



Certified Tech Developer

The Ultimate Degree

Infraestructura II

Ejercitación Ansible

Objetivo:

- Crear una instancia EC2 desde Ansible utilizando una clave SSH creada desde AWS.

Algunas recomendaciones

Lista de los módulos de AWS en Ansible. [Link](#)

Verificar que tenemos instalado correctamente los siguiente componentes en nuestras máquinas virtuales:

```
mati@mati-virtual-machine: ~  
mati@mati-virtual-machine:~$ pip list | grep boto  
boto3                1.26.83  
botocore              1.29.83  
mati@mati-virtual-machine:~$ python3 --version  
Python 3.10.6  
mati@mati-virtual-machine:~$ aws --version  
aws-cli/2.11.0 Python/3.11.2 Linux/5.19.0-35-generic exe/x86_64.ubuntu.22 prompt  
/off  
mati@mati-virtual-machine:~$ ansible --version  
ansible [core 2.14.3]  
  config file = /etc/ansible/ansible.cfg  
  configured module search path = ['/home/mati/.ansible/plugins/modules', '/usr/  
share/ansible/plugins/modules']  
  ansible python module location = /usr/lib/python3/dist-packages/ansible  
  ansible collection location = /home/mati/.ansible/collections:/usr/share/ansib  
le/collections  
  executable location = /usr/bin/ansible  
  python version = 3.10.6 (main, Nov 14 2022, 16:10:14) [GCC 11.3.0] (/usr/bin/p  
ython3)  
  jinja version = 3.0.3  
  libyaml = True  
mati@mati-virtual-machine:~$
```

```
mati@mati-virtual-machine: ~  
mati@mati-virtual-machine:~$ ansible-galaxy collection list amazon.aws  
  
# /home/mati/.ansible/collections/ansible_collections  
Collection Version  
-----  
amazon.aws 5.2.0  
  
# /usr/lib/python3/dist-packages/ansible_collections  
Collection Version  
-----  
amazon.aws 5.2.0  
mati@mati-virtual-machine:~$
```

Para instalar boto y boto3 debemos tener instalado pip. Para ello ejecutamos los siguientes comandos:

sudo apt install pip

pip install boto3

Template modelo de un playbook para levantar y ejecutar una instancia en AWS desde Ansible

```
---  
  
- hosts: localhost  
  
  tasks:  
  
    - name: ec2  
  
      amazon.aws.ec2_instance:  
  
        name: "nombre-de-tu-instancia"  
  
        key_name: <TU LLAVE>  
  
        availability_zone: us-east-1a  
  
        instance_type: t1.micro  
  
        network:  
  
          assign_public_ip: true  
  
          image_id: <AMI AWS>
```

Una vez que ya tenemos generado nuestro playbook lo ejecutamos de la siguiente manera:

“ansible-playbook aws-ec2-instance.yaml” —> En este caso el nombre del playbook es “aws-ec2-instance.yaml” y el comando para ejecutar el playbook es “ansible-playbook”

```
matias@matias-virtual-machine: ~/Documents/Ansible
matias@matias-virtual-machine:~/Documents/Ansible$ ansible-playbook aws-ec2-instance.yaml
[WARNING]: provided hosts list is empty, only localhost is available. Note that the implicit
localhost does not match 'all'
[WARNING]: Collection amazon.aws does not support Ansible version 2.10.8

PLAY [localhost] *****

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

TASK [ec2] *****
changed: [localhost]

PLAY RECAP *****
localhost : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=
0    ignored=0

matias@matias-virtual-machine:~/Documents/Ansible$
```

Obteniendo como resultado dicho mensaje de Éxito.

Challenge

Objetivo:

- Crear una llave RSA (desde nuestra computadora) para poder subir mediante un playbook dicha clave a AWS.
- Utilizar la clave para crear una nueva instancia EC2 (con sistema operativo Linux) que utilice la clave que creamos anteriormente.
- Conectarnos a esa instancia con la clave que generamos desde nuestra PC.