

A1.P4 Taller Ansible



ANSIBLE



Sergio García Márquez

I.E.S San Sebastián

Índice

1. Introducción	3
2. Creación e instalación de Ansible	3
3. Creación de los nodos en AWS.....	6
4. Conexión SSH con las instancias EC2 de AWS (taller9)	10
5. Instalación de herramientas adicionales (taller 7)	13
6. Configuración del inventario de Ansible (taller 8).....	14
7. Módulos de Ansible (taller 10).....	15
8. Playbooks (Taller 11).....	17
9. Instalar pila LAMP con variables (Taller 12)	19
10. Despliegue aplicación web sencilla (Taller 12)	21

1. Introducción

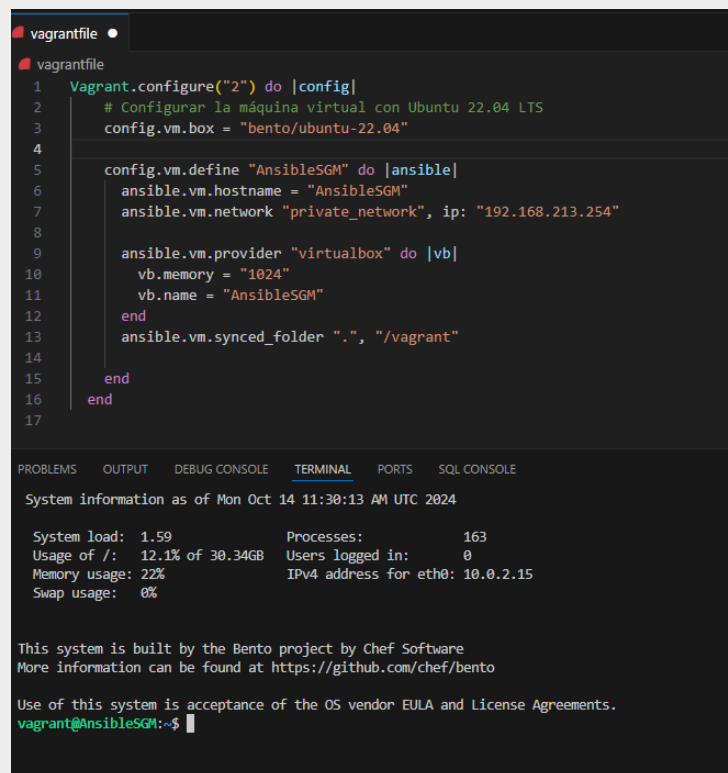
El escenario que vamos a utilizar en este tutorial estará formado por un nodo principal donde vamos a instalar Ansible y dos instancias EC2 de AWS que serán los nodos sobre los que vamos a realizar las tareas de configuración y administración.

El nodo principal no necesita ninguna característica especial, puede ser cualquier equipo. En este tutorial utilizaremos nuestro equipo de trabajo local con el sistema operativo Ubuntu.

Las instancias EC2 de AWS utilizarán la AMI con la última versión de Ubuntu Server disponible.

2. Creación e instalación de Ansible

Para el script de creación del Ansible, se usó el script de creación de la práctica anterior.



```
vagrantfile
1 Vagrant.configure("2") do |config|
2   # Configurar la máquina virtual con Ubuntu 22.04 LTS
3   config.vm.box = "bento/ubuntu-22.04"
4
5   config.vm.define "AnsibleSGM" do |ansible|
6     ansible.vm.hostname = "AnsibleSGM"
7     ansible.vm.network "private_network", ip: "192.168.213.254"
8
9     ansible.vm.provider "virtualbox" do |vb|
10      vb.memory = "1024"
11      vb.name = "AnsibleSGM"
12    end
13    ansible.vm.synced_folder ".", "/vagrant"
14  end
15 end
16
17
```

PROBLEMS OUTPUT DEBUG CONSOLE **TERMINAL** PORTS SQL CONSOLE

System information as of Mon Oct 14 11:30:13 AM UTC 2024

System load: 1.59	Processes: 163
Usage of /: 12.1% of 30.34GB	Users logged in: 0
Memory usage: 22%	IPv4 address for eth0: 10.0.2.15
Swap usage: 0%	

This system is built by the Bento project by Chef Software
More information can be found at <https://github.com/chef/bento>

Use of this system is acceptance of the OS vendor EULA and License Agreements.
vagrant@AnsibleSGM:~\$

Una vez tenemos lanzada la máquina, entraremos y procedemos a instalar ansible. Se hizo con los comandos que encontramos a continuación.

```
generating gpg configuration file /etc/gnupg/gpg.conf
vagrant@AnsibleSGM:~$ sudo apt install software-properties-common
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
software-properties-common is already the newest version (0.99.22.9).
0 upgraded, 0 newly installed, 0 to remove and 2 not upgraded.
vagrant@AnsibleSGM:~$ sudo add-apt-repository --yes --update ppa:ansible/ansible
Repository: 'deb https://ppa.launchpadcontent.net/ansible/ansible/ubuntu/ jammy main'
Description:
Ansible is a radically simple IT automation platform that makes your applications and systems easier to deploy. Avoid writing scripts
or custom code to deploy and update your applications- automate in a language that approaches plain English, using SSH, with no agents
to install on remote systems.

http://ansible.com/

If you face any issues while installing Ansible PPA, file an issue here:
https://github.com/ansible-community/ppa/issues
More info: https://launchpad.net/~ansible/+archive/ubuntu/ansible
Adding repository.
Adding deb entry to /etc/apt/sources.list.d/ansible-ubuntu-ansible-jammy.list
Adding disabled deb-src entry to /etc/apt/sources.list.d/ansible-ubuntu-ansible-jammy.list
Adding key to /etc/apt/trusted.gpg.d/ansible-ubuntu-ansible.gpg with fingerprint 6125E2A8C77F2818FB7BD15B93C4A3FD7BB9C367
Hit:1 http://security.ubuntu.com/ubuntu jammy-security InRelease
Hit:2 http://us.archive.ubuntu.com/ubuntu jammy InRelease
Hit:3 http://us.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:4 http://us.archive.ubuntu.com/ubuntu jammy-backports InRelease
```

```
vagrant@AnsibleSGM:~$ sudo add-apt-repository --yes --update ppa:ansible/ansible
Repository: 'deb https://ppa.launchpadcontent.net/ansible/ansible/ubuntu/ jammy main'
Description:
Ansible is a radically simple IT automation platform that makes your applications and systems easier to deploy. Avoid writing scri
or custom code to deploy and update your applications- automate in a language that approaches plain English, using SSH, with no ag
to install on remote systems.

http://ansible.com/

If you face any issues while installing Ansible PPA, file an issue here:
https://github.com/ansible-community/ppa/issues
More info: https://launchpad.net/~ansible/+archive/ubuntu/ansible
Adding repository.
Found existing deb entry in /etc/apt/sources.list.d/ansible-ubuntu-ansible-jammy.list
Adding deb entry to /etc/apt/sources.list.d/ansible-ubuntu-ansible-jammy.list
Found existing deb-src entry in /etc/apt/sources.list.d/ansible-ubuntu-ansible-jammy.list
Adding disabled deb-src entry to /etc/apt/sources.list.d/ansible-ubuntu-ansible-jammy.list
Adding key to /etc/apt/trusted.gpg.d/ansible-ubuntu-ansible.gpg with fingerprint 6125E2A8C77F2818FB7BD15B93C4A3FD7BB9C367
Hit:1 http://us.archive.ubuntu.com/ubuntu jammy InRelease
Hit:2 http://security.ubuntu.com/ubuntu jammy-security InRelease
Hit:3 http://us.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:4 http://us.archive.ubuntu.com/ubuntu jammy-backports InRelease
Hit:5 https://ppa.launchpadcontent.net/ansible/ansible/ubuntu jammy InRelease
Reading package lists... Done
```

```
Reading package lists... Done
vagrant@AnsibleSGM:~$ sudo apt install ansible -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  ansible-core python3-jmespath python3-kerberos python3-nacl python3-ntlm
  python3-requests-kerberos python3-requests-ntlm python3-resolvelib py
Suggested packages:
  python-nacl-doc python3-gssapi python3-invoke
The following NEW packages will be installed:
  ansible ansible-core python3-jmespath python3-kerberos python3-nacl py
```

Con esos comandos deberíamos tener ansible listo. También podremos comprobar que lo tenemos instalados mediante el comando ‘ansible –version’.

```
no VM guests are running outdated hypervisor (qemu) binaries on this host.
vagrant@AnsibleSGM:~$ ansible --version
ansible [core 2.17.5]
  config file = /etc/ansible/ansible.cfg
  configured module search path = ['/home/vagrant/.ansible/plugins/modules', '/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python3/dist-packages/ansible
  ansible collection location = /home/vagrant/.ansible/collections:/usr/share/ansible/collections
  executable location = /usr/bin/ansible
  python version = 3.10.12 (main, Sep 11 2024, 15:47:36) [GCC 11.4.0] (/usr/bin/python3)
  jinja version = 3.0.3
  libyaml = True
```

3. Creación de los nodos en AWS

El siguiente paso de la práctica será crear los nodos para Ansible. La versión de Ubuntu es la misma que la del Ansible.

Nombre y etiquetas [Información](#)

Nombre

Nodo1SGM

[Agregar etiquetas adicionales](#)

▼ Imágenes de aplicaciones y sistemas operativos (Imagen de máquina de Amazon) [Información](#)

Una AMI es una plantilla que contiene la configuración de software (sistema operativo, servidor de aplicaciones y aplicaciones) necesaria para lanzar la instancia. Busque o examine las AMI si no ve lo que busca a continuación.

Q

Busque en nuestro catálogo completo que incluye miles de imágenes de sistemas operativos y aplicaciones

Inicio rápido

Amazon Linux

aws

macOS

Mac

Ubuntu

ubuntu®

Windows

Microsoft

Red Hat

Red Hat

SUSE Linux

SUSE

Q

Buscar más AMI

Inclusión de AMI de AWS, Marketplace y la comunidad

Imágenes de máquina de Amazon (AMI)

Ubuntu Server 22.04 LTS (HVM), SSD Volume Type

Apto para la capa gratuita ▼

ami-005fc0f236362e99f (64 bits (x86)) / ami-07ee04759daf109de (64 bits (Arm))

Virtualización: hvm Activado para ENA: true Tipo de dispositivo raíz: ebs

Descripción

Ubuntu Server 22.04 LTS (HVM),EBS General Purpose (SSD) Volume Type. Support available from Canonical

Procedemos a asignarle el par de claves de ‘vockey’, así como crearemos un grupo de seguridad que permita el acceso desde ssh. Con esas indicaciones ya podríamos lanzar la instancia.

Una vez tenemos lanzado el primer nodo, toca el segundo.

Nombre y etiquetas Información

Nombre

[Agregar etiquetas adicionales](#)

▼ Imágenes de aplicaciones y sistemas operativos (Imagen de máquina de Amazon)
 Información

Una AMI es una plantilla que contiene la configuración de software (sistema operativo, servidor de aplicaciones y aplicaciones) necesaria para lanzar la instancia. Busque o examine las AMI si no ve lo que busca a continuación.

Recientes Inicio rápido

Amazon Linux
aws

macOS
Mac

Ubuntu
ubuntu

Windows
Microsoft

Red Hat
Red Hat

SUSE Li
SUS

Buscar más AMI
Inclusión de AMI de AWS, Marketplace y la comunidad

Imágenes de máquina de Amazon (AMI)

Ubuntu Server 22.04 LTS (HVM), SSD Volume Type
 ami-005fc0f236362e99f (64 bits (x86)) / ami-07ee04759daf109de (64 bits (Arm))
 Virtualización: hvm Activado para ENA: true Tipo de dispositivo raíz: ebs

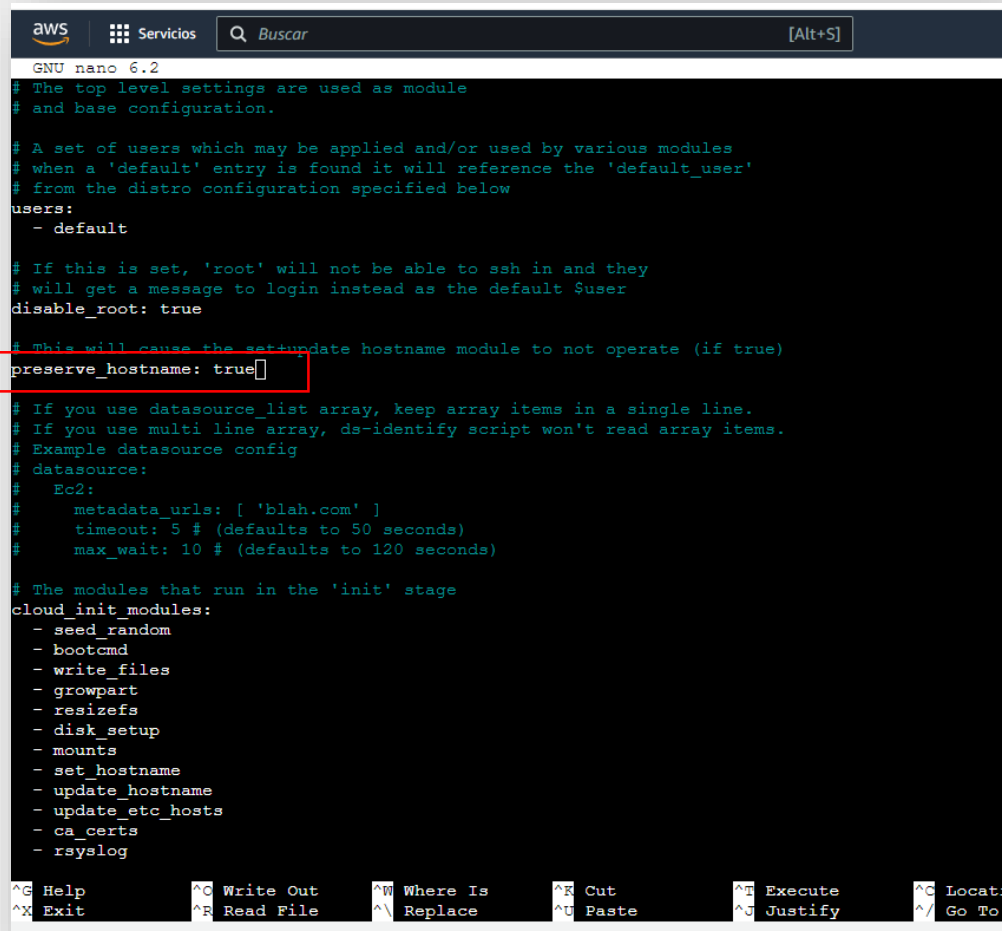
Descripción
 Ubuntu Server 22.04 LTS (HVM),EBS General Purpose (SSD) Volume Type. Support available from Canonical (<http://www.ubuntu.com/cloud/services>).

Importante en la configuración de red seleccionar el mismo grupo de seguridad que hemos creado junto al primer nodo. Toda la configuración es la misma al primer nodo. Ya podemos lanzarlo y tendremos ambos nodos creados.

Instancias (2) Información								
Buscar instancia por atributo o etiqueta (case-sensitive)								
<input type="checkbox"/>	Name	ID de la instancia	Estado de la i...	Tipo de inst...	Comprobación de	Estado de la alarma	Zona de dispon...	DNS de IPv4 pública
<input type="checkbox"/>	Nodo1SGM	i-05740401555f94080	En ejecución	t2.micro	2/2 comprobador	Ver alarmas	us-east-1b	ec2-3-85-88-165.comp...
<input type="checkbox"/>	Nodo2SGM	i-0674d5abaa84a725b	En ejecución	t2.micro	2/2 comprobador	Ver alarmas	us-east-1b	ec2-3-91-37-163.comp...

Para una mejor comprensión a la hora de ver la práctica, cambiaremos el nombre de host de ambos nodos.

Lo primero será hacer que el host no cambie nunca de nombre, para ello tendremos que modificar el archivo `/etc/cloud.cfg`. Lo que debemos hacer es cambiar la opción de `“preserve_hostname”` a `“true”`. Esto hará que no cambie.



```
aws Servicios [Alt+S]
GNU nano 6.2
# The top level settings are used as module
# and base configuration.

# A set of users which may be applied and/or used by various modules
# when a 'default' entry is found it will reference the 'default_user'
# from the distro configuration specified below
users:
- default

# If this is set, 'root' will not be able to ssh in and they
# will get a message to login instead as the default $user
disable_root: true

# This will cause the setupupdate hostname module to not operate (if true)
preserve_hostname: true

# If you use datasource_list array, keep array items in a single line.
# If you use multi line array, ds-identify script won't read array items.
# Example datasource config
# datasource:
#   Ec2:
#     metadata_urls: [ 'blah.com' ]
#     timeout: 5 # (defaults to 50 seconds)
#     max_wait: 10 # (defaults to 120 seconds)

# The modules that run in the 'init' stage
cloud_init_modules:
- seed_random
- bootcmd
- write_files
- growpart
- resizefs
- disk_setup
- mounts
- set_hostname
- update_hostname
- update_etc_hosts
- ca_certs
- rsyslog

^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Locat.
^X Exit      ^R Read File  ^_ Replace    ^U Paste      ^J Justify    ^/_ Go To
```

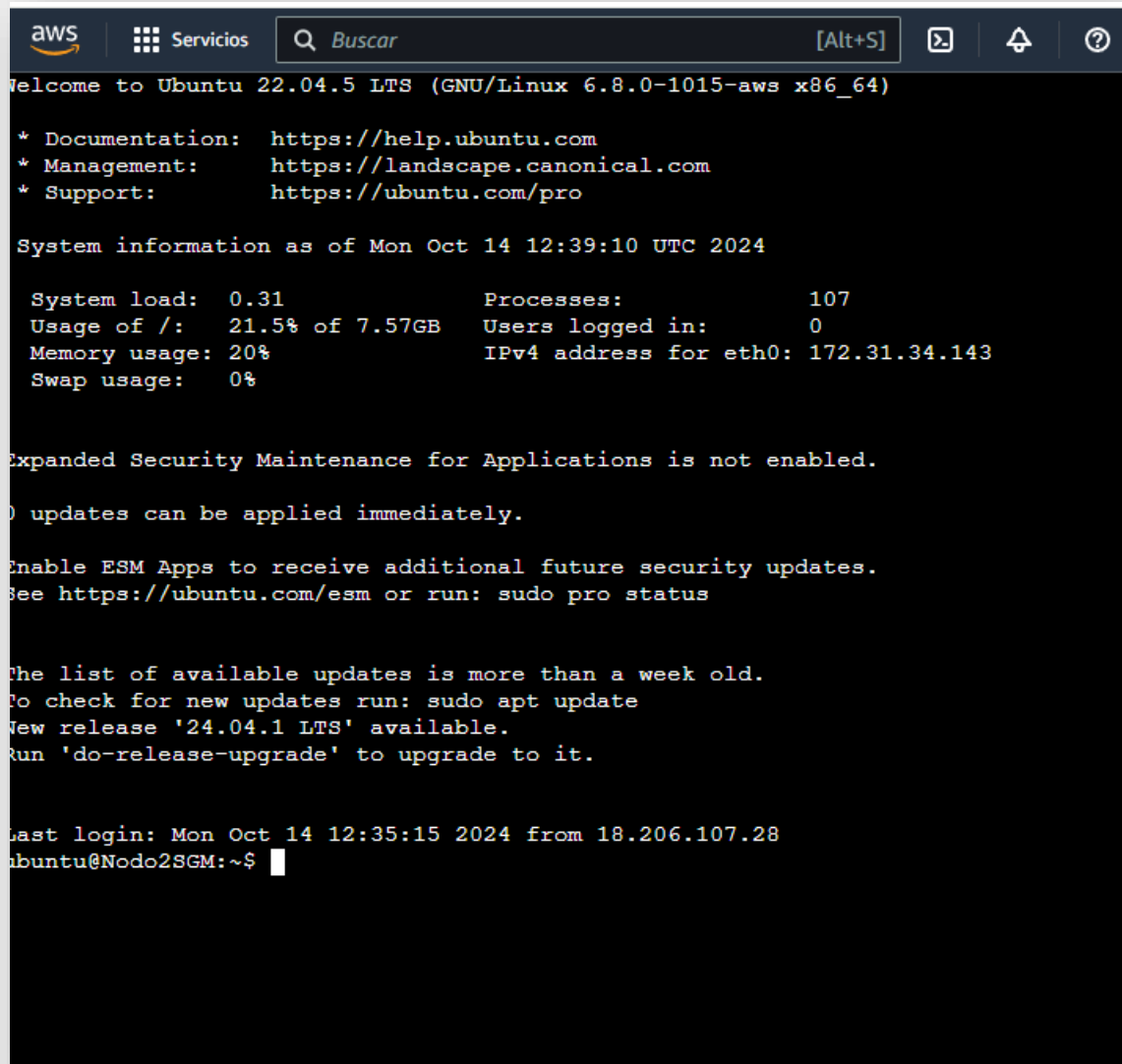
Listo, una vez hecho eso, haremos el comando que aparece a continuación.



```
Hardware Model: hvm dom0
ubuntu@ip-172-31-38-129:~$ sudo hostnamectl set-hostname Nodo1SGM
ubuntu@ip-172-31-38-129:~$
```

i-05740401555f94080 (Nodo1SGM)
PublicIPs: 3.85.88.165 PrivateIPs: 172.31.38.129

Haremos lo mismo con el segundo nodo, pero poniendo obviamente Nodo2SGM. Al finalizar, nos queda reiniciar ambos servidores y el cambio estará aplicado.



```
aws Servicios Q Buscar [Alt+S]
Welcome to Ubuntu 22.04.5 LTS (GNU/Linux 6.8.0-1015-aws x86_64)

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

System information as of Mon Oct 14 12:39:10 UTC 2024

System load:  0.31           Processes:            107
Usage of /:   21.5% of 7.57GB Users logged in:       0
Memory usage: 20%           IPv4 address for eth0: 172.31.34.143
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

The list of available updates is more than a week old.
To check for new updates run: sudo apt update
New release '24.04.1 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Mon Oct 14 12:35:15 2024 from 18.206.107.28
ubuntu@Nodo2SGM:~$
```

4. Conexión SSH con las instancias EC2 de AWS (taller9)

Para que el nodo principal pueda conectarse con las instancias EC2 de AWS vamos a necesitar una clave SSH privada que nos habrá proporcionado AWS. En nuestro caso como estamos utilizando Amazon Academy vamos a utilizar la clave SSH que se nos asigna por defecto en el entorno de laboratorio. El archivo con la clave privada SSH se llama `vockey.pem`.

La clave SSH debe tener permisos de sólo lectura para el propietario del archivo, por lo que debemos realizar varias configuraciones.

Lo primero que debemos hacer es irnos al inicio de nuestro laboratorio, descargarnos el archivo `.pem` y meterlo en la carpeta “`vagrant`” que tenemos compartida. Si os habéis pasado el archivo y no os aparece en la carpeta, debéis reiniciar la máquina.

```
vagrant@AnsibleSGM:~/vagrant$ ls -la
total 17
drwxrwxrwx 1 vagrant vagrant 4096 Oct 14 12:17 .
drwxr-xr-x 21 root    root    4096 Oct 14 11:30 ..
-rwxrwxrwx 1 vagrant vagrant 1674 Oct 14 12:05 labsuser.pem
-rwxrwxrwx 1 vagrant vagrant 1438 Oct 14 12:05 labsuser.ppk
drwxrwxrwx 1 vagrant vagrant  0 Oct 14 12:18 .vagrant
-rwxrwxrwx 1 vagrant vagrant 493 Oct 14 11:51 vagrantfile
```

Este archivo tenemos que moverlo a un directorio nuevo que hemos creado (`P4_SGM`)

```
vagrant@AnsibleSGM:~/vagrant$ mkdir /home/vagrant/P4_SGM
vagrant@AnsibleSGM:~/vagrant$ cp labsuser.pem /home/vagrant/P4_SGM/
vagrant@AnsibleSGM:~/vagrant$ cd /home/vagrant/P4_SGM/
vagrant@AnsibleSGM:~/P4_SGM$
```

Esto se debe a que desde la carpeta compartida no podremos cambiarle los permisos y queremos darle permisos de sólo ejecución.

Pues ahora podremos hacerlo e iniciar sesión por ssh a ambos nodos

```
vagrant@AnsibleSGM:~/P4_SGM$ chmod 400 labsuser.pem
vagrant@AnsibleSGM:~/P4_SGM$ ls -l labsuser.pem
-r----- 1 vagrant vagrant 1674 Oct 14 12:32 labsuser.pem
vagrant@AnsibleSGM:~/P4_SGM$ ssh -i labsuser.pem ubuntu@3.85.88.165
Welcome to Ubuntu 22.04.5 LTS (GNU/Linux 6.8.0-1016-aws x86_64)

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

System information as of Mon Oct 14 12:33:29 UTC 2024

System load:  0.0               Processes:            102
Usage of /:   28.6% of 7.57GB   Users logged in:     1
Memory usage: 20%              IPv4 address for eth0: 172.31.38.129
Swap usage:   0%

 * Ubuntu Pro delivers the most comprehensive open source security and
   compliance features.

https://ubuntu.com/aws/pro

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

New release '24.04.1 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Mon Oct 14 12:02:38 2024 from 18.206.107.27
ubuntu@Nodo1SGM:~$
```

```
Connection to 3.91.37.163 closed.
vagrant@AnsibleSGM:~/P4_SGM$ ssh -i labsuser.pem ubuntu@3.91.37.163
The authenticity of host '3.91.37.163 (3.91.37.163)' can't be established.
ED25519 key fingerprint is SHA256:IZf9EoEq$CSNZy9e1A102gx5GrFXcR+JTygC2Hb0J2A.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '3.91.37.163' (ED25519) to the list of known hosts.
Welcome to Ubuntu 22.04.5 LTS (GNU/Linux 6.8.0-1015-aws x86_64)

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

System information as of Mon Oct 14 12:39:10 UTC 2024

System load:  0.31               Processes:           107
Usage of /:   21.5% of 7.57GB    Users logged in:    0
Memory usage: 20%               IPv4 address for eth0: 172.31.34.143
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

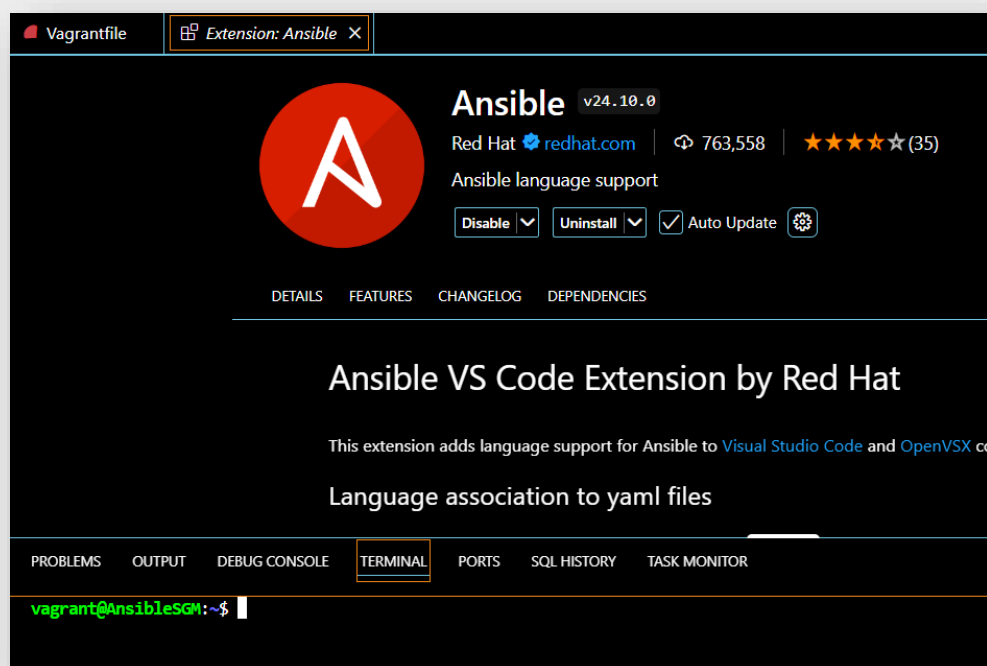
The list of available updates is more than a week old.
To check for new updates run: sudo apt update
New release '24.04.1 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Mon Oct 14 12:39:11 2024 from 18.206.107.29
ubuntu@Nodo2SGM:~$
```

5. Instalación de herramientas adicionales (taller 7)

De forma opcional podemos instalar algunas herramientas adicionales que nos ayuden a trabajar con Ansible.

La primera herramienta es la extensión de Ansible para Visual Studio Code. Lo que ofrece este plugin son resaltado sintáctico, validación y autocompletado, entre otras.



Otra herramienta importante a instalar es ansible-lint, que nos permite detectar errores en nuestros playbooks. Lo primero será instalar pip3 para poder instalarlo.

```
vagrant@AnsibleSGM:~$ sudo apt install python3 python3-pip -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
python3 is already the newest version (3.10.6-1~22.04.1).
python3-pip is already the newest version (22.0.2+dfsg-1ubuntu0.4).
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
vagrant@AnsibleSGM:~$ pip3 --version
```

Ahora ya podremos instalar ansible-lint.

```
Consider adding this directory to PATH or, if you prefer to suppress this warning, use --no-warn-script-location.
Successfully installed ansible-compat-24.9.1 ansible-lint-24.9.2 attrs-24.2.0 black-24.10.0 brace-2.5.post1 filelock-3.16.1 jsonschema-4.23.0 jsonschema-specifications-2024.10.1 markdown-it-py-3.0.0 mdurl-0.1.2 mypy-extensions-1.0.0 packaging-24.1 pathspec-0.12.1 platformdirs-4.3.6 pygments-2.18.0 referencing-0.35.1 rich-13.9.2 rpds-py-0.20.0 ruamel.yaml-0.18.6 ruamel.yaml.clib-0.2.8 subprocess-tee-0.4.2 tomli-2.0.2 typing-extensions-4.12.2 wcmatch-10.0 yamllint-1.35.1
vagrant@ansiblecon:~$ pip3 install "ansible-lint"
```

6. Configuración del inventario de Ansible (taller 8)

La única configuración que vamos a realizar en Ansible será editar el archivo de inventario del nodo principal, para incluir la lista de hosts sobre los que vamos a realizar tareas con Ansible.

El archivo de inventario que utiliza Ansible por defecto está en la ruta `/etc/ansible/hosts`. Lo modificaremos para que aparezcan las ip de nuestros nodos (no son elásticas así que tenemos que vigilar dichas IP).

```
GNU nano 6.2 /etc/ansible/hosts
# This is the default ansible 'hosts' file.

[aws]
54.198.165.190 ansible_user=ubuntu ansible_ssh_private_key_file=/home/vagrant/P4_SGM/labsuser.pem
35.153.99.76  ansible_user=ubuntu ansible_ssh_private_key_file=/home/vagrant/P4_SGM/labsuser.pem
```

Importante añadir la key privada, ya que, si no configuramos como en la imagen, no se conectará con los nodos (al menos en mi caso).

El archivo de inventario también se puede crear en otro directorio distinto a `/etc/ansible/hosts`, de hecho es habitual tener un archivo de inventario dentro de cada proyecto. En este caso, cuando vayamos a utilizar los comandos de Ansible deberemos indicar el directorio donde se encuentra el archivo de inventario con el parámetro `-i`.

7. Módulos de Ansible (taller 10)

Ansible dispone de una gran variedad de módulos que pueden ser utilizados desde la línea de comandos o en las tareas de los playbooks. Existen módulos para trabajar con clouds (Amazon, Azure, etc.), clustering (Kubernetes, Openshift, etc.), bases de datos (Influxdb, MongoDB, MySQL, PostgreSQL, etc.), monitorización, mensajería, etc.

A continuación, vamos a realizar una breve demostración de cómo utilizar algunos módulos con los comandos ad-hoc de Ansible:

- **Ping.** Esto comprobará que los nodos y el ansible se ven.

```
vagrant@ansible01:~/P4_50$ ansible all -m ping
[WARNING]: Platform linux on host 54.198.165.190 is using the discovered Python interpreter at /usr/bin/python3.10, but future installation of another Python interpreter could change the meaning of that path. See
https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
54.198.165.190 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3.10"
  },
  "changed": false,
  "ping": "pong"
}
[WARNING]: Platform linux on host 35.153.99.76 is using the discovered Python interpreter at /usr/bin/python3.10, but future installation of another Python interpreter could change the meaning of that path. See
https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
35.153.99.76 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3.10"
  },
  "changed": false,
  "ping": "pong"
}
vagrant@ansible01:~/P4_50$
```

- **Shell.** Nos permite ejecutar comando a través de una shell sobre cada uno de los nodos. En este caso el modificador -m nos permite indicar el módulo que queremos utilizar y el modificador -a nos permite indicar el comando.

```
vagrant@ansible01:~/P4_50$ ansible all -m shell -a "uname -a"
[WARNING]: Platform linux on host 54.198.165.190 is using the discovered Python interpreter at /usr/bin/python3.10, but future installation of another Python interpreter could change the meaning of that path. See
https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
54.198.165.190 | CHANGED | rc=0 >>
Linux Node150M 6.8.0-1015-aws #17-22.04.2-Ubuntu SMP Thu Sep 26 17:15:56 UTC 2024 x86_64 x86_64 GNU/Linux
[WARNING]: Platform linux on host 35.153.99.76 is using the discovered Python interpreter at /usr/bin/python3.10, but future installation of another Python interpreter could change the meaning of that path. See
https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
35.153.99.76 | CHANGED | rc=0 >>
Linux Node250M 6.8.0-1015-aws #16-22.04.1-Ubuntu SMP Mon Aug 19 19:38:17 UTC 2024 x86_64 x86_64 GNU/Linux
vagrant@ansible01:~/P4_50$
```

- **Command.** Ejecuta comandos en los hosts remotos sin pasar por el Shell.

```
vagrant@AnsibleSGM:~/P4_SGM$ ansible all -i /etc/ansible/hosts -m ansible.builtin.command -a "python3 --version"
[WARNING]: Platform linux on host 35.153.99.76 is using the discovered Python interpreter at /usr/bin/python3.10, but future installation of another Python interpreter could change the meaning of that path. See
https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
35.153.99.76 | CHANGED | rc=0 >>
Python 3.10.12
[WARNING]: Platform linux on host 54.198.165.190 is using the discovered Python interpreter at /usr/bin/python3.10, but future installation of another Python interpreter could change the meaning of that path. See
https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
54.198.165.190 | CHANGED | rc=0 >>
Python 3.10.12
vagrant@AnsibleSGM:~/P4_SGM$
```

- **Apt.** Es una forma automatizada de asegurarte de que todos los sistemas tienen la lista más actualizada de paquetes. Lo mismo que hacer `sudo apt update` en los nodos.

```
vagrant@AnsibleSGM:~/P4_SGM$ ansible all -m apt -a "update_cache=yes" -b
[WARNING]: Platform linux on host 35.153.99.76 is using the discovered Python interpreter at /usr/bin/python3.10, but future installation of another Python interpreter could change the meaning of that path. See
https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
35.153.99.76 | CHANGED => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3.10"
  },
  "cache_update_time": 1729174581,
  "cache_updated": true,
  "changed": true
}
[WARNING]: Platform linux on host 54.198.165.190 is using the discovered Python interpreter at /usr/bin/python3.10, but future installation of another Python interpreter could change the meaning of that path. See
https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
54.198.165.190 | CHANGED => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3.10"
  },
  "cache_update_time": 1729174581,
  "cache_updated": true,
  "changed": true
}
```


8. Playbooks (Taller 11)

Un Playbook es un archivo escrito en YAML donde se describen las tareas de configuración y administración que queremos realizar en cada uno de los nodos.

Un Playbook está formado por uno a varios Plays y un Play está formado por un conjunto de operaciones que vamos a realizar en los nodos.

En primer lugar, vamos a crear un playbook en el nodo principal. El nombre del archivo del archivo será `apache.yml` y las tareas que vamos a realizar en él serán la instalación del servidor web Apache y la activación del módulo `rewrite`.

El archivo `apache` debería quedar así (sin los símbolos de la imagen):

A screenshot of a terminal window with a dark background. The title bar at the top shows 'GNU nano 6.2' on the left and 'apache.yml *' on the right. The terminal content displays a YAML configuration for an Ansible play. It starts with 'hosts: aws' and 'become: yes'. The 'tasks' section contains two tasks: 'Install apache2' using the 'apt' module and 'Enable mod_rewrite' using the 'apache2_module' module. A 'notify' section follows with 'Restart apache2'. The 'handlers' section contains 'Restart apache2'. Finally, the 'service' section is configured with 'name: apache2' and 'state: restarted'.

```
---
- hosts: aws
  become: yes
  tasks:
    - name: Install apache2
      apt:
        name: apache2
        update_cache:
        state: latest
    - name: Enable mod_rewrite
      apache2_module:
        name: rewrite
        state: present
  notify:
    - Restart apache2
  handlers:
    - name: Restart apache2
  service:
    name: apache2
    state: restarted
```

Ahora que tenemos nuestro playbook listo, debemos ejecutarlo de la siguiente manera:

```
vagrant@ansible01:~/P4_SQ$ ansible-playbook -i /etc/ansible/hosts apache.yml

PLAY [aws]
TASK [Gathering Facts]
[WARNING]: Platform linux on host 54.198.165.198 is using the discovered Python interpreter at /usr/bin/python3.10, but future installation of another Python interpreter could change the meaning of that path. See
https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
ok: [54.198.165.198]
[WARNING]: Platform linux on host 35.153.99.76 is using the discovered Python interpreter at /usr/bin/python3.10, but future installation of another Python interpreter could change the meaning of that path. See
https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
ok: [35.153.99.76]
TASK [Install apache2]
changed: [35.153.99.76]
changed: [54.198.165.198]
TASK [Enable mod_rewrite]
changed: [35.153.99.76]
changed: [54.198.165.198]
PLAY RECAP
35.153.99.76 : ok=3 changed=2 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0
54.198.165.198 : ok=3 changed=2 unreachable=0 failed=0 skipped=0 rescued=0 ignored=0
vagrant@ansible01:~/P4_SQ$
```

Ahora comprobamos que se instaló correctamente abriendo la IP de uno de los nodos en nuestro navegador. (importante tener abierto el puerto 80 en las instancias)



9. Instalar pila LAMP con variables (Taller 12)

La pila LAMP es un conjunto de tecnologías que se utilizan comúnmente para desarrollar aplicaciones web. El acrónimo LAMP se descompone en los siguientes componentes:

- Linux
- Apache
- MySQL
- PHP

La combinación de estos cuatro componentes permite a los desarrolladores crear aplicaciones web dinámicas y escalables. LAMP es popular por su flexibilidad, facilidad de uso y la gran cantidad de recursos y documentación disponible.

El primer paso del ejemplo 5 será crear la carpeta ‘vars’, donde irán el archivo con las variables.

```
vagrant@AnsibleSGM:~/P4_SGM$ mkdir -p ~/P4_SGM/vars
vagrant@AnsibleSGM:~/P4_SGM$ nano ~/P4_SGM/vars/variables.yml
vagrant@AnsibleSGM:~/P4_SGM$ nano vars/variables.yml
```

Dentro del archivo de las variables, meteremos lo siguiente:

```
GNU nano 6.2 vars/variables.yml *
php_packages:
- php
- php-mysql
- libapache2-mod-php
```

Y para instalar LAMP, lo que pondremos en su archivo será lo siguiente:

```
GNU nano 6.2                                install_lamp.yml *
--
- name: Playbook para instalar la pila LAMP
  hosts: aws
  become: yes
  vars_files:
    - vars/variables.yml
  tasks:
    - name: Actualizar los repositorios
      apt:
        update_cache: yes
    - name: Instalar el servidor web Apache
      apt:
        name: apache2
        state: present
    - name: Instalar el sistema gestor de bases de datos MySQL
      apt:
        name: mysql-server
        state: present
    - name: Instalar PHP y los módulos necesarios
      apt:
        name: "{{ php_packages }}"
        state: present
    - name: Reiniciar el servidor web Apache
      service:
        name: apache2
        state: restarted
```

Finalmente, lo ejecutaremos y tendremos lista la instalación.

```
vagrant@ansible01:~/P4_509$ ansible-playbook -i /etc/ansible/hosts install_lamp.yml

PLAY [Playbook para instalar la pila LAMP] *****

TASK [Gathering Facts] *****
[WARNING]: Platform linux on host 35.153.99.76 is using the discovered Python interpreter at /usr/bin/python3.10, but future installation of another Python interpreter could change the meaning of that path. See
https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
ok: [35.153.99.76]
[WARNING]: Platform linux on host 54.198.165.190 is using the discovered Python interpreter at /usr/bin/python3.10, but future installation of another Python interpreter could change the meaning of that path. See
https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
ok: [54.198.165.190]

TASK [Actualizar los repositorios] *****
changed: [35.153.99.76]
changed: [54.198.165.190]

TASK [Instalar el servidor web Apache] *****
ok: [35.153.99.76]
ok: [54.198.165.190]

TASK [Instalar el sistema gestor de bases de datos MySQL] *****
ok: [54.198.165.190]
ok: [35.153.99.76]

TASK [Instalar PHP y los módulos necesarios] *****
ok: [54.198.165.190]
ok: [35.153.99.76]

TASK [Reiniciar el servidor web Apache] *****
changed: [35.153.99.76]
changed: [54.198.165.190]

PLAY RECAP *****
35.153.99.76      : ok=6  changed=2  unreachable=0  failed=0  skipped=0  rescued=0  ignored=0
54.198.165.190   : ok=6  changed=2  unreachable=0  failed=0  skipped=0  rescued=0  ignored=0
```

10. Despliegue aplicación web sencilla (Taller 12)

Para la instalación de una aplicación web sencilla, necesitaremos configurar tres archivos. El primero será 'main.yml', donde vendrán configurados los '.yml' que vamos a ejecutar en nuestro playbook.

```
GNU nano 6.2 main.yml *
---
- import_playbook: playbooks/install_lamp.yml
- import_playbook: playbooks/install_tools.yml
- import_playbook: playbooks/deploy_web.yml

```

El siguiente paso será configurar las variables que se ejecutarán con el playbook en vars/variables.yml.

```
GNU nano 6.2 vars/variables.yml *
db:
  name: lamp_db
  user: lamp_user
  password: lamp_password

phpmyadmin:
  user: pma_user
  password: pma_password
  db_name: db_name

```

También crearemos una carpeta de playbooks, para tenerlo mejor ordenado y meter el playbook que nos despliega la web.

```
vagrant@AnsibleSGM:~/P4_SGM$ nano main.yml
vagrant@AnsibleSGM:~/P4_SGM$ nano vars/variables.yml
vagrant@AnsibleSGM:~/P4_SGM$ mkdir playbooks
vagrant@AnsibleSGM:~/P4_SGM$ nano playbooks/deploy_web.yml

```

El siguiente paso es configurar el archivo con todas las especificaciones para crear todo lo relacionado con las bases de datos.

```
GNU nano 6.2                                playbooks/deploy t
--
- name: Playbook para hacer el deploy de una aplicación web
  hosts: aws
  become: yes
  vars_files:
    - vars/variables.yml
  tasks:
    - name: Instalamos el gestor de paquetes de Python pip3
      apt:
        name: python3-pip
        state: present

    - name: Instalamos el módulo de pymysql
      pip:
        name: pymysql
        state: present

    - name: Crear una base de datos
      mysql_db:
        name: "{{ db.name }}"
        state: present
        login_unix_socket: /var/run/mysqld/mysqld.sock

    - name: Crear el usuario de la base de datos
      mysql_user:
        name: "{{ db.user }}"
        password: "{{ db.password }}"
        priv: "{{ db.name }}.*:ALL"
        state: present
        login_unix_socket: /var/run/mysqld/mysqld.sock

    - name: Descargar el código fuente de GitHub
      git:
        repo: https://github.com/josejuansanchez/iaw-practica-lamp.git
        dest: /tmp/iaw-practica-lamp
        force: yes
        update: yes

    - name: Copiar el código fuente
      copy:
        src: /tmp/iaw-practica-lamp/src/
        dest: /var/www/html
        remote_src: yes
        mode: '0755'

    - name: Editar el nombre de la base de datos en el archivo de configuración
```

Otro archivo importante es el ‘install_tools’, en el que instalaremos todos los comandos/aplicaciones necesarias para que se instale/ejecuten.

```
GNU nano 6.2                                playbooks/install_tools.yml *
---
- name: Playbook para instalar herramientas necesarias
  hosts: aws
  become: yes
  tasks:
  - name: Instalar Git
    apt:
      name: git
      state: present
      update_cache: yes
  - name: Instalar unzip
    apt:
      name: unzip
      state: present
  - name: Instalar curl
    apt:
      name: curl
      state: present
  - name: Instalar otras herramientas útiles
    apt:
      name:
        - htop
        - vim
        - net-tools
      state: present
```

Todos los archivos listos, procedemos a ejecutar el playbook por ‘main.yml’.

```
vsrgar@ansible01:~/P4_50$ ansible-playbook -i /etc/ansible/hosts main.yml

PLAY [Playbook para instalar herramientas adicionales] *****

TASK [gathering Facts] *****
[WARNING]: Platform linux on host 35.153.99.76 is using the discovered Python interpreter at /usr/bin/python3.10, but future installation of another Python interpreter could change the meaning of that path. See
https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
ok: [35.153.99.76]
[WARNING]: Platform linux on host 54.198.165.190 is using the discovered Python interpreter at /usr/bin/python3.10, but future installation of another Python interpreter could change the meaning of that path. See
https://docs.ansible.com/ansible-core/2.17/reference_appendices/interpreter_discovery.html for more information.
ok: [54.198.165.190]

TASK [descargar phpMyAdmin] *****
changed: [35.153.99.76]
changed: [54.198.165.190]

TASK [instalar unzip] *****
changed: [35.153.99.76]
changed: [54.198.165.190]

TASK [descomprimir phpMyAdmin] *****
changed: [35.153.99.76]
changed: [54.198.165.190]

TASK [copiar phpMyAdmin] *****
changed: [54.198.165.190]
changed: [35.153.99.76]

TASK [cambiar el propietario y el grupo de phpMyAdmin] *****
changed: [54.198.165.190]
changed: [35.153.99.76]

TASK [instalar pip3] *****
changed: [35.153.99.76]
changed: [54.198.165.190]

TASK [instalar PyMySQL] *****
changed: [35.153.99.76]
changed: [54.198.165.190]

TASK [crear la base de datos para phpMyAdmin] *****
changed: [35.153.99.76]
changed: [54.198.165.190]

TASK [importar la base de datos de phpMyAdmin] *****
changed: [54.198.165.190]
changed: [35.153.99.76]

TASK [crear el usuario de phpMyAdmin] *****
[WARNING]: Option column_case_sensitive is not provided. The default is now false, so the column's name will be uppercased. The default will be changed to true in community.mysql 4.8.0.
changed: [54.198.165.190]
changed: [35.153.99.76]
```

```
ok: [35.153.99.76]
ok: [54.198.165.190]

TASK [Instalamos el módulo de pymysql] *****
ok: [35.153.99.76]
ok: [54.198.165.190]

TASK [Crear una base de datos] *****
changed: [35.153.99.76]
changed: [54.198.165.190]

TASK [Crear el usuario de la base de datos] *****
changed: [35.153.99.76]
changed: [54.198.165.190]

TASK [Descargar el código fuente de GitHub] *****
changed: [35.153.99.76]
changed: [54.198.165.190]

TASK [Copiar el código fuente] *****
changed: [35.153.99.76]
changed: [54.198.165.190]

TASK [Editar el nombre de la base de datos en el archivo de configuración] *****
ok: [35.153.99.76]
ok: [54.198.165.190]

TASK [Editar el usuario de la base de datos en el archivo de configuración] *****
ok: [35.153.99.76]
ok: [54.198.165.190]

TASK [Editar el password del usuario de la base de datos en el archivo de configuración] *****
ok: [35.153.99.76]
ok: [54.198.165.190]

TASK [Importar la base de datos] *****
changed: [35.153.99.76]
changed: [54.198.165.190]

TASK [Eliminar el directorio iaw-practica-lamp] *****
changed: [35.153.99.76]
changed: [54.198.165.190]

TASK [Cambiar el propietario del directorio /var/www/html] *****
changed: [35.153.99.76]
changed: [54.198.165.190]

PLAY RECAP *****
35.153.99.76      : ok=29   changed=8    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
54.198.165.190   : ok=29   changed=8    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

vagrant@AnsibleSQM:~/P4 SQM$ ansible --version
```


Ya está configurada e instalada. Nos queda verificar que se ha instalado correctamente.

La primera forma es entrar en el primer nodo, por ejemplo, y entrar en sql.

Si entramos en sql y que nos muestre la base de datos instalada.

```
ubuntu@Nodo1SGM:~$ sudo mysql -u root
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 30
Server version: 8.0.39-0ubuntu0.22.04.1 (Ubuntu)

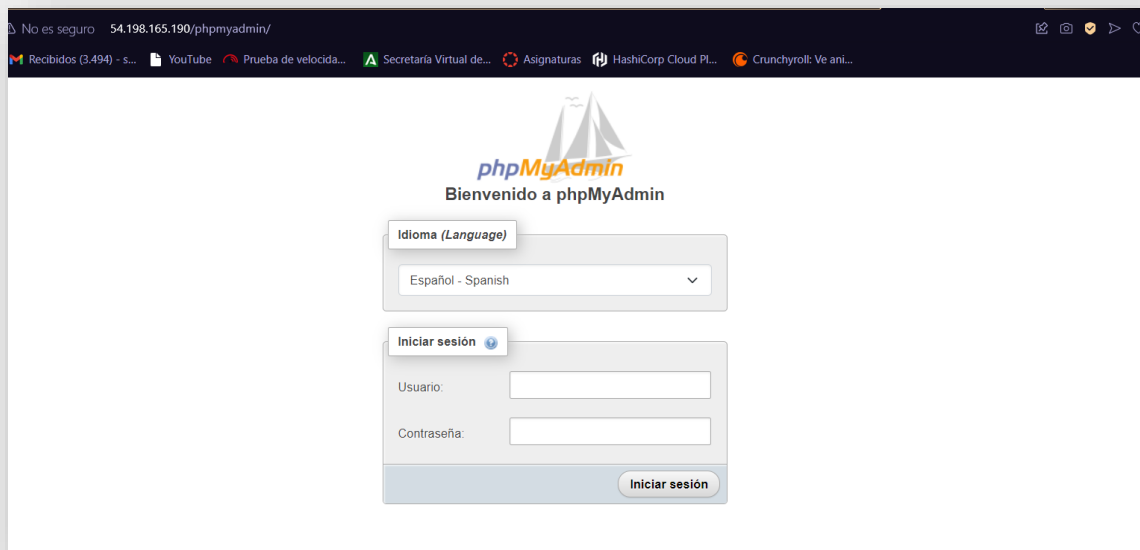
Copyright (c) 2000, 2024, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> SHOW DATABASES;
+-----+
| Database |
+-----+
| db_name  |
| information_schema |
| lamp_db  |
| mysql    |
| performance_schema |
| phpmyadmin |
| sys      |
+-----+
7 rows in set (0.01 sec)
```

La otra forma es pegar la IP de la siguiente forma “XXXXXXX/phpmyadmin/” y accederemos al login de la aplicación que hemos creado.



Iniciamos con las credenciales por defecto que teníamos en las variables y ya estaríamos.

