EDGAR ALVARADO

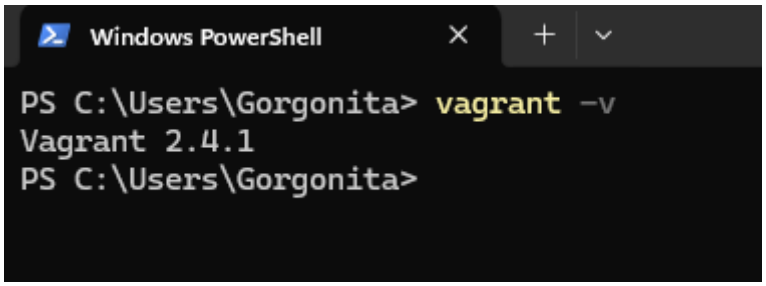
DEVOPS BOOTCAMP

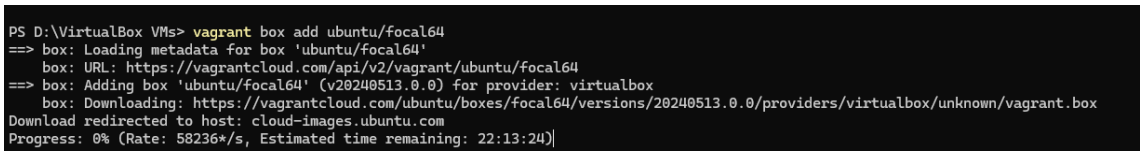
DESAFIO 11

Etapa 1: Vagrant

1. Crear una máquina virtual utilizando Vagrant, recomendamos el uso de una box que utilice Ubuntu como base, el objetivo es crear nuestra máquina de trabajo en la cual tendremos nuestras herramientas (terraform + ansible) para poder seguir así nuestro desafío.

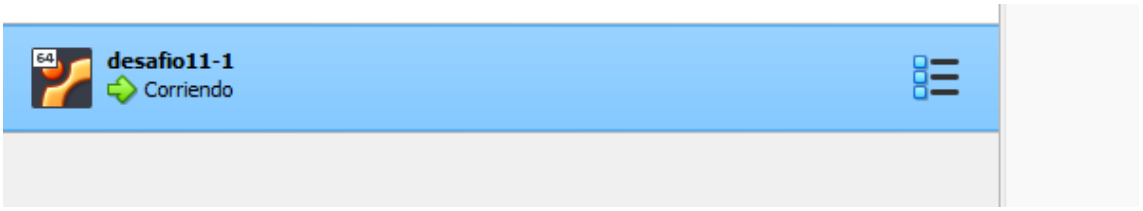
Instalado Vagrant y configurando maquina virtual con Ubuntu





```
PS D:\VirtualBox VMs\desafio_11_ubuntu> vagrant up
Bringing machine 'desafio11' up with 'virtualbox' provider...
==> desafio11: Importing base box 'ubuntu/focal64'...
==> desafio11: Matching MAC address for NAT networking...
==> desafio11: Checking if box 'ubuntu/focal64' version '20240513.0.0' is up to date...
==> desafio11: Setting the name of the VM: desafio11-1
Vagrant is currently configured to create VirtualBox synced folders with
the `SharedFoldersEnableSymlinksCreate` option enabled. If the Vagrant
guest is not trusted, you may want to disable this option. For more
information on this option, please refer to the VirtualBox manual:

  https://www.virtualbox.org/manual/ch04.html#sharedfolders

This option can be disabled globally with an environment variable:

  VAGRANT_DISABLE_VBOXSYMLINKCREATE=1

or on a per folder basis within the Vagrantfile:

  config.vm.synced_folder '/host/path', '/guest/path', SharedFoldersEnableSymlinksCreate: false
==> desafio11: Clearing any previously set network interfaces...
==> desafio11: Preparing network interfaces based on configuration...
    desafio11: Adapter 1: nat
    desafio11: Adapter 2: hostonly
==> desafio11: Forwarding ports...
    desafio11: 22 (guest) => 2222 (host) (adapter 1)
==> desafio11: Running 'pre-boot' VM customizations...
==> desafio11: Booting VM...
==> desafio11: Waiting for machine to boot. This may take a few minutes...
```

```
VirtualBox Guest Additions: kernel modules and services 7.0.12 r159484 reloaded
VirtualBox Guest Additions: NOTE: you may still consider to re-login if some
user session specific services (Shared Clipboard, Drag and Drop, Seamless or
Guest Screen Resize) were not restarted automatically
Unmounting Virtualbox Guest Additions ISO from: /mnt
==> desafio11: Checking for guest additions in VM...
==> desafio11: Setting hostname...
==> desafio11: Configuring and enabling network interfaces...
==> desafio11: Mounting shared folders...
    desafio11: /vagrant => D:/VirtualBox VMs/Desafio_11_Ubuntu
PS D:\VirtualBox VMs\desafio_11_ubuntu>
```



```
PS D:\VirtualBox VMs\desafio_11_ubuntu>  vagrant ssh
Welcome to Ubuntu 20.04.6 LTS (GNU/Linux 5.4.0-182-generic x86_64)

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

 System information as of Wed May 15 16:52:07 UTC 2024

  System load:  0.08              Processes:               119
  Usage of /:   4.2% of 38.70GB   Users logged in:         0
  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

New release '22.04.3 LTS' available.
Run 'do-release-upgrade' to upgrade to it.


vagrant@desafio11:~$
```

```
vagrant@desafio11:~$ sudo apt update && sudo apt upgrade
Get:1 http://security.ubuntu.com/ubuntu focal-security InRelease [114 kB]
Hit:2 http://archive.ubuntu.com/ubuntu focal InRelease
Get:3 http://archive.ubuntu.com/ubuntu focal-updates InRelease [114 kB]
Get:4 http://security.ubuntu.com/ubuntu focal-security/main amd64 Packages [2951 kB]
Get:5 http://archive.ubuntu.com/ubuntu focal-backports InRelease [108 kB]
Get:6 http://archive.ubuntu.com/ubuntu focal/universe amd64 Packages [8628 kB]
```

Instalando Terraform

```
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
vagrant@desafio11:~$ wget -O- https://apt.releases.hashicorp.com/gpg | sudo gpg --dearmor -o /usr/share/keyrings/hashicorp-archive-keyring.gpg
--2024-05-15 17:41:40--  https://apt.releases.hashicorp.com/gpg
Resolving apt.releases.hashicorp.com (apt.releases.hashicorp.com)... 3.160.90.23, 3.160.90.102, 3.160.90.44, ...
Connecting to apt.releases.hashicorp.com (apt.releases.hashicorp.com)|3.160.90.23|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 3980 (3.9K) [binary/octet-stream]
Saving to: 'STDOUT'

-                           100%[=================================================================>]   3.89K  --.-KB/s    in 0s

2024-05-15 17:41:40 (61.9 MB/s) - written to stdout [3980/3980]

vagrant@desafio11:~$ echo "deb [signed-by=/usr/share/keyrings/hashicorp-archive-keyring.gpg] https://apt.releases.hashicorp.com $(lsb_release -cs) m
ain" | sudo tee /etc/apt/sources.list.d/hashicorp.list
deb [signed-by=/usr/share/keyrings/hashicorp-archive-keyring.gpg] https://apt.releases.hashicorp.com focal main
vagrant@desafio11:~$
```

```
vagrant@desafio11:~$ sudo apt update && sudo apt install terraform
Get:1 https://apt.releases.hashicorp.com focal InRelease [16.4 kB]
Hit:2 http://security.ubuntu.com/ubuntu focal-security InRelease
Hit:3 http://archive.ubuntu.com/ubuntu focal InRelease
Hit:4 http://archive.ubuntu.com/ubuntu focal-updates InRelease
Get:5 https://apt.releases.hashicorp.com focal/main amd64 Packages [129 kB]
Hit:6 http://archive.ubuntu.com/ubuntu focal-backports InRelease
Fetched 145 kB in 2s (63.1 kB/s)
Reading package lists... Done
Building dependency tree
Reading state information... Done
All packages are up to date.
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  terraform
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 27.7 MB of archives.
After this operation, 88.2 MB of additional disk space will be used.
Get:1 https://apt.releases.hashicorp.com focal/main amd64 terraform amd64 1.8.3-1 [27.7 MB]
Fetched 27.7 MB in 6s (4291 kB/s)
Selecting previously unselected package terraform.
(Reading database ... 69445 files and directories currently installed.)
Preparing to unpack .../terraform_1.8.3-1_amd64.deb ...
Unpacking terraform (1.8.3-1) ...
Setting up terraform (1.8.3-1) ...
vagrant@desafio11:~$
```

```
Setting up terraform (1.8.3-1) ...
vagrant@desafio11:~$ terraform -version
Terraform v1.8.3
on linux_amd64
vagrant@desafio11:~$
```

Instalando Ansible

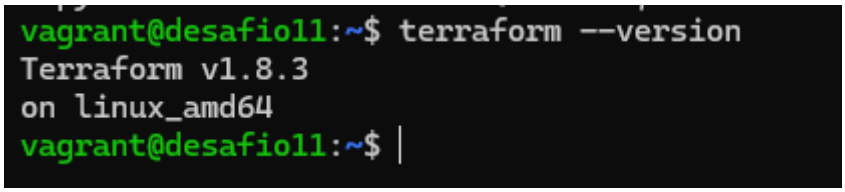
```
vagrant@desafio11:~$ sudo apt update && sudo apt install ansible
Hit:1 https://download.docker.com/linux/ubuntu focal InRelease
Hit:2 https://apt.releases.hashicorp.com focal InRelease
Get:3 http://security.ubuntu.com/ubuntu focal-security InRelease [114 kB]
Hit:4 http://archive.ubuntu.com/ubuntu focal InRelease
Get:5 http://archive.ubuntu.com/ubuntu focal-updates InRelease [114 kB]
Hit:6 http://archive.ubuntu.com/ubuntu focal-backports InRelease
Fetched 228 kB in 2s (118 kB/s)
Reading package lists... Done
Building dependency tree
Reading state information... Done
All packages are up to date.
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  ieee-data python3-argcomplete python3-crypto python3-dnspython python3-jmespath python3-kerberos python3-libcloud python3-lockfile
  python3-netaddr python3-ntlm-auth python3-requests-kerberos python3-requests-ntlm python3-selinux python3-winrm python3-xmltodict
Suggested packages:
```

```
vagrant@desafio11:~$ ansible --version
ansible 2.9.6
  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
  executable location = /usr/bin/ansible
  python version = 3.8.10 (default, Nov 22 2023, 10:22:35) [GCC 9.4.0]
vagrant@desafio11:~$
```

Etapa 2: Terraform

NOTA: Esta parte del desafío es la más importante por lo tanto la que mayor impacto tendrá sobre la nota.

1. Ya teniendo terraform instalado en la VM, tendrán que crear un entorno de trabajo de terraform el cual constara del provider de la nube de su elección y los archivos de configuración requeridos para la práctica.



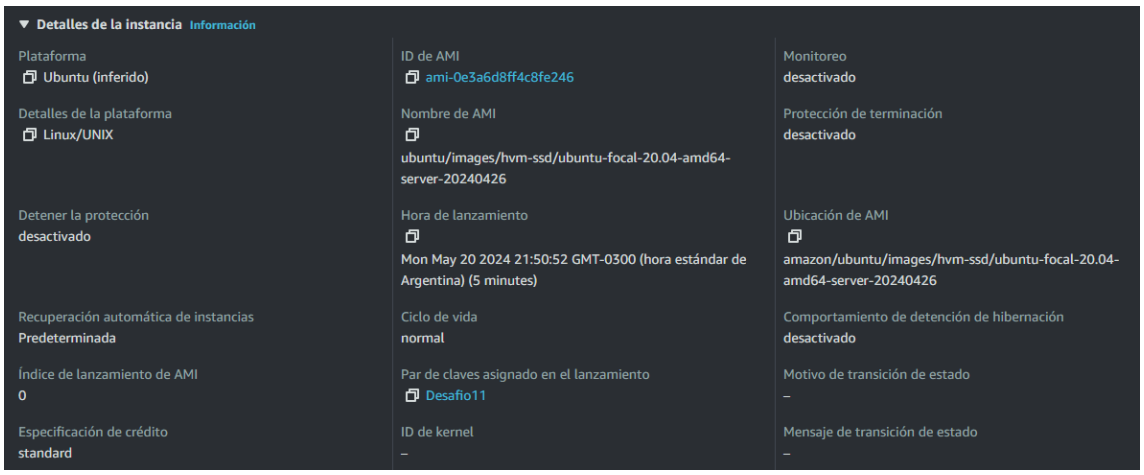
2. Hay múltiples formas de hacer lo solicitado en el desafio: se pueden utilizar modulos, se puede hardcodear valores, se pueden utilizar múltiples variables, pueden utilizar la VPC default o una VPC creada por ustedes, etc. La complejidad de la solución y documentación de esta, tendrá impacto directo en la nota del desafío.

Terraform > main.tf

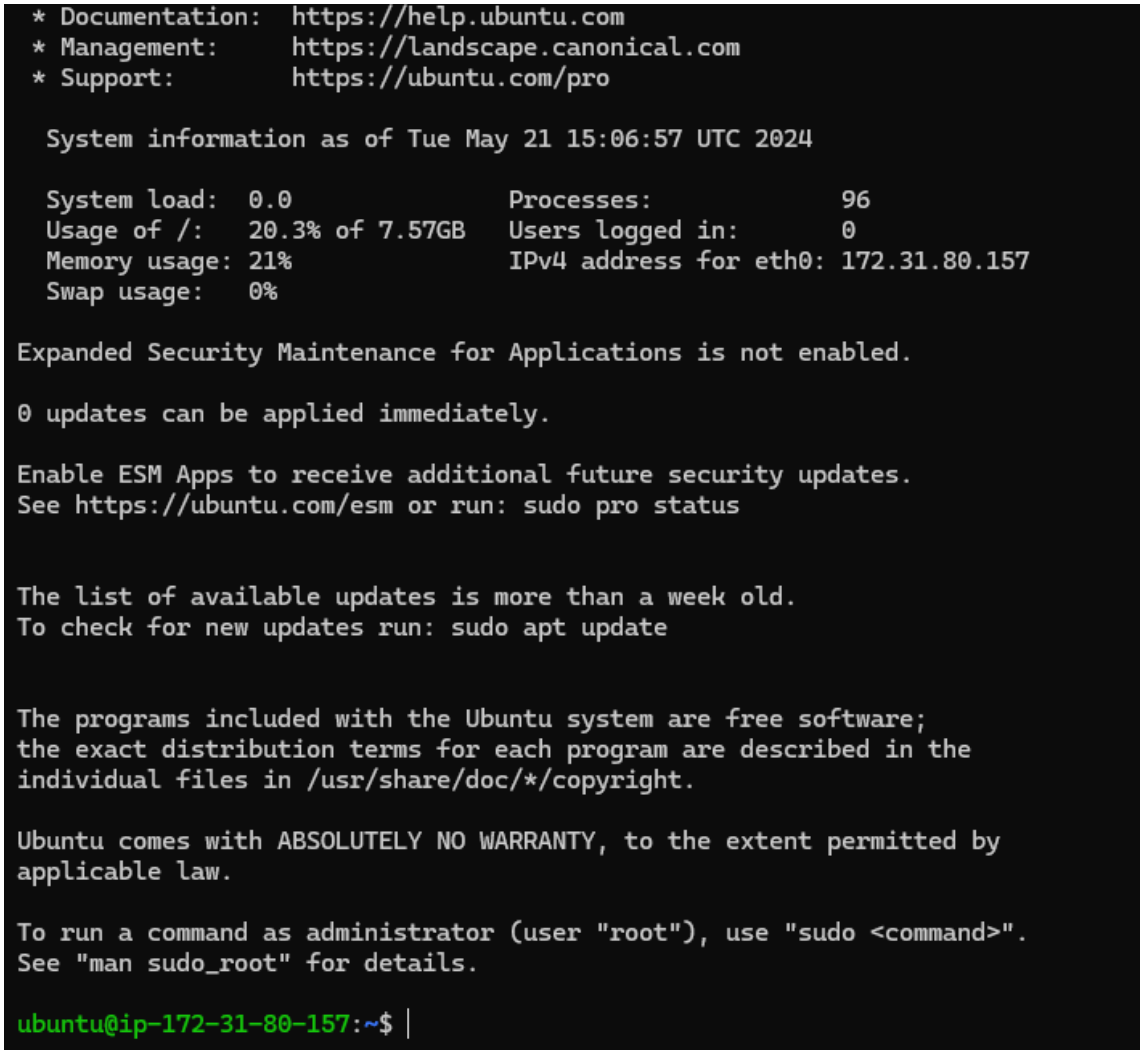
```
provider "aws" {
    access_key = "${var.access_key}"
    secret_key = "${var.secret_key}"
    region = "us-east-1"
}

data “aws_ami” “ubuntu” {
most_recent = true

filter {
name = “name”
values = [“ubuntu/images/hvm-ssd/ubuntu-focal-20.04-amd64-server-*”]
}

filter {
name = “virtualization-type”
values = [“hvm”]
}

owners = [“099720109477”] # Canonical
}

resource "aws_instance" "ec2_instance" {
    ami = data.aws_ami.ubuntu.id
    count = "${var.number_of_instances}"
    subnet_id = "${var.subnet_id}"
    instance_type = "${var.instance_type}"
    key_name = "${var.ami_key_pair_name}"
    Name = "${var.instance_name}"
}
```

Terraform > variables.tf

```
variable "access_key" {
        description = "Access key to AWS console"
}
variable "secret_key" {
        description = "Secret key to AWS console"
}

variable "instance_name" {
        description = "Name of the instance to be created"
        default = "Desafio11"
}

variable "instance_type" {
        default = "t2.micro"
}

variable "subnet_id" {
        description = "The VPC subnet the instance(s) will be created in"
        default = "subnet-04039c229f950def1"
}

variable "number_of_instances" {
        description = "number of instances to be created"
        default = 1
}

variable "ami_key_pair_name" {
        default = "Desafio11"
}
```



3. Verificar el acceso a la instancia por ssh.



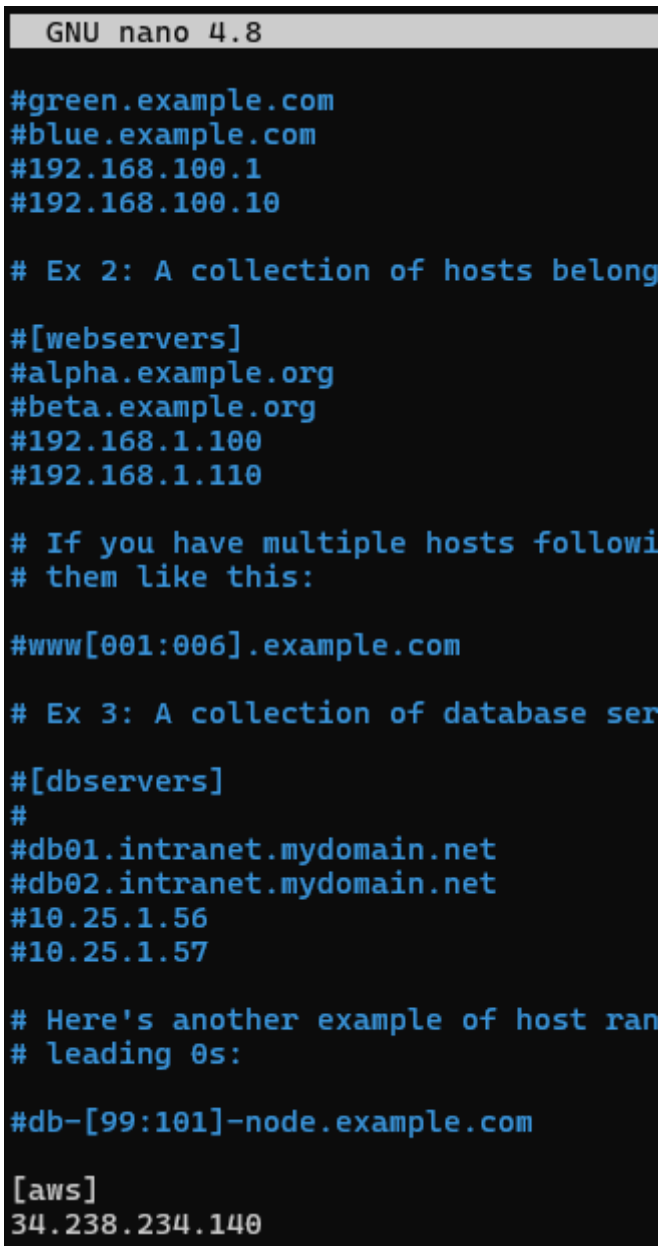
Consideraciones: Tendrán que basarse en la documentación de los recursos para ciertos aspectos de configuración como por ejemplo la cantidad de storage asignada al volumen root de la instancia, la configuración de la llave para conectarnos por ssh, etc.

Es muy importante que, a la hora de configurar las credenciales de la nube, lo hagan de una forma segura o al menos no poniéndolas en texto plano en los archivos de configuración.

En caso de encontrar credenciales en los entregables, se restará puntos.

Etapa 3: Ansible

1. El primer paso será crear un inventario agregando la IP publica de nuestra instancia.



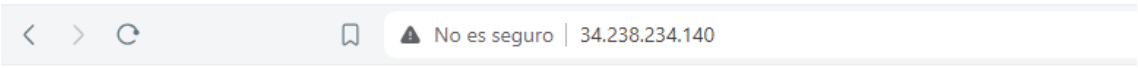
2. Una vez creado el inventario, tendrán que ejecutar el comando ping para probar la conexión de ansible a la instancia, tengan en cuenta que tendrán que usar la llave SSH para la conexión, como asi tambien confirmar el usuario que usaran (ec2-user, ubuntu, etc)

```
vagrant@desafio11:/etc/ansible$ sudo ansible -m ping all
34.238.234.140 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
vagrant@desafio11:/etc/ansible$
```

3. Una vez confirmada la conexión, tendrán que crear un playbook que instale un servidor web, y cambiar el html para que al probar el servidor web nos muestre otro mensaje.

```
  GNU nano 4.8
---
- name: Install Apache on Ubuntu
  hosts: ec2
  become: yes

  tasks:
    - name: Update apt package cache
      apt:
        update_cache: yes

    - name: Install Apache
      apt:
        name: apache2
        state: present

    - name: copiar index.html del localhost
      copy:
        src: /etc/ansible/index.html
        dest: /var/www/html/

    - name: Start Apache service
      service:
        name: apache2
        state: started
```

```
vagrant@desafio11:/etc/ansible$ sudo nano index.html
vagrant@desafio11:/etc/ansible$ sudo ansible-playbook apache.yml

PLAY [Install Apache on Ubuntu] ************************************************************************************************************************************

TASK [Gathering Facts] *********************************************************************************************************************************************
ok: [34.238.234.140]

TASK [Update apt package cache] ************************************************************************************************************************************
changed: [34.238.234.140]

TASK [Install Apache] **********************************************************************************************************************************************
ok: [34.238.234.140]

TASK [copiar index.html del localhost] *****************************************************************************************************************************
changed: [34.238.234.140]

TASK [Start Apache service] ****************************************************************************************************************************************
ok: [34.238.234.140]

PLAY RECAP *********************************************************************************************************************************************************
34.238.234.140             : ok=5    changed=2    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

vagrant@desafio11:/etc/ansible$
```



PAGINA WEB DE PRUEBA. DEPLOY VIA ANSIBLE/AWS. DESAFIO 11.

Consideraciones: Tengan en cuenta el usuario a utilizar para la conexión, los tipos de tareas (apt si es basado en ubuntu, yum si es basado en centos, etc).

Además, tengan en consideración la llave a utilizar para la conexión