

**OBLIGATORIO**  
**TALLER DE SERVIDORES**  
**LINUX**

**VALENTÍN RODRÍGUEZ**

**JUAN MÓNIGO**

**PROFESOR: ALEJANDRO**  
**VARTABEDIAN**

**FECHA: 6/8/2024**

# ÍNDICE

PARTE A.....	3
Ansible.....	3
GIT:.....	4
Configuración SSH: .....	5
PARTE B.....	7
UBUNTU .....	7
CentOS: .....	8
PARTE C .....	9
PARTE D .....	14

# PARTE A

## Ansible

### Instalación de ansible

```
ansible@controlador-ansible:~/obligatorio$ sudo apt install pipx -y
[sudo] password for ansible:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  python3-argcomplete python3-packaging python3-pip-whl python3-platformdirs python3-psutil python3-setuptools-whl python3-userpath python3-venv
The following NEW packages will be installed:
  pipx python3-argcomplete python3-packaging python3-pip-whl python3-platformdirs python3-psutil python3-setuptools-whl python3-userpath python3-venv
0 upgraded, 10 newly installed, 0 to remove and 5 not upgraded.
Need to get 3,506 kB of archives.
After this operation, 7,835 kB of additional disk space will be used.
Get:1 http://uy.archive.ubuntu.com/ubuntu noble/universe amd64 python3-pip-whl all 24.0+dfsg-1ubuntu1 [1,702 kB]
Get:2 http://uy.archive.ubuntu.com/ubuntu noble/universe amd64 python3-setuptools-whl all 68.1.2-2ubuntu1 [715 kB]
Get:3 http://uy.archive.ubuntu.com/ubuntu noble-updates/universe amd64 python3.12-venv amd64 3.12.3-1ubuntu0.1 [5,678 B]
Get:4 http://uy.archive.ubuntu.com/ubuntu noble/universe amd64 python3-venv amd64 3.12.3-0ubuntu1 [1,032 B]
Get:5 http://uy.archive.ubuntu.com/ubuntu noble/universe amd64 python3-argcomplete all 3.1.4-1 [33.5 kB]
Get:6 http://uy.archive.ubuntu.com/ubuntu noble/main amd64 python3-packaging all 24.0-1 [41.1 kB]
Get:7 http://uy.archive.ubuntu.com/ubuntu noble/main amd64 python3-platformdirs all 4.2.0-1 [16.1 kB]
Get:8 http://uy.archive.ubuntu.com/ubuntu noble/universe amd64 python3-userpath all 1.9.1-1 [9,416 B]
Get:9 http://uy.archive.ubuntu.com/ubuntu noble/universe amd64 pipx all 1.4.3-1 [787 kB]
Get:10 http://uy.archive.ubuntu.com/ubuntu noble/main amd64 python3-psutil amd64 5.9.8-2build2 [195 kB]
Fetched 3,506 kB in 1s (5,673 kB/s)
Selecting previously unselected package python3-pip-whl.
```

```
ansible@controlador-ansible:~/obligatorio$ pipx install --include-deps ansible
installed package ansible 10.2.0, installed using Python 3.12.3
These apps are now globally available
- ansible
- ansible-community
- ansible-config
- ansible-connection
- ansible-console
- ansible-doc
- ansible-galaxy
- ansible-inventory
- ansible-playbook
- ansible-pull
- ansible-test
- ansible-vault

⚠ Note: '/home/ansible/.local/bin' is not on your PATH environment variable. These apps will not be globally accessible until your PATH is
done! ✨ ✨ ✨

ansible@controlador-ansible:~/obligatorio$ pipx ensurepath
/home/ansible/.local/bin has been added to PATH, but you need to open a new terminal or re-login for this PATH change to take effect.

You will need to open a new terminal or re-login for the PATH changes to take effect.

Otherwise pipx is ready to go! ✨ ✨ ✨

ansible@controlador-ansible:~/obligatorio$ pipx inject ansible
usage: pipx inject [-h] [--quiet] [--verbose] [--include-apps] [--include-deps] [--system-site-packages] [--index-url INDEX_URL] [--editable]
pipx inject: error: the following arguments are required: dependencies

ansible@controlador-ansible:~/obligatorio$ pipx inject ansible argcomplete
injected package argcomplete into venv ansible
done! ✨ ✨ ✨

ansible@controlador-ansible:~/obligatorio$ pipx inject ansible ansible-lint
injected package ansible-lint into venv ansible
done! ✨ ✨ ✨

ansible@controlador-ansible:~/obligatorio$ activate-global-python-argcomplete
Defaulting to system-wide installation.
usage: activate-global-python-argcomplete [-h] [--dest DEST] [--user]
activate-global-python-argcomplete: error: path /usr/local/share/zsh/site-functions does not exist and could not be created: [Errno 13] Permi
ansible@controlador-ansible:~/obligatorio$ activate-global-python-argcomplete --user
Adding shellcode to /home/ansible/.zshenv...
Added.
Adding shellcode to /home/ansible/.bash_completion...
Added.
Please restart your shell or source the installed file to activate it.

ansible@controlador-ansible:~/obligatorio$ cd ..
ansible@controlador-ansible:~$ cat .bash
.bash_completion .bash_history .bash_logout .bashrc
ansible@controlador-ansible:~$ cat .bash_
.bash_completion .bash_history .bash_logout
ansible@controlador-ansible:~$ cat .bash_completion

# Begin added by argcomplete
source "/usr/lib/python3/dist-packages/argcomplete/bash_completion.d/_python-argcomplete"
# End added by argcomplete

ansible@controlador-ansible:~$
```

## GIT:

```
drwxr-x-- 6 ansible ansible 4096 Jul 25 17:18 ../
drwxrwxr-x 7 ansible ansible 4096 Jul 25 17:19 .git/
ansible@controlador-ansible:~/obligatorio$ git config --global user.name "Alumno ORT"
ansible@controlador-ansible:~/obligatorio$ git config --global user.email "alumno@ort.edu.uy"
ansible@controlador-ansible:~/obligatorio$ git config --global color.ui true
ansible@controlador-ansible:~/obligatorio$ git config --global color.status auto
ansible@controlador-ansible:~/obligatorio$ git config --global color.branch auto
ansible@controlador-ansible:~/obligatorio$ git c
checkout cherry cherry-pick citool clean clone commit config
ansible@controlador-ansible:~/obligatorio$ git c
checkout cherry cherry-pick citool clean clone commit config
ansible@controlador-ansible:~/obligatorio$ git config --global core.editor nano
ansible@controlador-ansible:~/obligatorio$ git config --global merge.tool vimdiff
ansible@controlador-ansible:~/obligatorio$ git co
commit config
ansible@controlador-ansible:~/obligatorio$ git co
commit config
ansible@controlador-ansible:~/obligatorio$ git config --list
user.name=Alumno ORT
user.email=alumno@ort.edu.uy
color.ui=true
color.status=auto
color.branch=auto
core.editor=nano
merge.tool=vimdiff
core.repositoryformatversion=0
core.filemode=true
core.bare=false
core.logallrefupdates=true
ansible@controlador-ansible:~/obligatorio$
```

```
ansible@controlador-ansible:~/obligatorio$ git log
commit 928159829dc79f28cba2f040a9348d36654c1af1 (HEAD -> master)
Author: Alumno ORT <alumno@ort.edu.uy>
Date: Thu Jul 25 17:31:02 2024 +0000

    commit de prueba
ansible@controlador-ansible:~/obligatorio$
```

## Configuración SSH:

Configuramos SSH para poder administrar el controlador ansible desde la máquina host.

```
ansible@controlador-ansible: X + v
Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

Last login: Tue Jul 23 17:12:12 2024 from 10.0.2.2
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ansible@controlador-ansible:~$ ssh-keygen
Generating public/private ed25519 key pair.
Enter file in which to save the key (/home/ansible/.ssh/id_ed25519):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/ansible/.ssh/id_ed25519
Your public key has been saved in /home/ansible/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:IWBi/nC84w3p/BvW+A4J7HutXfPIo/0/0EC9c3AkUrI ansible@controlador-ansible
The key's randomart image is:
+--[ED25519 256]--+
| o o   o.o . |
| o + .   = o |
| o o . . E o . |
| = o . o +   |
|  O S . o . |
| = +   o o   |
| = B.. + .   |
| +.=oo.=     |
| ..++=+o+.   |
+-----[SHA256]-----+
ansible@controlador-ansible:~$
```

```
ansible@controlador-ansible:~$ ssh sysadmin@192.168.1.3
Last login: Tue Jul 23 13:56:23 2024 from 192.168.1.1
[sysadmin@CentOS-srv ~]$ exit
logout
Connection to 192.168.1.3 closed.
ansible@controlador-ansible:~$ ssh sysadmin@192.168.1.2
Welcome to Ubuntu 24.04 LTS (GNU/Linux 6.8.0-38-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

System information as of Tue Jul 23 05:56:34 PM UTC 2024

System load:  0.08          Processes:            125
Usage of /:   30.4% of 6.94GB Users logged in:          1
Memory usage: 11%          IPv4 address for enp0s3: 10.0.2.15
Swap usage:   0%

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

Last login: Tue Jul 23 17:55:43 2024 from 192.168.1.1
sysadmin@ubuntu-srv:~$
```

Creamos la clave publica/privada para acceder desde el controlador a los servidores sin contraseña.

```
Connection to 192.168.1.3 closed.
● ansible@controlador-ansible:~$ ssh-copy-id sysadmin@192.168.1.3
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/ansible/.ssh/id_ed25519.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 the new keys
sysadmin@192.168.1.3's password:

Number of key(s) added: 1

Now try logging into the machine, with: "ssh 'sysadmin@192.168.1.3'"
and check to make sure that only the key(s) you wanted were added.
● ansible@controlador-ansible:~$ ssh sysadmin@192.168.1.3
Last login: Fri Aug 2 14:35:55 2024 from 192.168.1.1
[sysadmin@CentOS-srv ~]$ exit
logout
Connection to 192.168.1.3 closed.
○ ansible@controlador-ansible:~$
```

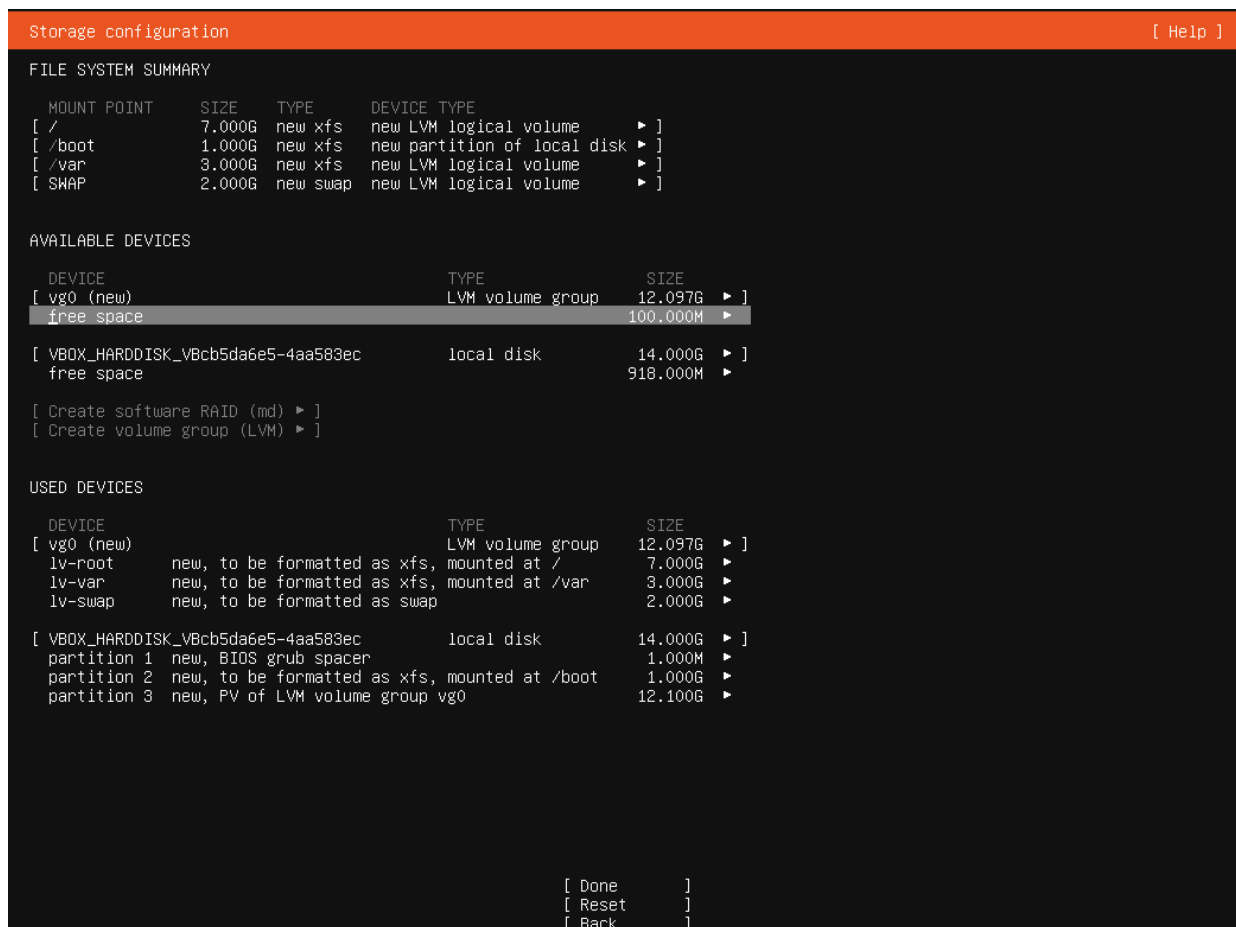
```
● ansible@controlador-ansible:~$ ssh-copy-id sysadmin@192.168.1.2
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/ansible/.ssh/id_ed25519.pub"
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed

/usr/bin/ssh-copy-id: WARNING: All keys were skipped because they already exist on the remote system.
(if you think this is a mistake, you may want to use -f option)
○ ansible@controlador-ansible:~$ ssh sysadmin@192.168.1.2
Last login: Fri Aug 2 18:18:53 2024 from 192.168.1.1
sysadmin@ubuntu-srv:~$
```

# PARTE B

## UBUNTU

En el proceso de instalación del servidor Ubuntu creamos una partición primaria para el kernel ya que este no es compatible con LVM y otra en la cual creamos un physical volume que contiene un volume group llamado “vg0” en el cual crearemos los logical volumes que contendrán las carpetas del sistema mencionadas en la letra.



Creamos el usuario no root sysadmin y nombramos a la máquina como Ubuntu-srv



## CentOS:

Creamos las particiones con lvm siguiente el mismo esquema que en Ubuntu.

MANUAL PARTITIONING CENTOS STREAM 9 INSTALLATION

Done us Help!

▼ New CentOS Stream 9 Installation

SYSTEM	
/	7 GiB
cs-root	
/var	3 GiB
cs-var	
/boot	1024 MiB
sda1	
<b>swap</b>	<b>2 GiB</b>
cs-swap	

+ - ↺

AVAILABLE SPACE: 1016 MiB TOTAL SPACE: 14 GiB

1 storage device selected

cs-swap

Mount Point:

Desired Capacity:

Device Type: LVM ☐ Encrypt

File System: swap ☒ Reformat

Device(s): ATA VBOX HARDISK (sda) Modify...

Volume Group: cs (4 MiB free) Modify...

Label:

Name: swap

Update Settings

*Note: The settings you make on this screen will not be applied until you click on the main menu's 'Begin Installation' button.*

Discard All Changes



## PARTE C

### **1) Descripción del Playbook1:**

- 1- Definir variables, usuario que le da el privilegio a las tareas y el grupo al que se le ejecutarán.
- 2- Las primeras dos tareas son para detectar que distribución Linux se usará y así poder actualizar el sistema.
- 3- De igual manera instalamos el JDK de Java según la distribución que se implementará.
- 4- Creamos un usuario llamado “tomcat” el cual no podrá loguearse ni se le creará una carpeta para dicho usuario. Este está creado con el fin de ser utilizado por el servicio tomcat
- 5- Chequeamos que no exista el archivo de instalación de tomcat en caso de que el servidor ya lo tenga, para así evitar procesos. En caso de que no lo tenga, procederá a descargarlo e instalarlo en la carpeta correspondiente, la cual se corroborará si existe y en caso negativo la creará y otorgará los permisos de propietario al usuario tomcat creado posteriormente, también modificando el esquema de permisos.
- 6- Copiamos el archivo de configuración tomcat.service el cual es el encargado de registrar el servicio en Systemd.
- 7- Copiamos la aplicación todo.war a la carpeta de aplicaciones web del servicio tomcat y también su archivo de configuración a la carpeta de donde lo obtiene dicha aplicación.
- 8- Luego, habilitamos e iniciamos el servicio.
- 9- Posteriormente creamos una regla de firewall para el puerto 8080 del tomcat.
- 10- Para finalizar, recargamos el firewall.

## Prueba de ejecución:

```
● ansible@controlador-ansible:~/obligatorio$ ansible-playbook -i ./inventory/hosts.yml --ask-become-pass ./playbooks/playbook1.yml
BECOME password:

PLAY [deployear java, tomcat y la aplicación] *****

TASK [Gathering Facts] *****
ok: [rh1]

TASK [Actualizar SO para distribución de familia Red Hat] *****
ok: [rh1]

TASK [Actualizar SO para distribución de familia Debian] *****
skipping: [rh1]

TASK [Instalar jdk de java en distribuciones Red Hat] *****
ok: [rh1]

TASK [instalar jdk de java en distribuciones Debian] *****
skipping: [rh1]

TASK [Crear usuario para Tomcat] *****
ok: [rh1]

TASK [Checkear que no exista el archivo de instalación de tomcat] *****
ok: [rh1]

TASK [Descargar paquete de instalación de Tomcat desde el repositorio original] *****
changed: [rh1]

TASK [Instalar comandos para extraer el archivo] *****
ok: [rh1]

TASK [Checkear que no exista la carpeta donde van los archivos de instalación] *****
ok: [rh1]

TASK [Crea la carpeta si no existe] *****
skipping: [rh1]

TASK [Descomprimir el paquete descargado] *****
changed: [rh1]

TASK [Cambiar permisos de la carpeta de Tomcat] *****
changed: [rh1]

TASK [Copiar archivo de configuracion plantilla tomcat.service para Systemd] *****
ok: [rh1]

TASK [Crear carpeta de tomcat que alojará el archivo de configuración de mi aplicación] *****
ok: [rh1]

TASK [Copiar archivo de configuración a la carpeta] *****
ok: [rh1]

TASK [Copiar app todo.war desde el controlador al servidor con tomcat] *****
ok: [rh1]

TASK [Habilitar e iniciar servicio Tomcat] *****
changed: [rh1]

TASK [Habilitar tráfico a través del puerto 8080 tomcat] *****
ok: [rh1]

TASK [Recargar el firewall] *****
changed: [rh1]

PLAY RECAP *****
rh1                : ok=17   changed=5   unreachable=0   failed=0   skipped=3   rescued=0   ignored=0

○ ansible@controlador-ansible:~/obligatorio$
```

## **2) Descripción del Playbook2:**

- 1- Definir variables, usuario que le da el privilegio a las tareas y el grupo al que se le ejecutarán.
- 2- Las primeras dos tareas son para detectar que distribución Linux se usará y así poder actualizar el sistema.
- 3- Instalamos el servicio MariaDB y lo aseguramos con el script `mysql_secure_installation`.
- 4- También instalamos los requisitos de este servicio, el cuál es python3-pymysql, este servicio se instala con el manejador de paquetes adecuado a la distribución.
- 5- Permitimos el acceso desde cualquier ip a la base de datos editando el archivo `50-server.cnf`
- 6- Creamos la base de datos `“todo”`
- 7- Creamos el usuario `“prueba”` al cual se le otorgaran todos los permisos a la base de datos.
- 8- Dentro de la base de datos hicimos las tablas `“users”` y `“todos”`, también otorgándole los permisos antes mencionados al usuario prueba que es el que utilizará la aplicación.
- 9- Habilitamos el tráfico ssh y el tráfico a través del puerto 3306, el cual usa MariaDB.

## Prueba de ejecución:

```
ansible@controlador-ansible:~/obligatorio$ ansible-playbook -i ./inventory/hosts.yml --ask-become-pass ./playbooks/playbook2.yml
BECOME password:

PLAY [instalar mysql y configurar base de datos] *****

TASK [Gathering Facts] *****
ok: [ub1]

TASK [Actualizar SO para distribución de familia Red Hat] *****
skipping: [ub1]

TASK [Actualizar SO para distribución de familia Debian] *****
ok: [ub1]

TASK [instalar mariadb service] *****
ok: [ub1]

TASK [Asegurar el servicio Mariadb con mysql_secure_installation] *****
changed: [ub1]

TASK [Instalar pymysql requerido para interacción con base de datos (Debian)] *****
ok: [ub1]

TASK [Instalar pymysql requerido para interacción con base de datos (Debian)] *****
skipping: [ub1]

TASK [Permitir acceso a la base de datos desde cualquier ip] *****
ok: [ub1]

TASK [Crear base de datos todo] *****
changed: [ub1]

TASK [Crear usuario prueba] *****
ok: [ub1]

TASK [Crear tablas de usuarios y tareas y luego dar permisos sobre la misma a prueba] *****
changed: [ub1]

TASK [Habilitar tráfico de SSH] *****
ok: [ub1]

TASK [Habilitar tráfico a través del puerto 3306 de MariaDB] *****
ok: [ub1]

PLAY RECAP *****
ub1                : ok=11  changed=3  unreachable=0  failed=0  skipped=2  rescued=0  ignored=0

ansible@controlador-ansible:~/obligatorio$
```

Prueba de aplicación en funcionamiento:

The screenshot shows a web browser window titled 'User Management Applic x' with the address bar displaying '192.168.1.3:8080/todo/list'. The page has an orange header with 'Todo App' and 'Logout' links. The main content area is titled 'List of Todos' and features a green 'Add Todo' button. Below the button is a table with the following data:

Title	Target Date	Todo Status	Actions
Sacarse 100 en el obligatorio	2024-08-04	false	<a href="#">Edit</a> <a href="#">Delete</a>

The footer of the page displays '© 2019 Copyright: Java Guides'. The browser window is running within an Oracle VM VirtualBox environment, as indicated by the title bar 'Controlador Ansible [Corriendo] - Oracle VM VirtualBox'.

## PARTE D

Creamos el repositorio en Github en el cual subimos todos los archivos de configuración como los playbooks que utilizamos, el directorio en el cual incluimos este documento y un archivo README.md el cual describe como implementar todos los archivos del repositorio.

