo bastión con Ansible. Agregue un usuario ansible, dele permisos con SUDO sin contraseña. Desde el equipo bastión o controlador, copie la clave pública para poder conectarse al servidor.

B) 60 % Tareas a realizar con Ansible Debe crear los playbooks necesarios para la instalación y configuración de un servidor Tomcat 9, un servidor de base de datos MySQL o Mariadb incluyendo la creación de la base de datos y el usuario para la aplicación, y un servidor web apache que oficie de proxy reverso. Incluir todas las dependencias que sean necesarias. La aplicación debe ser accesible solamente a través del servidor apache.

El servidor tomcat debe quedar instalado en el /opt, debe ejecutar con el usuario tomcat y grupo tomcat. Debe iniciar mediante un servicio.

El playbook debe poder aplicarse tanto a un servidor con Rocky (tipo rhel), y un servidor con Ubuntu (tipo debian)

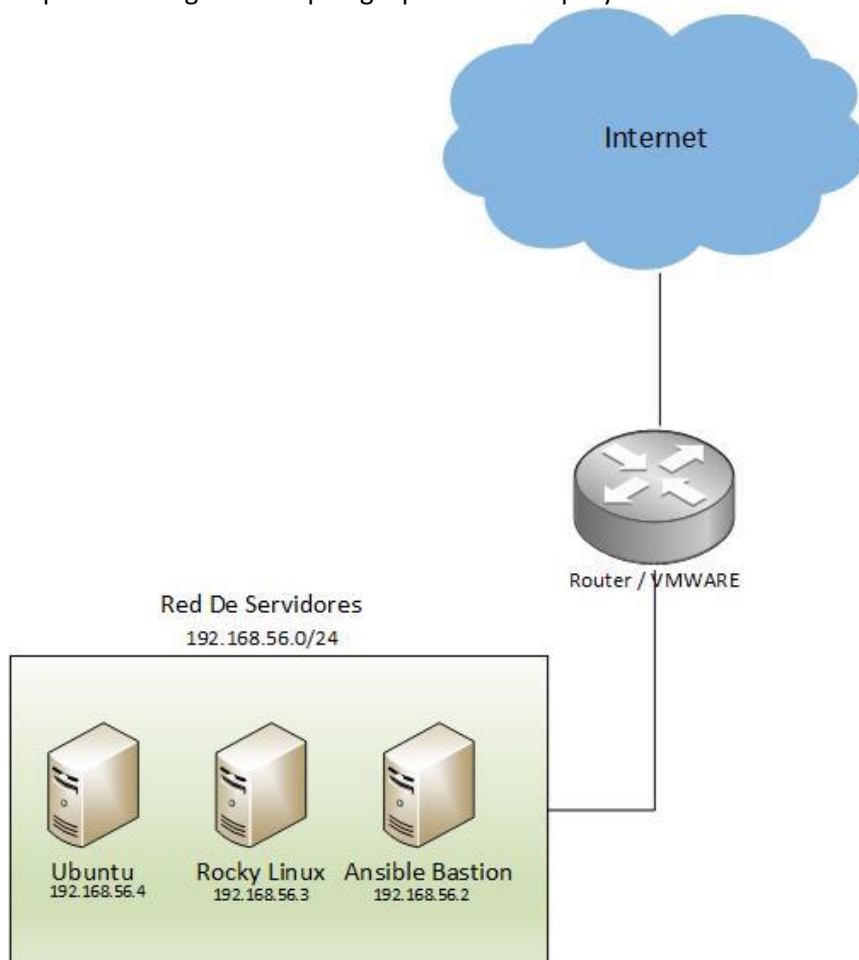
Los playbooks deben ser versionados utilizando git y debe crearse un repositorio remoto en github

C) 10% Documentación Actualice el archivo README.md explicando los cambios realizados y como utilizar el rol. Debe documentar cada paso de las tareas realizadas mediante capturas de pantalla. En el caso del playbook debe mostrar la ejecución correcta de todas las tareas.

Agregue la documentación al repositorio. Descargue el repositorio como archivo zip y súbalo a Gestión como entrega.

# Topología

Se plantea la siguiente topología para nuestro proyecto:



Para configurar esta topología debemos primero definir ciertos parámetros en el programa virtual box:

- 1-Realizamos la instalación de la virtual con la distribución Rocky Linux y Ubuntu.
- 2- Todos los usuarios agregados a sudo serán Ansible y sysadmin (password sysadmin)

64	Rocky_Linux_Taller	Guardada
64	Ubuntu_Taller	Apagada
64	Ansible_Controller	Guardada

En esta topología se utilizarán dos tarjetas de red por servidor, donde la tarjeta 1 estará configurada como NAT con DHCP y una segunda tarjeta configurada como Adaptador solo-anfitrión con IP FIJA.

# A - Ubuntu Controller

---

-Este equipo va a cumplir la función del bastión de ansible, desde este equipo se realizarán las automatizaciones a los otros servidores.  
-El OS de este servidor será Ubuntu  
-Para las dos tarjetas de red al equipo hay que agregarle 2 tarjetas de red con la configuración que se detalló anteriormente, para modificar la configuración de red se debe modificar el yaml siguiente  
/etc/netplan/00-installer-config.yaml  
Instalación de Ansible:  
apt install ansible

## Configuración de Red:

INT=enp0s3 - Conectado como NAT y obtendrá una IP de forma dinámica  
INT=enp0s8 – Conectado como adaptador solo-anfitrión con IP FIJA 192.168.56.2/24

```
# This is the network config written by 'subiquity'
network:
  ethernets:
    enp0s3:
      dhcp4: true
    enp0s8:
      dhcp4: no
      addresses: [192.168.56.2/24 1]
      nameservers:
        addresses: [8.8.8.8, 8.8.4.4]
  version: 2
~
~
~
~
```

## GIT:

-Para la instalación de git se utiliza el comando apt install git en ubuntu  
-Se setea el usuario y mail que se va a utilizar ejemplo:  
git config --global user.username "Nombre Apellido"  
git config --global user.email "email@dominio.com"  
-Para establecer un par de claves se utilizará la misma clave que usaremos para los servidores ubuntu y rocky, la única diferencia es que después la colocaremos dentro de la configuración (GitHub) y DeployKeys

## Deploy keys

Add deploy key



SSH

mf default

SHA256:8/yRoattnV/LHkehc1D340MMyLHw5TUHTMETJx1natM

Added on Feb 15, 2022 by @MFPriest64

Last used within the last week — Read/write

Delete

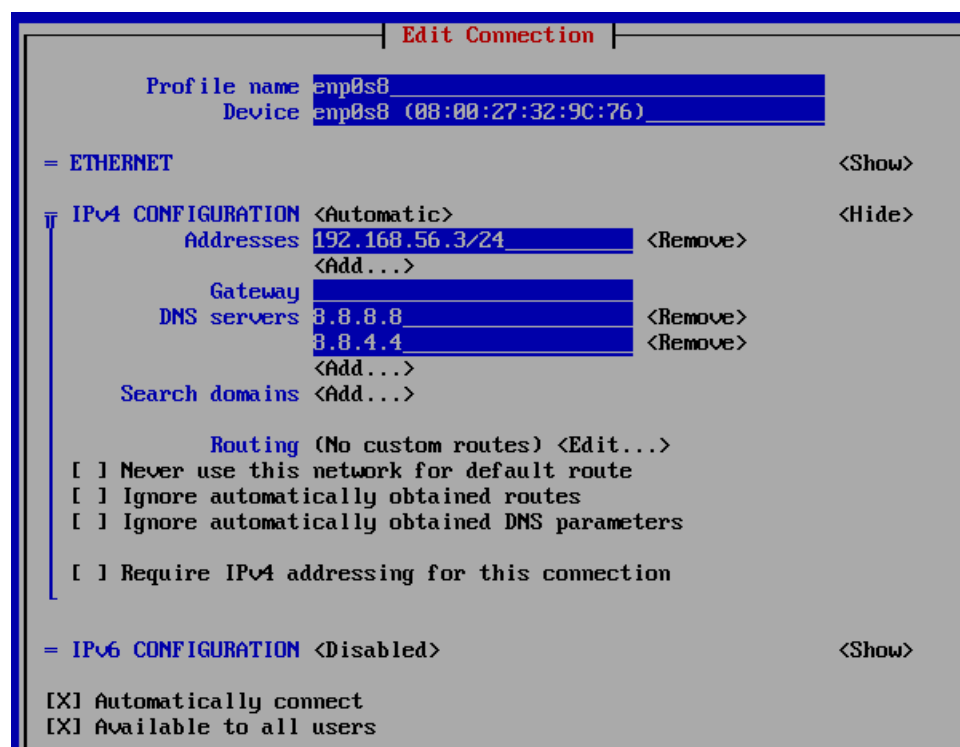
# A - Rocky Linux

-Este servidor será utilizado para instalar los servicios del punto B, para esto se debe preparar como indica el punto A.

## Configuración de red:

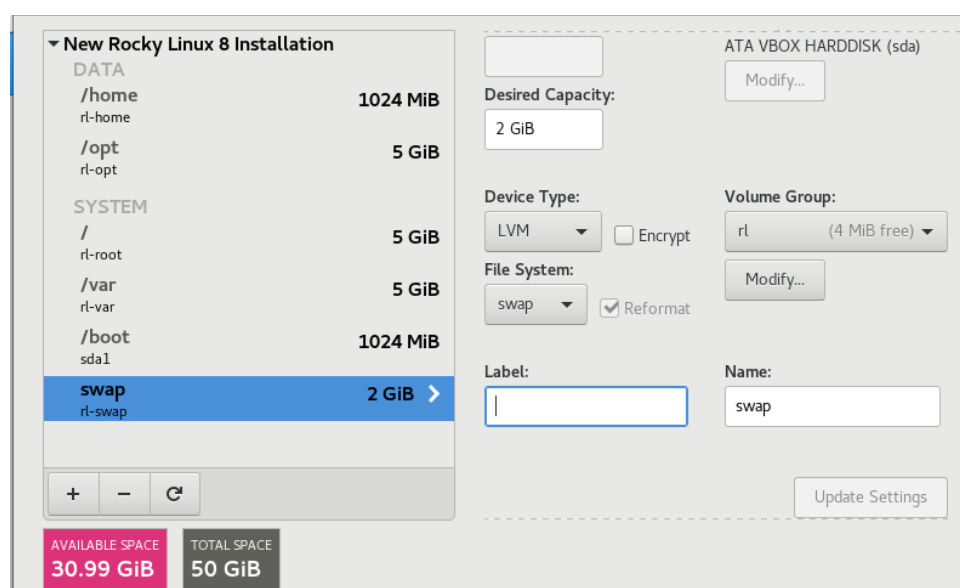
INT=enp0s3 - Conectado como NAT y obtendrá una IP de forma dinámica

INT=enp0s8 – Conectado como adaptador solo-anfitrión con IP FIJA 192.168.56.3/24



## Particiones Rocky Linux:

- 1G para /boot
- LVM
  - 5 GB /
  - 5 GB /var
  - 1 GB /home
  - 5 GB /opt
  - 2 GB Swap



## Creación usuario Ansible:

-Se crea el usuario Ansible con el comentario de uso:

```
useradd -m -c "Usuario para automatismos" Ansible
```

```
[root@localhost conf.d]# getent passwd | grep Ansible
Ansible:x:1001:1001:Usuario para automatismos:/home/Ansible:/bin/bash
```

-Para elevar privilegios sin contraseña se debe de agregar la linea Ansible `ALL=(ALL) NOPASSWD:ALL` al archivo `sudoers.d` (visudo)



```

defaults    secure_path = /sbin:/bin:/usr/sbin:/usr/bin

## Next comes the main part: which users can run what software on
## which machines (the sudoers file can be shared between multiple
## systems).
## Syntax:
##
##      user    MACHINE=COMMANDS
##
## The COMMANDS section may have other options added to it.
##
## Allow root to run any commands anywhere
root    ALL=(ALL)    ALL

## Allows members of the 'sys' group to run networking, software,
## service management apps and more.
# %sys ALL = NETWORKING, SOFTWARE, SERVICES, STORAGE, DELEGATING, PROCESSES, LOCATE, DRIVERS

## Allows people in group wheel to run all commands
%wheel  ALL=(ALL)    ALL

## Same thing without a password
# %wheel    ALL=(ALL)    NOPASSWD: ALL

## Allows members of the users group to mount and unmount the
## cdrom as root
# %users    ALL=/sbin/mount /mnt/cdrom, /sbin/umount /mnt/cdrom

## Allows members of the users group to shutdown this system
# %users    localhost=/sbin/shutdown -h now

## Read drop-in files from /etc/sudoers.d (the # here does not mean a comment)
#includedir /etc/sudoers.d

Ansible ALL=(ALL) NOPASSWD:ALL

```

## A - Ubuntu

---

-Este servidor será utilizado para instalar los servicios del punto B, para esto se debe preparar como indica el punto A.

### Configuración de red:

INT=enp0s3 - Conectado como NAT y obtendrá una IP de forma dinámica

INT=enp0s8 – Conectado como adaptador solo-anfitrión con IP FIJA 192.168.56.4/24

```

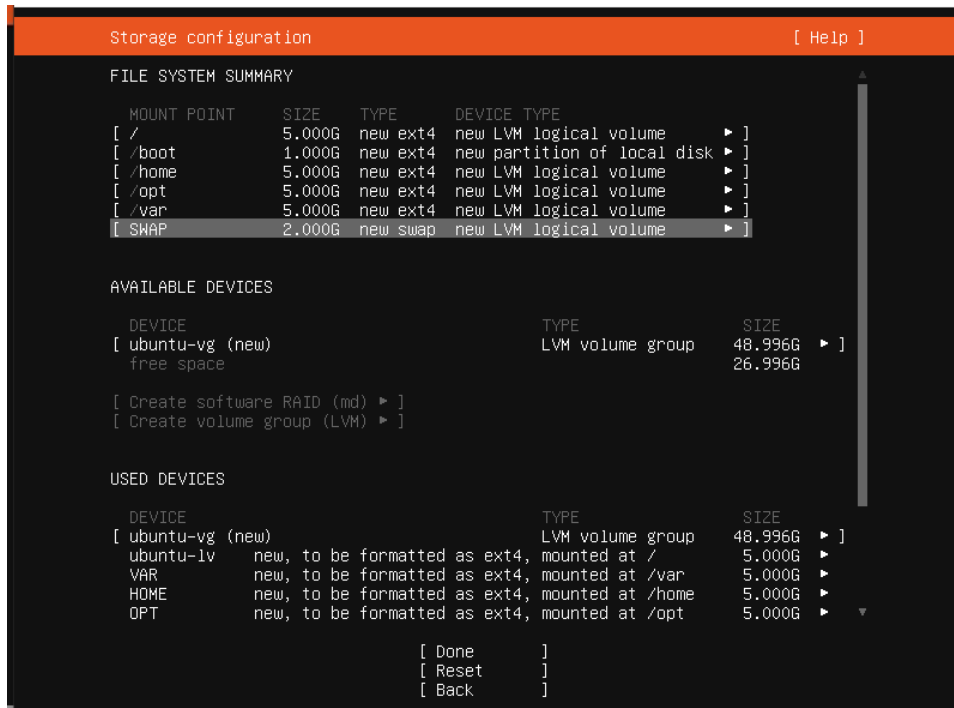
# This is the network config written by 'subiquity'
network:
  ethernets:
    enp0s3:
      dhcp4: true
    enp0s8:
      dhcp4: no
      addresses: [192.168.56.4/24 ]
      nameservers:
        addresses: [8.8.8.8, 8.8.4.4]
  version: 2
~
~

```

### Particiones Ubuntu:

- 1G para /boot
- LVM

- 5 GB /
- 5 GB /var
- 1 GB /home
- 5 GB /opt
- 2 GB Swap



## Creación usuario Ansible:

-Se crea el usuario Ansible con el comentario de uso:

```
useradd -m -c "Usuario para automatismos" Ansible
```

```
Ansible:x:1001:1001:Usuario para automatismos:/home/Ansible:/bin/sh
root@sysadmin:/etc/netplan# getent passwd | grep Ansible
```

-Para elevar privilegios sin contraseña se debe de agregar la línea `Ansible ALL=(ALL) NOPASSWD:ALL` con el comando visudo

```

Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
GNU nano 4.8 /etc/sudoers.tmp
# This file MUST be edited with the 'visudo' command as root.
#
# Please consider adding local content in /etc/sudoers.d/ instead of
# directly modifying this file.
# See the man page for details on how to write a sudoers file.
#
Defaults        env_reset
Defaults        mail_badpass
Defaults        secure_path="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/snap/bin"

# Host alias specification

# User alias specification

# Cmnd alias specification

# User privilege specification
root    ALL=(ALL:ALL) ALL

# Members of the admin group may gain root privileges
%admin   ALL=(ALL) ALL

# Allow members of group sudo to execute any command
%sudo   ALL=(ALL:ALL) ALL

# See sudoers(5) for more information on "#include" directives:

#includedir /etc/sudoers.d

Ansible  ALL=(ALL) NOPASSWD:ALL

```

## A - Pares de clave

---

Una vez que tenemos todos los servidores levantados con SSH y direcciones IP correctamente configuradas, debemos crear las claves publicas hacia cada uno de los servidores.

### Creación:

**ssh-keygen** (por defecto nos crea en el directorio `/root/.ssh`, nos creara una clave privada y una pública .pub)

```

Enter file in which to save the key (/root/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa
Your public key has been saved in /root/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:3Gx+kuB/hMQrQXxo4HYxQgri5hC17atoX76f1YnwNEs root@sysadmin
The key's randomart image is:
+----[RSA 3072]-----+
|o.o  .+oo.          |
|.o + o ..=o.         |
|.o. o oo.o          |
|+  . . 0.o0          |
| . . .SE+o           |
| . . .*+B.o          |
|..  .*+++.           |
| o .o  o. o.         |
|o o. 000  ..         |
+----[SHA256]-----+

```

```

root@sysadmin:~/.ssh# ls -ls
total 36
4 -rw----- 1 root root 399 Feb 15 15:18 ansible
4 -rw-r--r-- 1 root root 89 Feb 15 15:18 ansible.pub
4 -rw-r--r-- 1 root root 11 Feb 18 00:21 apt
0 -rw----- 1 root root 0 Feb 14 04:41 authorized_keys
4 drwxr-xr-x 4 root root 4096 Feb 18 15:07 GitTaller
4 -rw----- 1 root root 464 Feb 15 14:31 id_ed25519
4 -rw-r--r-- 1 root root 97 Feb 15 14:31 id_ed25519.pub
4 -rw----- 1 root root 2602 Feb 18 18:31 id_rsa
4 -rw-r--r-- 1 root root 567 Feb 18 18:31 id_rsa.pub
4 -rw-r--r-- 1 root root 2218 Feb 17 15:05 known_hosts
root@sysadmin:~/.ssh#

```

Para exportar las claves a los otros servidores se utiliza el siguiente comando:

```
ssh-copy-id -i <path> <usuario>@<ip>
```

Ejemplo:

```
ssh-copy-id -i /root/.ssh/id_rsa.pub sysadmin@192.168.9.4
```

(por defecto en los ubuntu el usuario root no puede autenticarse por ssh)

```

root@sysadmin:~/.ssh# ssh-copy-id -i /root/.ssh/id_rsa.pub sysadmin@192.168.9.4
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/root/.ssh/id_rsa.pub"
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install all the new keys
Enter passphrase for key '/root/.ssh/id_ed25519':
sysadmin@192.168.9.4's password:

Number of key(s) added: 1

Now try logging into the machine, with: "ssh 'sysadmin@192.168.9.4'"
and check to make sure that only the key(s) you wanted were added.

root@sysadmin:~/.ssh#

```

## B - Ansible

La parte B del procedimiento implica la ejecución del playbook creado en Ansible, el cual se compone de 4 roles, cabe aclarar que todos los roles están implementados tanto para sistemas operativos de la familia Redhat como de la familia Debian, en este caso, ejecutamos el playbook en un sistema Rocky y Ubuntu respectivamente. La descripción general de los roles sería la siguiente:

**update:** Este rol consiste en la instalación de los paquetes básicos necesarios para correr la aplicación web solicitada como requerimiento, así como también la habilitación del firewall con SSH habilitado.

**dbserver:** Este rol se encarga de la implementación de todo lo relacionado a la base de datos del sistema, desde iniciar el servicio MariaDB o MySQL, hasta la creación del usuario que utilizará la base de datos, incluyendo también la creación de la misma con las tablas correspondientes.

**webserver:** Este rol especifica todas las tareas necesarias para implementar el apache como reverse proxy, incluyendo también todas las configuraciones pertinentes.

**tomcat:** Este último rol implica todas las tareas relacionadas al correcto funcionamiento del servidor tomcat, incluyendo el despliegue de la aplicación web solicitada como requerimiento.

A continuación, incluimos el resultado de la ejecución del playbook en los equipos previamente mencionados. El comando para ejecutar dicho playbook es:

```
Run ansible-playbook main.yml como usuario Ansible
```

```
sysadmin@sysadmin:~/GitTaller$ ansible-playbook main.yml
```

```
PLAY [Update de servidores] *****
```

```
TASK [Gathering Facts] *****
```

```
ok: [rocky01]
```

```
ok: [ubuntu01]
```

```
TASK [update : Incluir variables] *****
```

```
ok: [rocky01]
```

```
ok: [ubuntu01]
```

```
TASK [update : Actualizar paquetes en Red Hat] *****
```

```
skipping: [ubuntu01]
```

```
changed: [rocky01]
```

```
TASK [update : Actualizar paquetes en Debian] *****
```

```
skipping: [rocky01]
```

```
changed: [ubuntu01]
```

```
TASK [update : Instalar Epel en Rocky] *****
```

```
skipping: [ubuntu01]
```

```
changed: [rocky01]
```

```
TASK [update : Firewall activado con ssh habilitado en Debian] *****
```

```
skipping: [rocky01] => (item=22)
```

```
changed: [ubuntu01] => (item=22)
```

```
TASK [update : Firewall activado con ssh habilitado en RedHat] *****
```

```
skipping: [ubuntu01] => (item=22/tcp)
```

```
changed: [rocky01] => (item=22/tcp)
```

TASK [update : Instalar paquetes en RedHat] \*\*\*\*\*  
skipping: [ubuntu01] => (item=httpd)  
skipping: [ubuntu01] => (item=java-17-openjdk)  
skipping: [ubuntu01] => (item=mysql)  
skipping: [ubuntu01] => (item=mysql-server)  
skipping: [ubuntu01] => (item=tar)  
skipping: [ubuntu01] => (item=python3-PyMySQL)  
changed: [rocky01] => (item=httpd)  
changed: [rocky01] => (item=java-17-openjdk)  
changed: [rocky01] => (item=mysql)  
changed: [rocky01] => (item=mysql-server)  
changed: [rocky01] => (item=tar)  
changed: [rocky01] => (item=python3-PyMySQL)

TASK [update : Instalar paquetes en Debian] \*\*\*\*\*  
skipping: [rocky01] => (item=apache2)  
skipping: [rocky01] => (item=openjdk-17-jdk)  
skipping: [rocky01] => (item=mariadb-server)  
skipping: [rocky01] => (item=mariadb-client)  
skipping: [rocky01] => (item=policycoreutils-python-utils)  
skipping: [rocky01] => (item=python3-pip)  
skipping: [rocky01] => (item=python3-mysqldb)  
skipping: [rocky01] => (item=default-jdk)  
changed: [ubuntu01] => (item=apache2)  
changed: [ubuntu01] => (item=openjdk-17-jdk)  
changed: [ubuntu01] => (item=mariadb-server)  
changed: [ubuntu01] => (item=mariadb-client)  
changed: [ubuntu01] => (item=policycoreutils-python-utils)  
changed: [ubuntu01] => (item=python3-pip)  
changed: [ubuntu01] => (item=python3-mysqldb)  
changed: [ubuntu01] => (item=default-jdk)

RUNNING HANDLER [update : Reiniciar servidor] \*\*\*\*\*  
changed: [ubuntu01]  
changed: [rocky01]

PLAY [Desplegar Servidores Web] \*\*\*\*\*

TASK [Gathering Facts] \*\*\*\*\*  
ok: [ubuntu01]  
ok: [rocky01]

TASK [webserver : Incluir variables] \*\*\*\*\*  
ok: [rocky01]  
ok: [ubuntu01]

TASK [webserver : Firewall activar puerto 80 Red Hat] \*\*\*\*\*  
skipping: [ubuntu01] => (item=80/tcp)  
changed: [rocky01] => (item=80/tcp)

TASK [webserver : Firewall activar puerto 80 Debian] \*\*\*\*\*

skipping: [rocky01] => (item=80)  
changed: [ubuntu01] => (item=80)

TASK [webserver : RedHat - Permitir a httpd conectar con mysql] \*\*\*\*\*  
skipping: [ubuntu01]  
changed: [rocky01]

TASK [webserver : Configuración Proxy Reverso RedHat] \*\*\*\*\*  
skipping: [ubuntu01]  
changed: [rocky01]

TASK [webserver : Configuración Proxy Reverso Debian] \*\*\*\*\*  
skipping: [rocky01]  
changed: [ubuntu01]

RUNNING HANDLER [webserver : Reiniciar servicio httpd] \*\*\*\*\*  
changed: [rocky01]

RUNNING HANDLER [webserver : Reiniciar servicio apache] \*\*\*\*\*  
changed: [ubuntu01]

PLAY [Desplegar servidor de bases de datos] \*\*\*\*\*

TASK [Gathering Facts] \*\*\*\*\*  
ok: [ubuntu01]  
ok: [rocky01]

TASK [dbserver : Incluir variables] \*\*\*\*\*  
ok: [rocky01]  
ok: [ubuntu01]

TASK [dbserver : Iniciar MySql en RedHat] \*\*\*\*\*  
skipping: [ubuntu01]  
changed: [rocky01]

TASK [dbserver : Iniciar MariaDB en Debian] \*\*\*\*\*  
skipping: [rocky01]  
ok: [ubuntu01]

TASK [dbserver : RedHat - Permitir conexiones a mysql - SELinux] \*\*\*\*\*  
skipping: [ubuntu01]  
changed: [rocky01]

TASK [dbserver : RedHat - Iniciar y habilitar MySQL] \*\*\*\*\*  
skipping: [ubuntu01]  
ok: [rocky01]

TASK [dbserver : Debian - Iniciar y habilitar MySQL] \*\*\*\*\*  
skipping: [rocky01]  
ok: [ubuntu01]

TASK [dbserver : RedHat - Creacion regla de firewall MariaDB] \*\*\*\*\*  
skipping: [ubuntu01]  
changed: [rocky01]

TASK [dbserver : Debian - Creacion regla de firewall MySQL] \*\*\*\*\*  
skipping: [rocky01]  
changed: [ubuntu01]

TASK [dbserver : RedHat - Crear usuario en DB] \*\*\*\*\*  
skipping: [ubuntu01]  
[WARNING]: Module did not set no\_log for update\_password  
changed: [rocky01]

TASK [dbserver : Debian - Crear usuario en DB] \*\*\*\*\*  
skipping: [rocky01]  
changed: [ubuntu01]

TASK [dbserver : RedHat - Crear DB de aplicacion] \*\*\*\*\*  
skipping: [ubuntu01]  
changed: [rocky01]

TASK [dbserver : Debian - Crear DB de aplicacion] \*\*\*\*\*  
skipping: [rocky01]  
changed: [ubuntu01]

TASK [dbserver : Mover archivo creacion de tablas todo.sql] \*\*\*\*\*  
skipping: [ubuntu01]  
changed: [rocky01]

TASK [dbserver : RedHat - Crear tabla DB de aplicacion] \*\*\*\*\*  
skipping: [ubuntu01]  
changed: [rocky01]

TASK [dbserver : Mover archivo creacion de tablas todo.sql] \*\*\*\*\*  
skipping: [rocky01]  
changed: [ubuntu01]

TASK [dbserver : Debian - Crear tabla DB de aplicacion] \*\*\*\*\*  
skipping: [rocky01]  
changed: [ubuntu01]

TASK [dbserver : Debian - Eliminar usuario anonimo] \*\*\*\*\*  
skipping: [rocky01]  
ok: [ubuntu01]

TASK [dbserver : RedHat - Eliminar usuario anonimo] \*\*\*\*\*  
skipping: [ubuntu01]  
ok: [rocky01]

PLAY [Desplegar servidor Tomcat] \*\*\*\*\*



TASK [Gathering Facts] \*\*\*\*\*  
ok: [rocky01]  
ok: [ubuntu01]

TASK [tomcat : Incluir variables] \*\*\*\*\*  
ok: [rocky01]  
ok: [ubuntu01]

TASK [tomcat : Agregar directorio tomcat] \*\*\*\*\*  
changed: [ubuntu01]  
changed: [rocky01]

TASK [tomcat : Agregar grupo tomcat] \*\*\*\*\*  
changed: [ubuntu01]  
changed: [rocky01]

TASK [tomcat : Agregar usuario 'tomcat'] \*\*\*\*\*  
changed: [ubuntu01]  
changed: [rocky01]

TASK [tomcat : Descargar e instalar tomcat] \*\*\*\*\*  
changed: [ubuntu01]  
changed: [rocky01]

TASK [tomcat : Cambiar owner de directorio opt/tomcat usuario tomcat] \*\*\*\*\*  
changed: [ubuntu01]  
changed: [rocky01]

TASK [tomcat : Instalar servicio Tomcat para redhat] \*\*\*\*\*  
skipping: [ubuntu01]  
changed: [rocky01]

TASK [tomcat : Instalar servicio Tomcat para ubuntu] \*\*\*\*\*  
skipping: [rocky01]  
changed: [ubuntu01]

TASK [tomcat : Tomcat está habilitado e iniciado] \*\*\*\*\*  
changed: [rocky01]  
changed: [ubuntu01]

TASK [tomcat : Copiar archivo todo.war a servidores] \*\*\*\*\*  
changed: [ubuntu01]  
changed: [rocky01]

TASK [tomcat : Agregar directorio config] \*\*\*\*\*  
changed: [ubuntu01]  
changed: [rocky01]

TASK [tomcat : Copiar archivo app.properties a opt/config] \*\*\*\*\*  
changed: [ubuntu01]  
changed: [rocky01]

TASK [tomcat : Firewall activar puerto tomcat RedHat] \*\*\*\*\*  
skipping: [ubuntu01] => (item=8080/tcp)  
changed: [rocky01] => (item=8080/tcp)

TASK [tomcat : Firewall activar puerto tomcat Debian] \*\*\*\*\*  
skipping: [rocky01] => (item=8080)  
changed: [ubuntu01] => (item=8080)

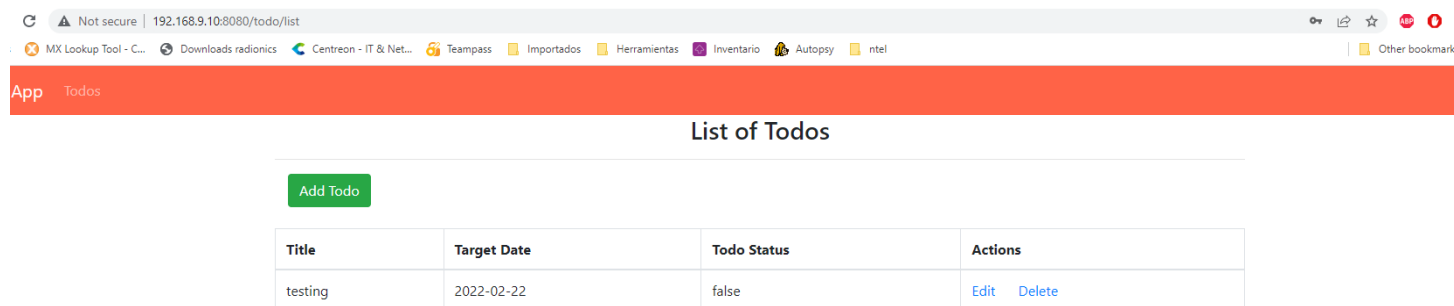
RUNNING HANDLER [tomcat : Reiniciar servicio] \*\*\*\*\*  
changed: [rocky01]  
changed: [ubuntu01]

RUNNING HANDLER [tomcat : Reiniciar firewall] \*\*\*\*\*  
changed: [rocky01]

PLAY RECAP \*\*\*\*\*  
rocky01 : ok=39 changed=29 unreachable=0 failed=0 skipped=15 rescued=0 ignored=0  
ubuntu01 : ok=35 changed=24 unreachable=0 failed=0 skipped=18 rescued=0 ignored=0

## C - Demostración:

A continuación, se detallan unas capturas de la aplicación web en correcto funcionamiento, demostrando la posibilidad de crear tareas (testing) con un usuario creado denominado “Gabriel”. Las pruebas fueron realizadas tanto en Rocky como en Ubuntu con éxito.



Not secure | 192.168.9.10:8080/todo/list

MX Lookup Tool - C... Downloads radionics Centreon - IT & Net... Teampass Importados Herramientas Inventario Autopsy intel Other bookmark

App Todos

### List of Todos

Add Todo

Title	Target Date	Todo Status	Actions
testing	2022-02-22	false	<a href="#">Edit</a> <a href="#">Delete</a>

## List of Todos

Add Todo

Title	Target Date	Todo Status	Actions
testing	2022-02-22	false	<a href="#">Edit</a> <a href="#">Delete</a>

## Referencias bibliográficas

- Material dicho en clase
- Clases grabadas en Aulas
- Material subido en Aulas