

# Obligatorio Taller de Linux

Matias Langlois – 281504

Victor Méndez – 235533

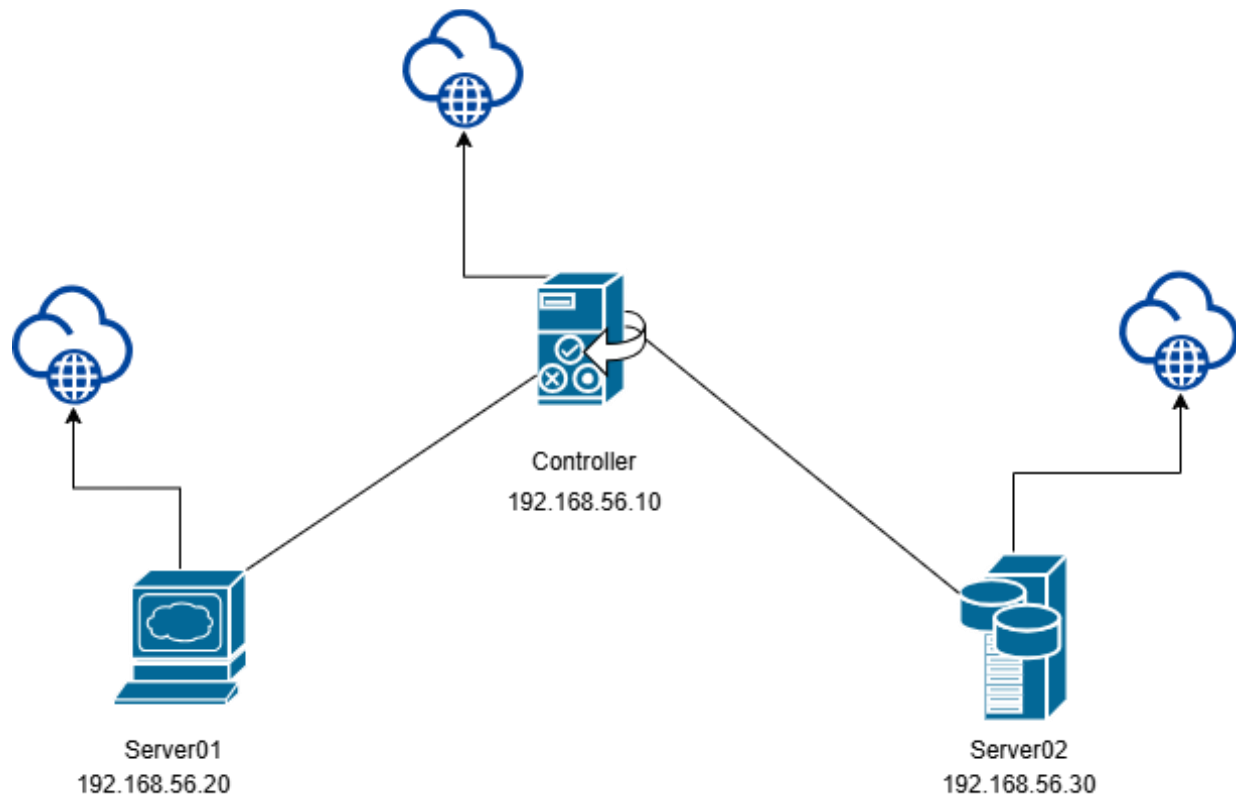
Docente: Enrique Verdes

## Índice

Servicios implementados .....	3
Instalación de servidores.....	4
Servidor Controller.....	4
Server1 .....	8
Server2 .....	12
Instalación Ansible.....	17
GitHub .....	18
Errores encontrados.....	19
Declaración de autoría .....	20
Bibliografía .....	21
SQL .....	21

## Servicios implementados

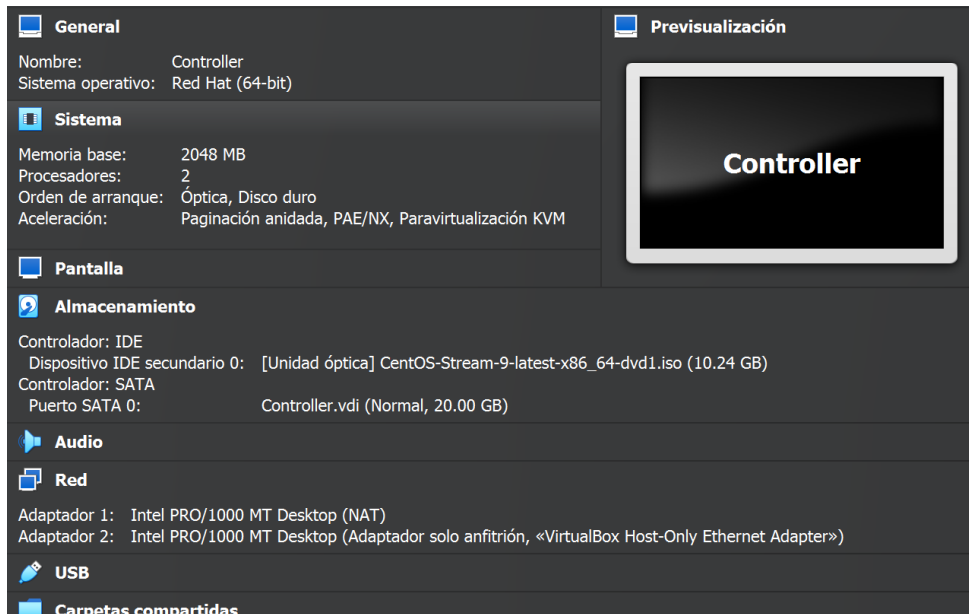
La solución propuesta para este caso incluye un servidor que funciona como controller para ejecuciones de ansible. Para satisfacer los requisitos establecidos, se desplegaron dos servidores asignados a distintos roles: Aplicación Web y uno para la base de datos de la aplicación web. Las cual contarán con dos interfaces de red una conectada a NAT y otra a Host-Only.



Hostname	IP	Función	Sistema Operativo
Controller	192.168.56.10	Controlador de ansible	CentOS
Server01	192.168.56.20	Aplicación Web	CentOS
Server02	192.168.56.30	Base de Datos	Ubuntu

# Instalación de servidores

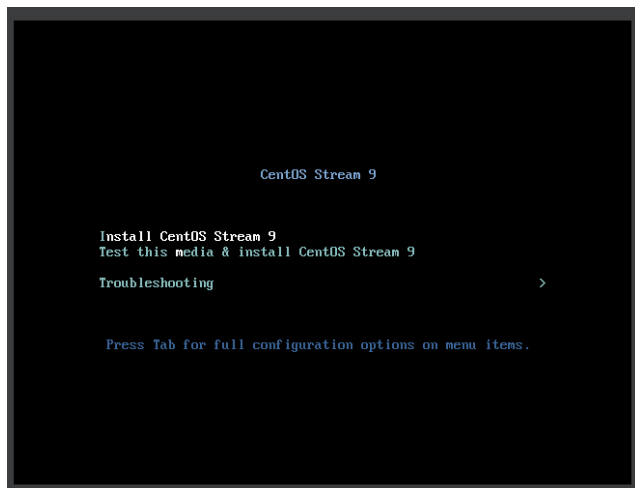
## Servidor Controller



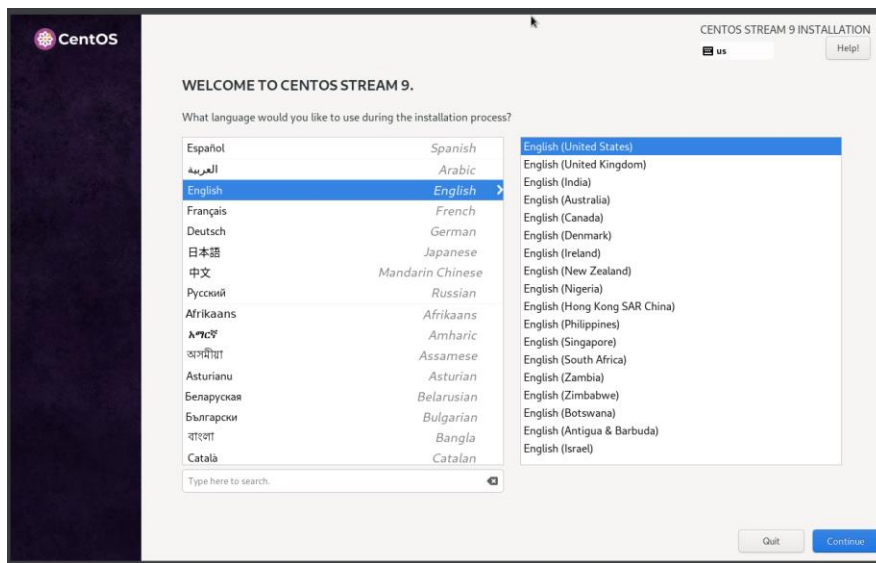
Se instala en el Controller el sistema operativo CentOS el cual será el encargado de la ejecución de los automatismos con las siguientes características:

- RAM: 2048 MB
- CPU: 2 Core
- Disco: 20GB
- Red:
  - NAT
  - Host-Only

Realizando los siguientes pasos:

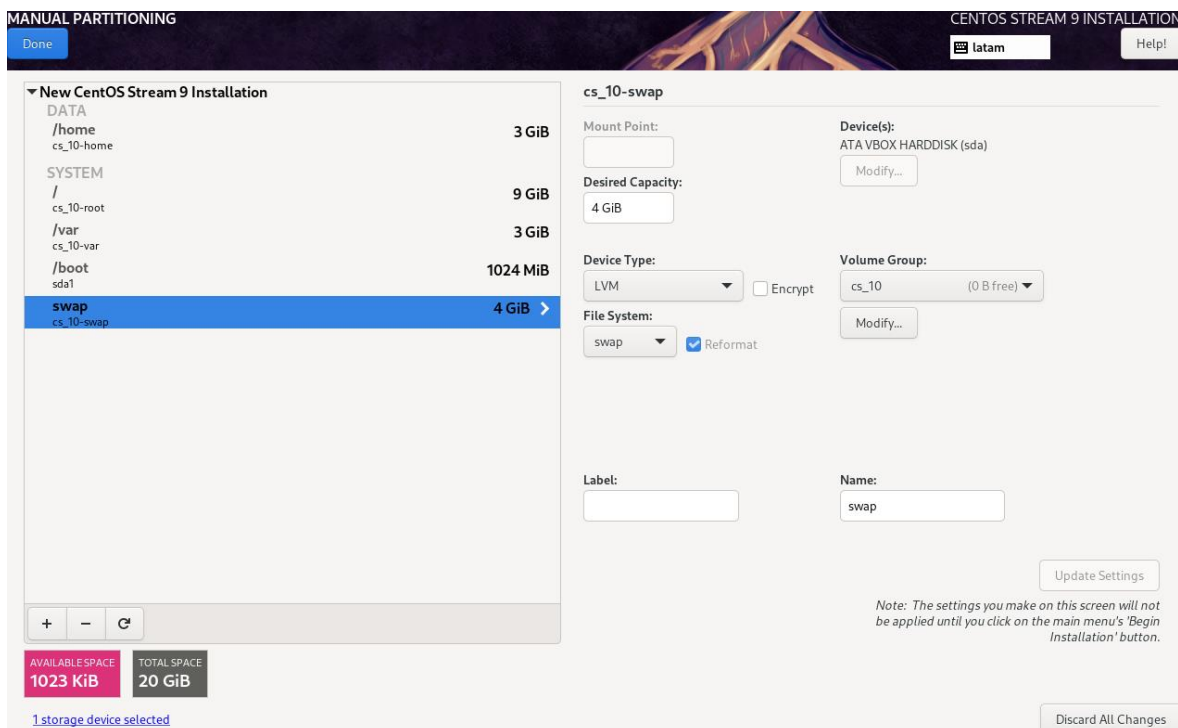


Se elige el idioma de la instalación:

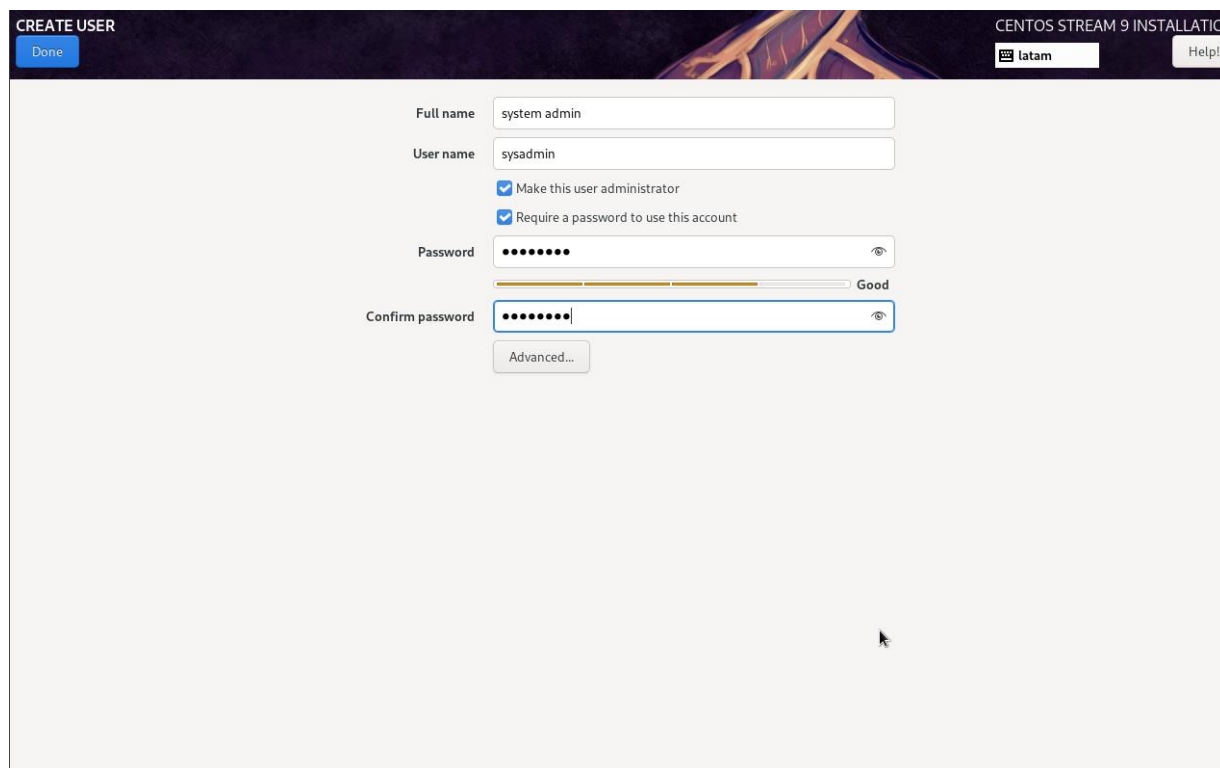


Se realiza la configuración de partición según el siguiente diseño:

- Partición de /boot – 1 GiB
- LVM:
  - / - 9 GiB
  - /home – 3 GiB
  - /var – 3 GiB
  - Swap – 4 GiB



Se establece el usuario de System admin (sysadmin) y su respectiva contraseña “tlxadmin” como el permiso de ejecutar comando como root (SUDO)



CREATE USER

Done

CENTOS STREAM 9 INSTALLATION

latam Help!

Full name system admin

User name sysadmin

☒ Make this user administrator

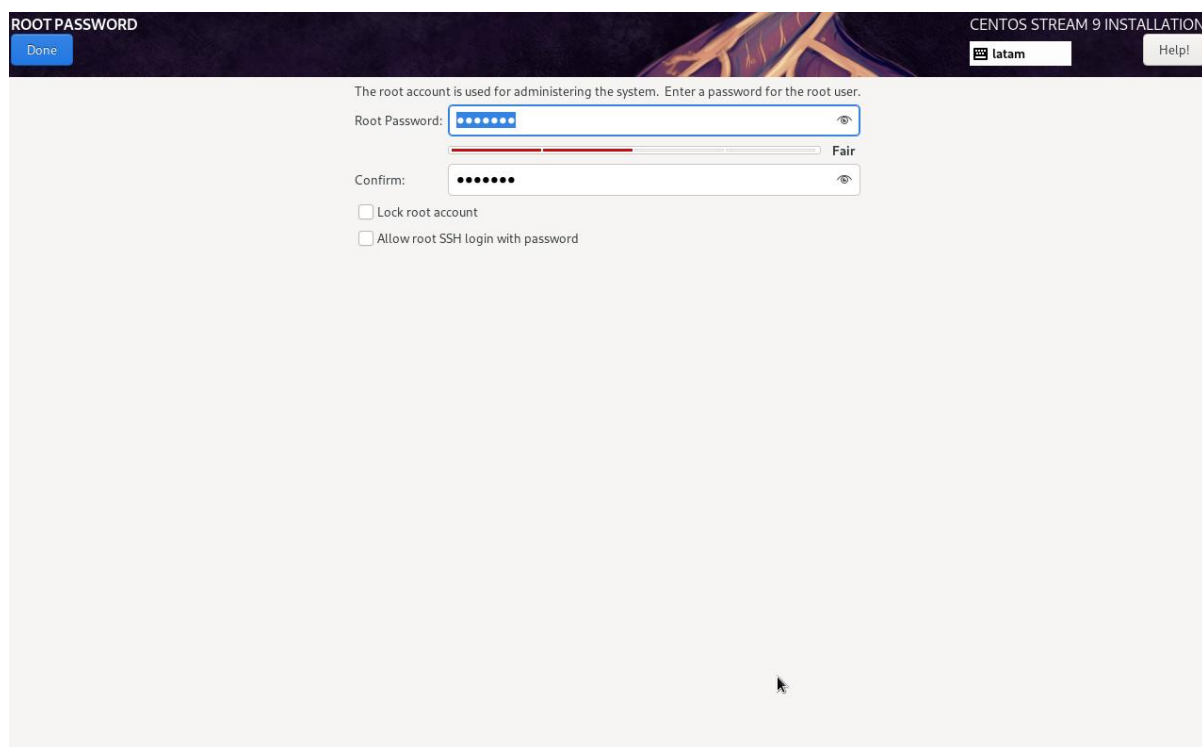
☒ Require a password to use this account

Password ●●●●●● Good

Confirm password ●●●●●●

Advanced...

Se establece las credenciales de root “tlxroot”



ROOT PASSWORD

Done

CENTOS STREAM 9 INSTALLATION

latam Help!

The root account is used for administering the system. Enter a password for the root user.

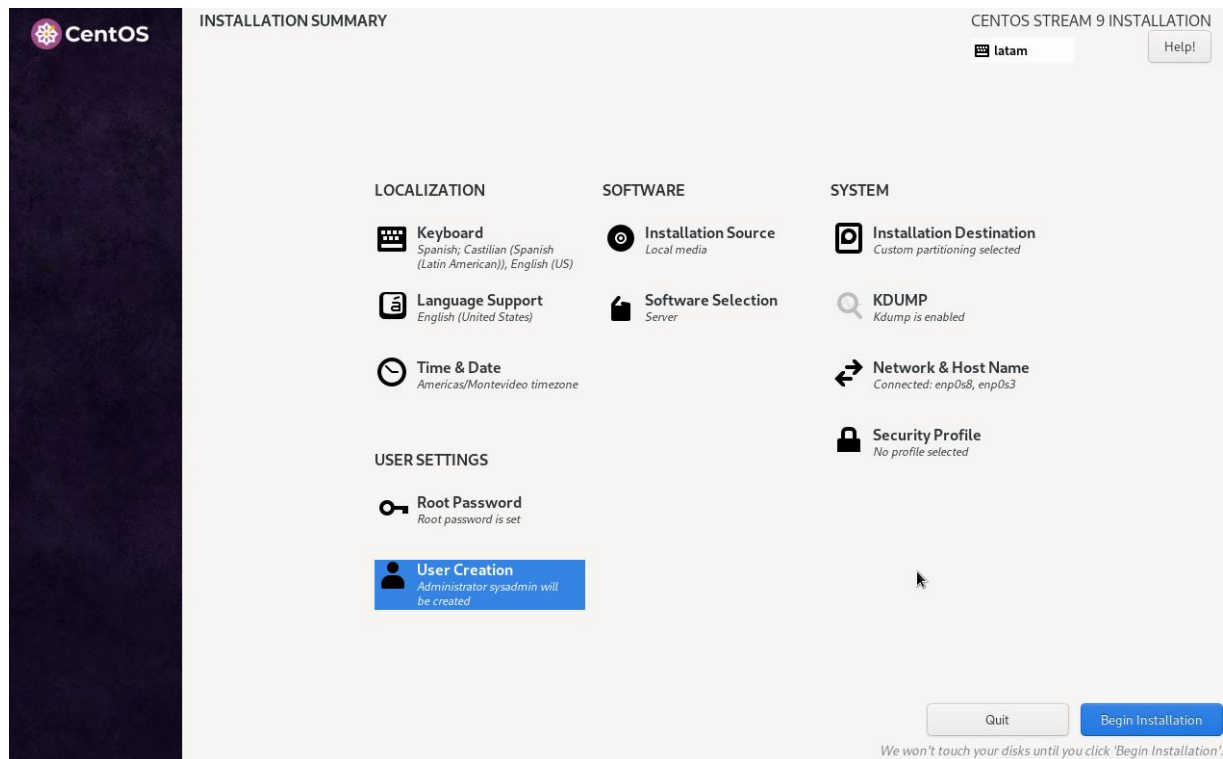
Root Password: ●●●●●● Fair

Confirm: ●●●●●●

☐ Lock root account

☐ Allow root SSH login with password

## Un breve resumen de instalación



Una vez finalizado la instalación se procede con la actualización, configuración de red y generación de clave de SSH.

Se realiza la actualización de sistema operativo

```
# dnf update -y
```

Se realiza la asignación de ip fija

```
# nmcli c mod enp0s8 ipv4.addresses 192.168.56.10/24 \
ipv4.method manual
```

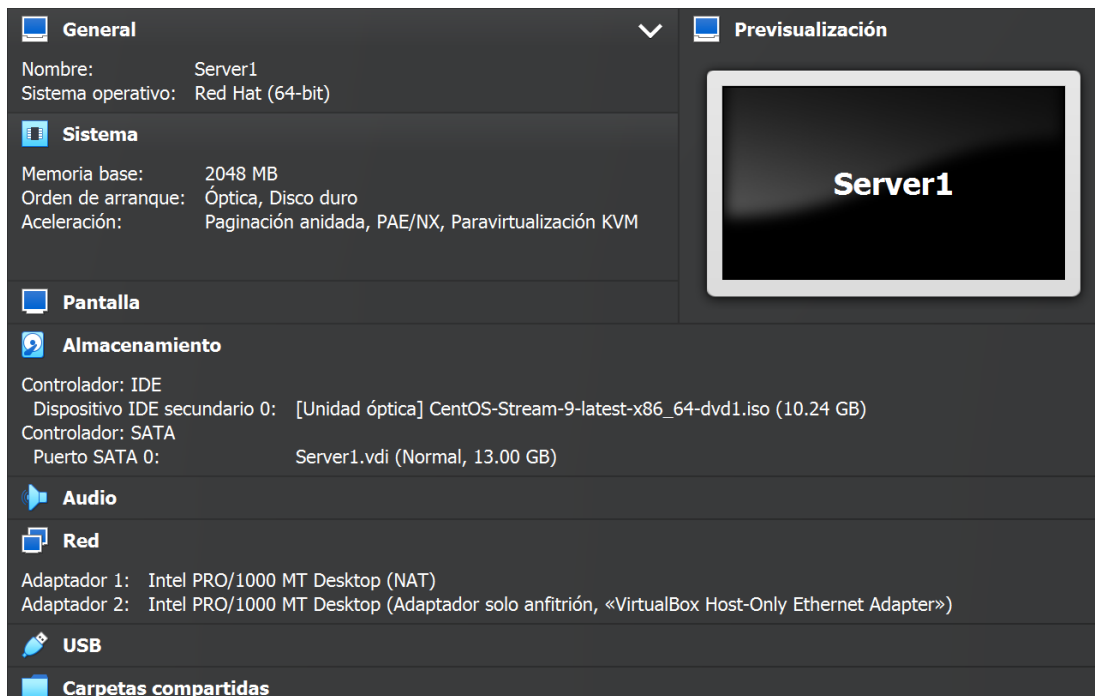
Se crea la llave publica /privada

```
$ ssh-keygen
```

Una vez creados los 3 servidores se copia la llave publica/privada al resto

```
$ ssh-copy-id 192.168.56.20
$ ssh-copy-id 192.168.56.30
```

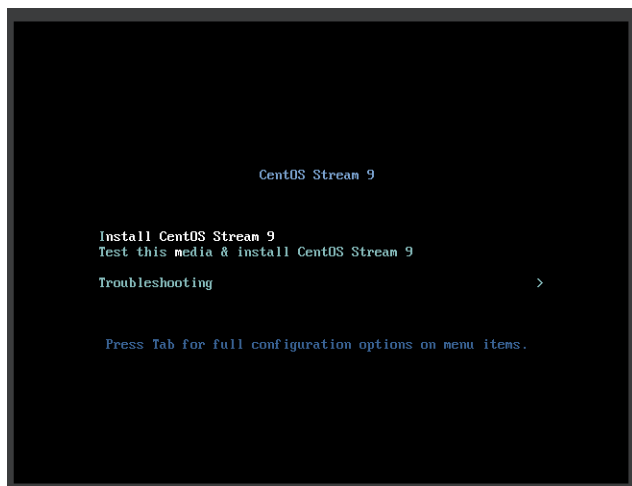
## Server1



Se instala en el Server1 el sistema operativo CentOS el cual será el encargado de alojar la aplicación web, con las siguientes características:

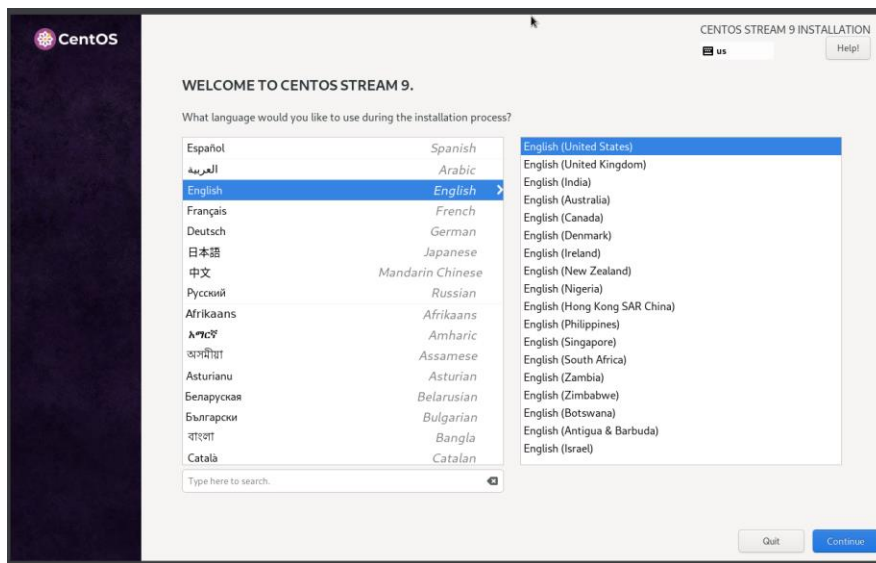
- RAM: 2048 MB
- CPU: 1 Core
- Disco: 13GB
- Red:
  - NAT
  - Host-Only

Realizando los siguientes pasos:



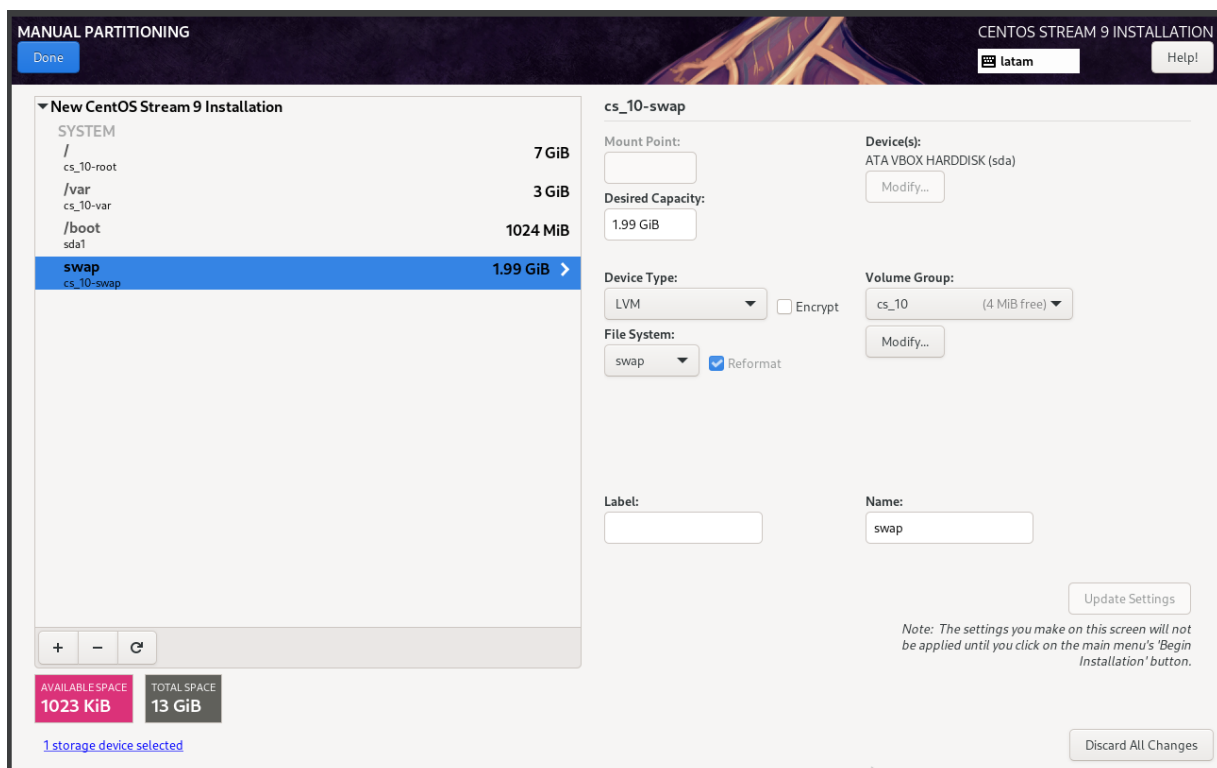


Se elige el idioma de la instalación:



Se realiza la configuración de partición según el siguiente diseño:

- Partición de /boot – 1GiB
- LVM:
  - / - 7 GiB
  - /var – 3 GiB
  - Swap – 2 GiB



Se establece el usuario de System admin (sysadmin) y su respectiva contraseña “tlxadmin” como el permiso de ejecutar comando como root (SUDO)

CREATE USER

Done

CENTOS STREAM 9 INSTALLATION

latam Help!

Full name system admin

User name sysadmin

☒ Make this user administrator

☒ Require a password to use this account

Password

Good

Confirm password

Advanced...

Se establece las credenciales de root “tlxroot”

ROOT PASSWORD

Done

CENTOS STREAM 9 INSTALLATION

latam Help!

The root account is used for administering the system. Enter a password for the root user.

Root Password:

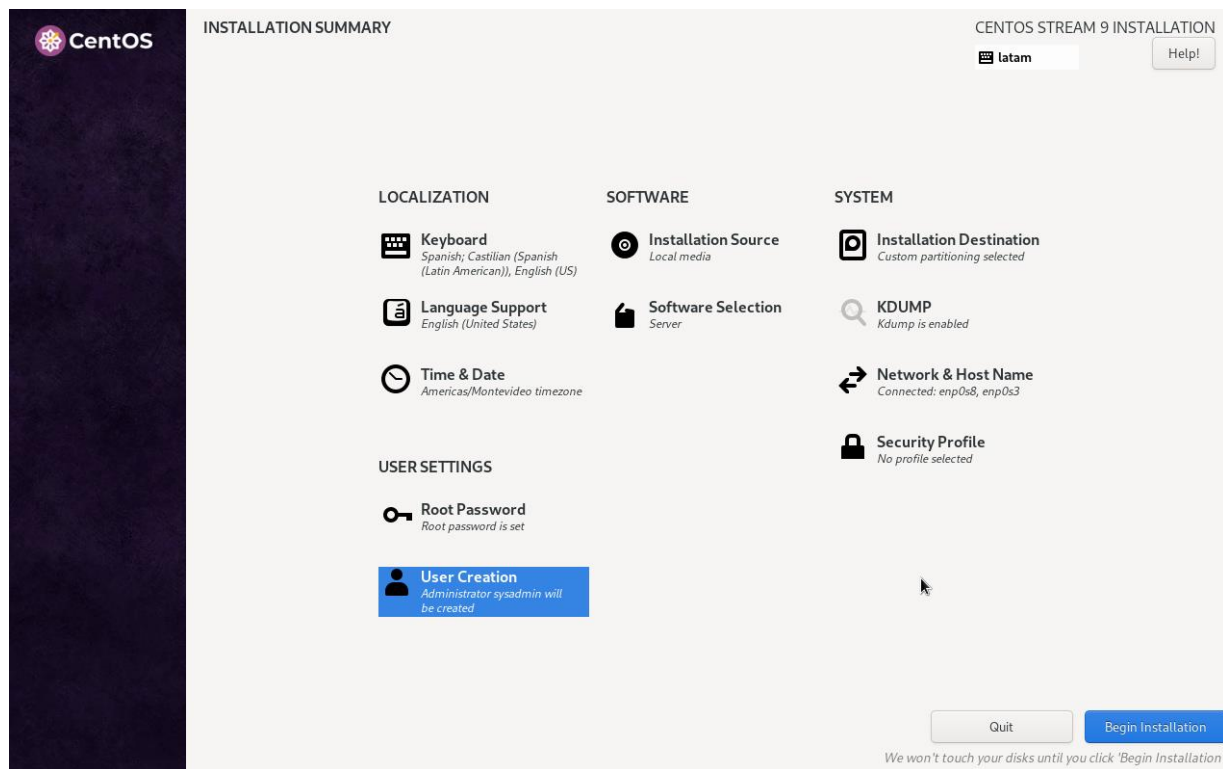
Fair

Confirm:

☐ Lock root account

☐ Allow root SSH login with password

## Un breve resumen de instalación



Una vez finalizado la instalación se procede con la actualización y configuración de red.

Se realiza la actualización de sistema operativo

```
# dnf update -y
```

Se realiza la asignación de ip fija

```
# nmcli c mod enp0s8 ipv4.addresses 192.168.56.20/24 \
ipv4.method manual
```

## Server2

The image shows the configuration window for a virtual machine named 'Server2'. The window is divided into two main sections: 'General' and 'Previsualización'. The 'General' section contains several tabs: 'General', 'Sistema', 'Pantalla', 'Almacenamiento', 'Audio', 'Red', and 'USB'. The 'General' tab is selected, showing the following settings: Name: Server2, Sistema operativo: Ubuntu (64-bit), Memoria base: 2048 MB, Orden de arranque: Disco duro, and Aceleración: Paginación anidada, Paravirtualización KVM. The 'Previsualización' section shows a preview of the virtual machine's display, which is currently black. The 'Almacenamiento' section shows the IDE controller with a secondary IDE device set to 'Vacío' and the SATA controller with a SATA device set to 'Server2.vdi (Normal, 13.00 GB)'. The 'Red' section shows two network adapters, both set to 'Intel PRO/1000 MT Desktop', with the first set to 'NAT' and the second set to 'VirtualBox Host-Only Ethernet Adapter'.

**General**

Nombre: Server2  
Sistema operativo: Ubuntu (64-bit)

**Sistema**

Memoria base: 2048 MB  
Orden de arranque: Disco duro  
Aceleración: Paginación anidada, Paravirtualización KVM

**Pantalla**

**Almacenamiento**

Controlador: IDE  
Dispositivo IDE secundario 0: [Unidad óptica] Vacío  
Controlador: SATA  
Puerto SATA 0: Server2.vdi (Normal, 13.00 GB)

**Audio**

**Red**

Adaptador 1: Intel PRO/1000 MT Desktop (NAT)  
Adaptador 2: Intel PRO/1000 MT Desktop (Adaptador solo anfitrión, «VirtualBox Host-Only Ethernet Adapter»)

**USB**

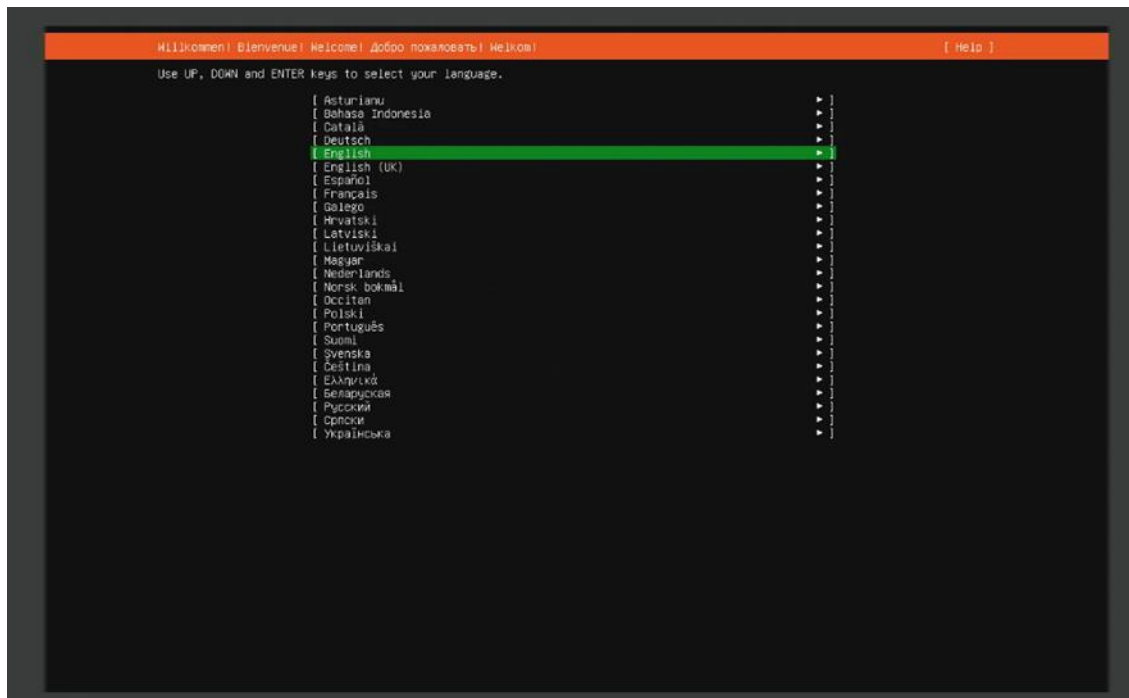
**Previsualización**

Se instala en el Server2 el sistema operativo Ubuntu el cual será el encargado de alojar la base de datos, con las siguientes características:

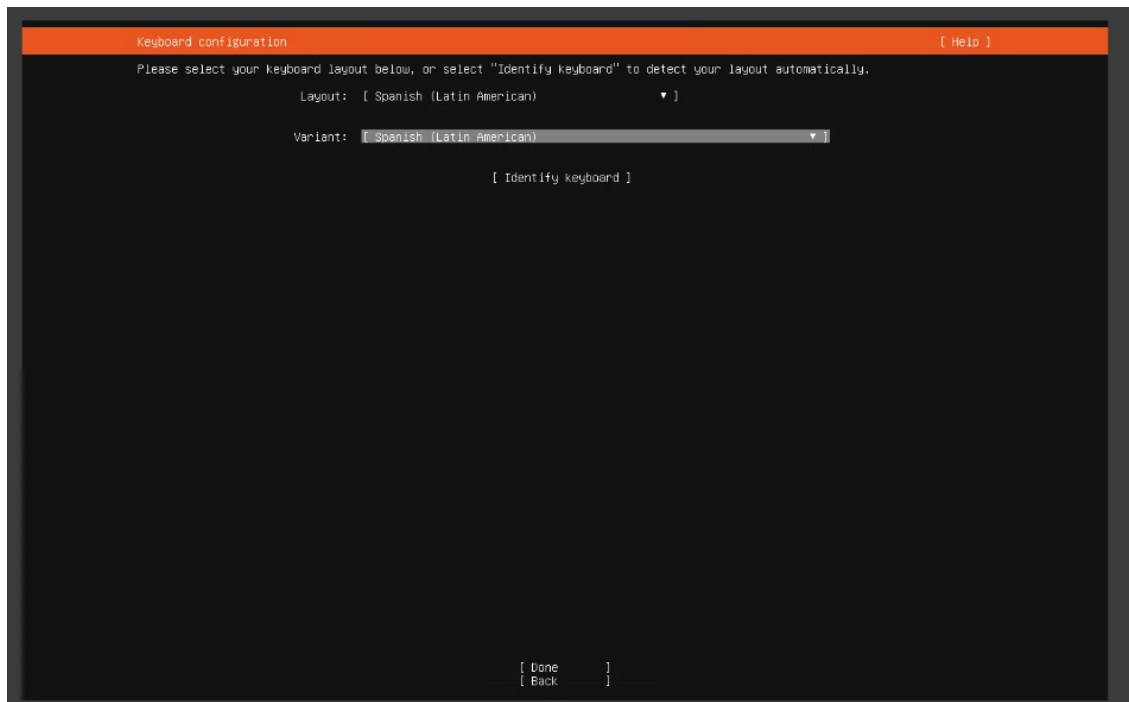
- RAM: 2048 MB
- CPU: 1 Core
- Disco: 13GB
- Red:
  - NAT
  - Host-Only

Realizando los siguientes pasos:

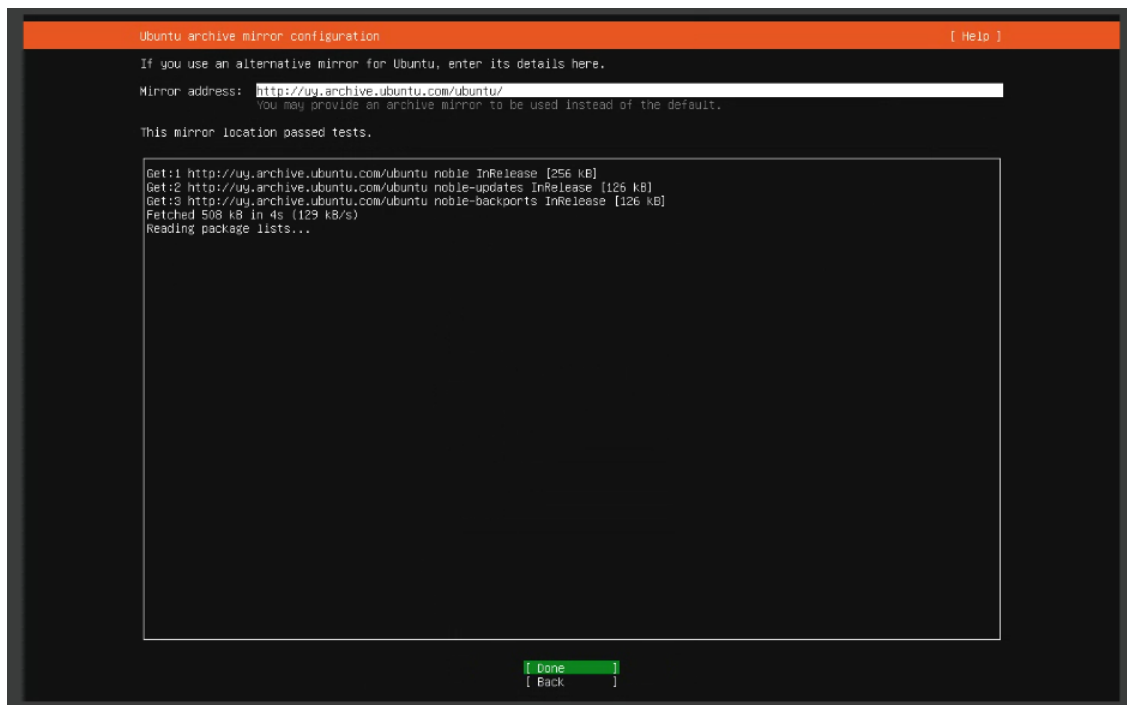
Se selecciona el idioma



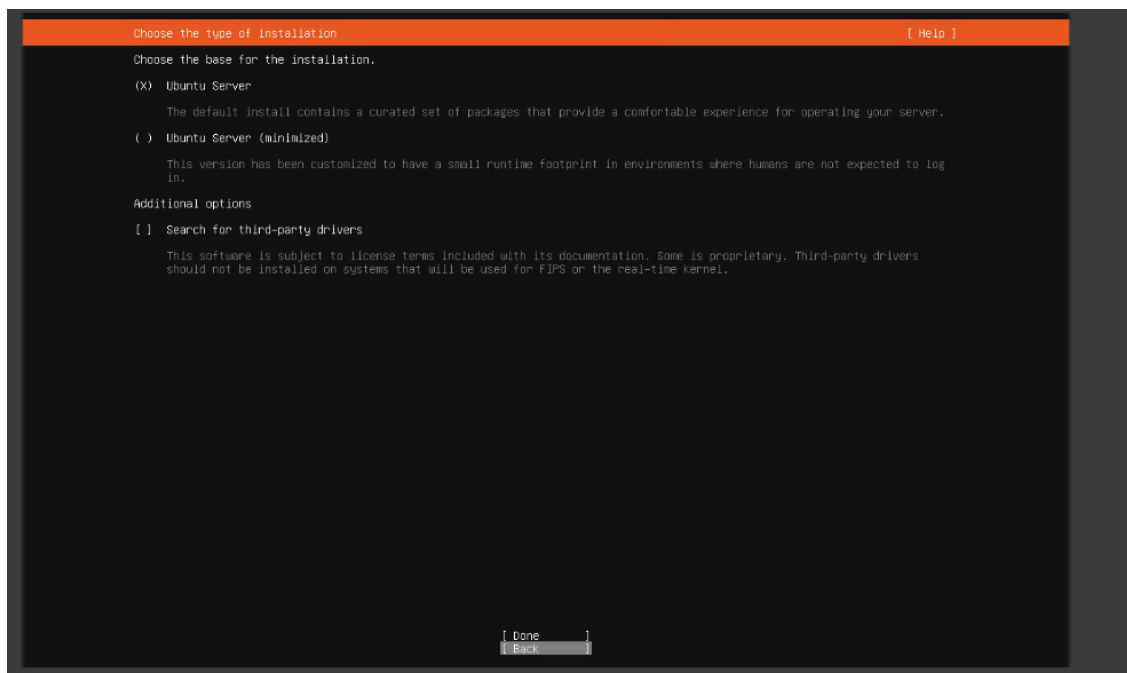
Se configura la distribución del teclado



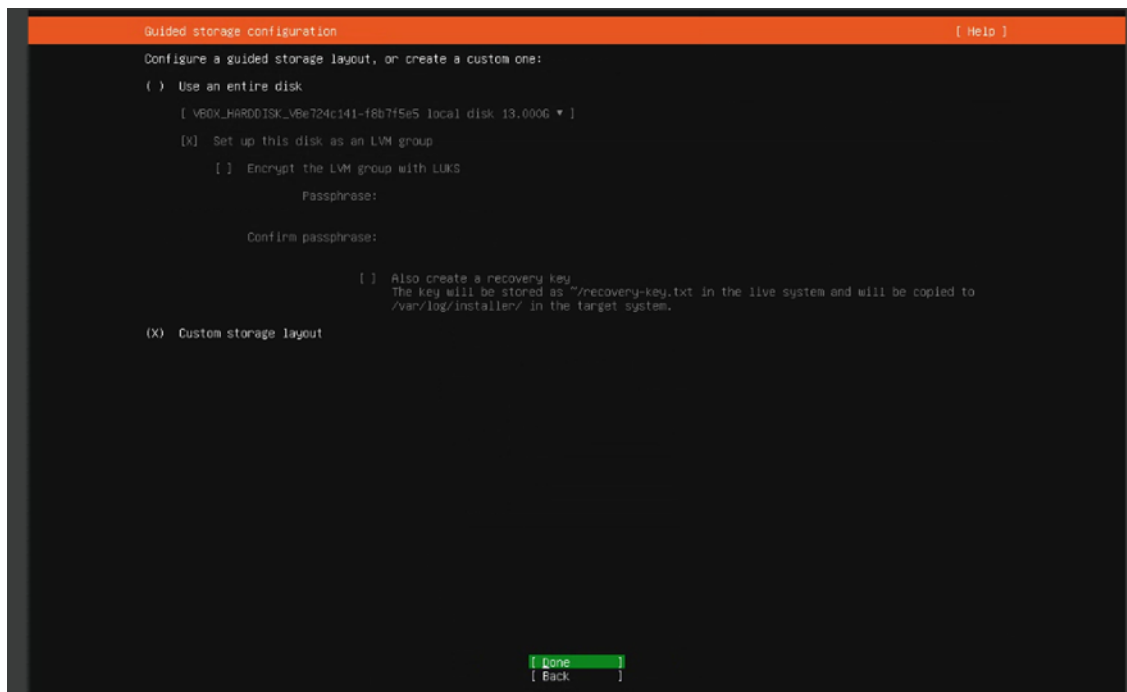
## Se configuran los repositorios de paquetes de Ubuntu



## Se configura el tipo de instalación, en este caso utilizaremos Ubuntu Server

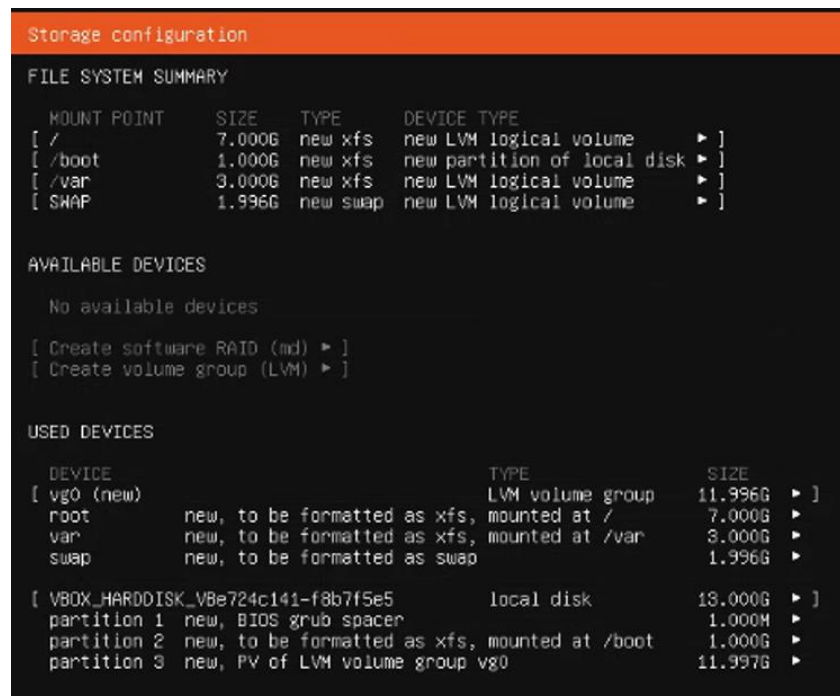


Se prepara el disco para configurarlo con LVM



Se realiza la configuración de partición según el siguiente diseño:

- Partición de /boot – 1GiB
- LVM:
  - / - 9 GiB
  - /home – 3GiB
  - /var – 3 GiB
  - Swap – 4 GiB



Se establece el usuario de System admin (sysadmin) y su respectiva contraseña "tlxadmin" como el permiso de ejecutar comando como root (SUDO)



Profile configuration [ Help ]

Enter the username and password you will use to log in to the system. You can configure SSH access on a later screen, but a password is still needed for sudo.

Your name:

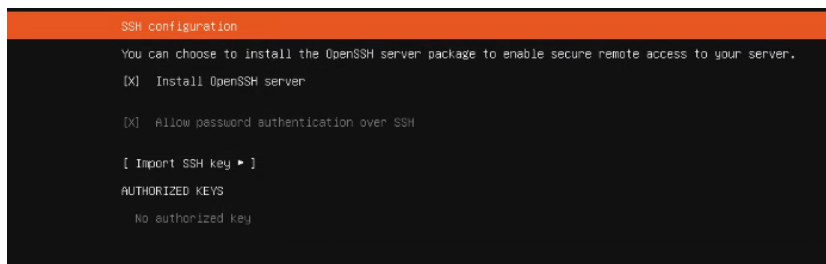
Your servers name:   
The name it uses when it talks to other computers.

Pick a username:

Choose a password:

Confirm your password:

Se instala SSH y con esto se finaliza la configuración aplicada para la instalación de Ubuntu



SSH configuration

You can choose to install the OpenSSH server package to enable secure remote access to your server.

☒ Install OpenSSH server

☒ allow password authentication over SSH

[ Import SSH key ▸ ]

AUTHORIZED KEYS

No authorized key

Una vez finalizado la instalación se procede con la actualización y configuración de red.

Se realiza la actualización de sistema operativo

```
# apt update && apt upgrade
```

Se realiza la asignación de ip fija

```
network:
  ethernets:
    enp0s3:
      dhcp4: true
    enp0s8:
      dhcp4: no
      dhcp6: no
      addresses: [192.168.56.30/24, ]
      gateway4: 192.168.56.1
  version: 2
```



## Instalación Ansible

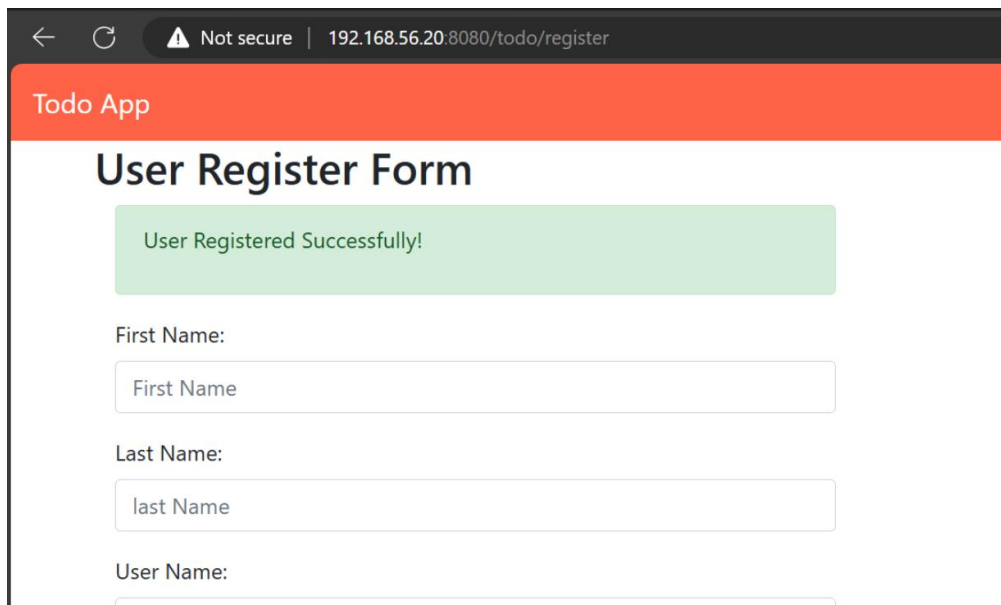
Se instala Ansible en el servidor Controller

```
# dnf update -y
# dnf clean all
# dnf install python3-pip
$ pip install pipx
$ pipx ensurepath
$ pipx install ansible-core
$ pipx inject ansible-core argcomplete
$ pipx inject ansible-core lint
$ activate-global-python-argcomplete --user
$ . /home/sysadmin/.bash_completion
```

Para la ejecución del playbooks completo

```
$ ansible-galaxy collection install -r collections/requirement.yml
$ ansible-playbook -i inventory/hosts playbooks/site.yml --ask-become-pass
```

Se valida funcionamiento del servicio Tomcat, ejecutando la aplicación web y su conexión a la base de datos.



← ↻ ⚠ Not secure | 192.168.56.20:8080/todo/register

Todo App

### User Register Form

User Registered Successfully!

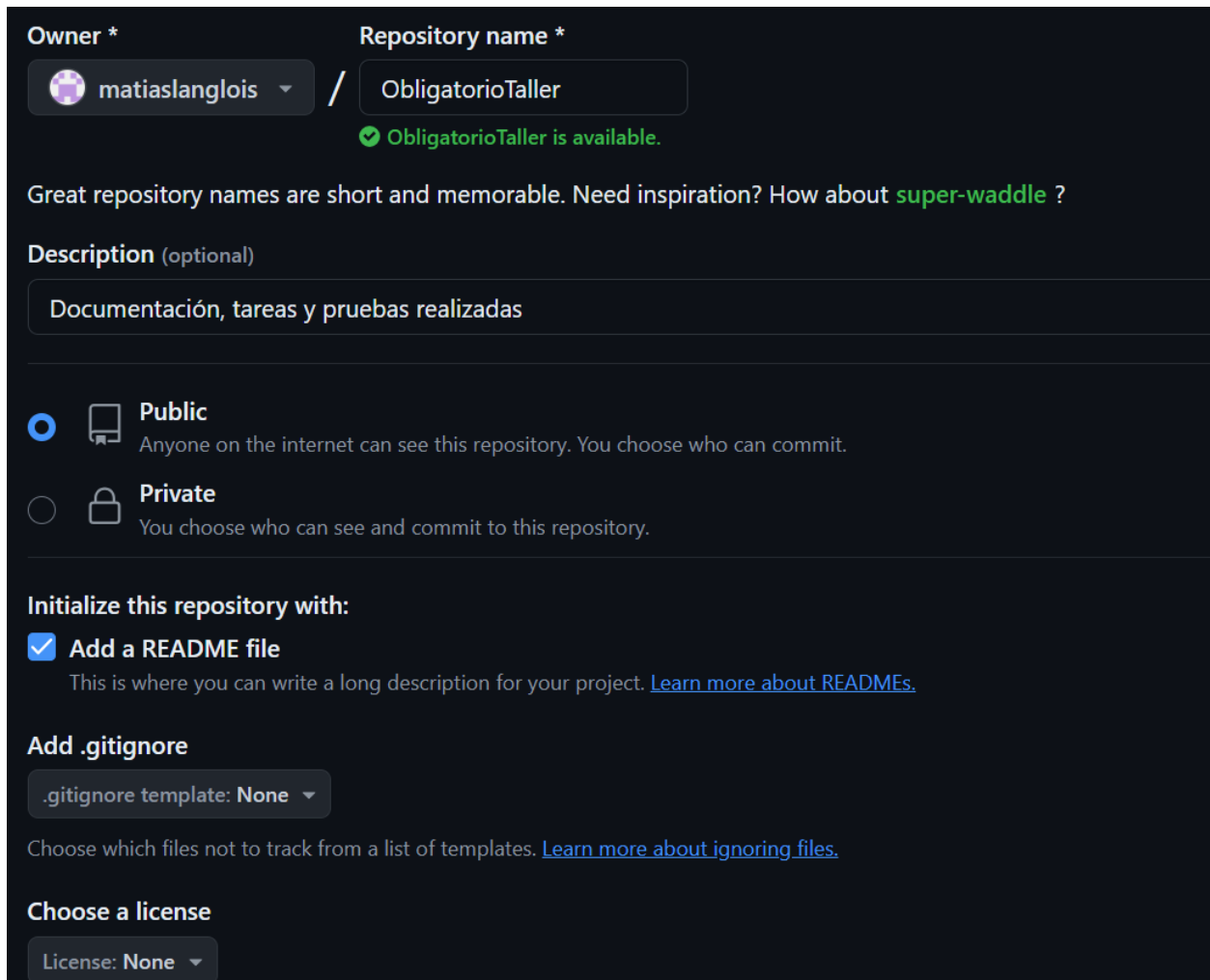
First Name:

Last Name:

User Name:

## GitHub

Se crea el repositorio de código en GitHub ObligatorioTaller



The screenshot shows the GitHub repository creation interface. At the top, the 'Owner' is set to 'matiaslanglois' and the 'Repository name' is 'ObligatorioTaller'. A green checkmark indicates that the repository name is available. Below this, a message suggests great repository names are short and memorable, with a link to 'super-waddle?'. The 'Description' field is optional and contains the text 'Documentación, tareas y pruebas realizadas'. The 'Public' option is selected, indicating that anyone on the internet can see the repository. The 'Private' option is also visible. Under 'Initialize this repository with:', the 'Add a README file' option is checked. Below this, there is a section for 'Add .gitignore' with a dropdown menu set to 'None'. At the bottom, there is a section for 'Choose a license' with a dropdown menu set to 'None'.

Owner \* / Repository name \*

matiaslanglois / ObligatorioTaller

✔ ObligatorioTaller is available.

Great repository names are short and memorable. Need inspiration? How about [super-waddle?](#)

Description (optional)

Documentación, tareas y pruebas realizadas

☒ Public  
Anyone on the internet can see this repository. You choose who can commit.

☐ Private  
You choose who can see and commit to this repository.

Initialize this repository with:

☒ Add a README file  
This is where you can write a long description for your project. [Learn more about READMEs.](#)

Add .gitignore

.gitignore template: None

Choose which files not to track from a list of templates. [Learn more about ignoring files.](#)

Choose a license

License: None

Desde el Controller se clona el git para realizar los playbooks de ansible

```
$ git clone git@github.com:matiaslanglois/ObligatorioTaller.git
```

Luego de finalizado el desarrollo se realizan los siguientes comandos para registrar los cambios y actualizar el repositorio:

```
$ git add  
$ git commit  
$ git push
```


## Errores encontrados

Se crearon las máquinas virtuales con los tres servidores solicitados, luego se inició el proceso de desarrollo de los automatismos. Se crearon los PlayBooks uno por uno hasta tener los solicitados para posteriormente comenzar las pruebas.

Primero se chequeo la sintaxis con `--syntax-check`, se detectaron errores de ortografía en la escritura de módulos y errores en la indentación.

Luego se probaron cada uno de los PlayBook para validar su correcto funcionamiento donde se detectaron varios errores:

- Error en módulos por ejemplo se escribió `ansible.builtin.firewalld` y era `ansible.posix.firewalld` el cual se resolvió utilizando la documentación de `Firewalld`
- Error de ruta de archivo incorrecta "src does not exist" simplemente estábamos usando una ruta corta y se reemplazó por la ruta completa por ejemplo `(/home/sysadmin/ObligatorioTaller/config/todo.war)`.
- Error por falta de state "Missing required arguments: state" al crear el directorio donde estará el archivo de configuración de la aplicación a la base de datos, faltaba el `state:directory`.
- Error "A MySQL module is required" el cual utilizando la documentación de MySQL (**`community.mysql.mysql_user module`**) donde se indica lo siguiente:



The screenshot shows the 'Ansible Community Documentation' website. On the left is a dark sidebar with navigation links: Playbook Keywords, Return Values, Ansible Configuration Settings, Controlling how Ansible behaves: precedence rules, YAML Syntax, Python 3 Support, Interpreter Discovery, Releases and maintenance, Testing Strategies, Sanity Tests, and Frequently Asked Questions. The main content area has a light blue background and displays two bullet points under the heading 'Requires the PyMySQL (Python 2.7 and Python 3.X) or MySQL-python (Python 2.X) package installed on the remote host. The Python package may be installed with apt-get install python-pymysql (Ubuntu; see ansible.builtin.apt) or yum install python2-PyMySQL (RHEL/CentOS/Fedora; see ansible.builtin.yum). You can also use dnf install python2-PyMySQL for newer versions of Fedora; see ansible.builtin.dnf.' The second bullet point states: 'Be sure you have mysqlclient, PyMySQL, or MySQLdb library installed on the target machine for the Python interpreter Ansible discovers. For example if ansible discovers and uses Python 3, you need to install the Python 3 version of PyMySQL or mysqlclient. If ansible discovers and uses Python 2, you need to install the Python 2 version of either PyMySQL or MySQL-python.'

Por lo que se optó por la integración de instalar el paquete (`python3-mysqldb`) luego de la instalación de MaríaDB y antes de configurar la contraseña root de MariaDB.

## Declaración de autoría

Matias Langlois y Victor Mendez con documento de identificación 5.137.494-1, 48306884 respectivamente y estudiantes de la carrera Analista en infraestructura, en relación con el Obligatorio de Taller de Servidores Linux presentado para su defensa y evaluación en el curso 2024, declaramos que se asume la originalidad de dicho trabajo, entendida en el sentido de que no se han usado fuentes sin citarlas debidamente.

## Bibliografía

[https://docs.ansible.com/ansible/latest/collections/ansible/builtin/get\\_url\\_module.html](https://docs.ansible.com/ansible/latest/collections/ansible/builtin/get_url_module.html)

[https://docs.ansible.com/ansible/latest/collections/ansible/builtin/unarchive\\_module.html](https://docs.ansible.com/ansible/latest/collections/ansible/builtin/unarchive_module.html)

[https://docs.ansible.com/ansible/latest/collections/ansible/builtin/file\\_module.html](https://docs.ansible.com/ansible/latest/collections/ansible/builtin/file_module.html)

[https://docs.ansible.com/ansible/latest/collections/ansible/builtin/copy\\_module.html#ansible-collections-ansible-builtin-copy-module](https://docs.ansible.com/ansible/latest/collections/ansible/builtin/copy_module.html#ansible-collections-ansible-builtin-copy-module)

<https://www.redeszone.net/tutoriales/servidores/administrar-servicios-linux-systemd/>

[https://docs.ansible.com/ansible/latest/collections/ansible/posix/firewalld\\_module.html](https://docs.ansible.com/ansible/latest/collections/ansible/posix/firewalld_module.html)

## SQL

[https://docs.ansible.com/ansible/latest/collections/community/mysql/mysql\\_user\\_module.html](https://docs.ansible.com/ansible/latest/collections/community/mysql/mysql_user_module.html)

[https://docs.ansible.com/ansible/latest/collections/community/mysql/mysql\\_query\\_module.html](https://docs.ansible.com/ansible/latest/collections/community/mysql/mysql_query_module.html)

[https://docs.ansible.com/ansible/latest/collections/community/mysql/mysql\\_db\\_module.html#ansible-collections-community-mysql-mysql-db-module](https://docs.ansible.com/ansible/latest/collections/community/mysql/mysql_db_module.html#ansible-collections-community-mysql-mysql-db-module)