Universidad ORT Uruguay **Facultad de Ingeniería: Escuela de Tecnología**

OBLIGATORIO_TALLER LINUX

Ezequiel Britos N.º 339443

Fernando Sosa N.º 182835

Grupo: M3A

Docente: Miguel Alejandro Vartabedián Ruiz Díaz

Servidores

ub01

Nombre: ub01

CPUs: 1

RAM: 2.0GB

FileSystem: / 10GB

/boot: 2GB

/var 5GB

SWAP 4GB

Red: Adaptador1: NAT (enp0s3-DHCP)

Adaptador2: Red Interna (enp0s8-192.168.1.20)

```
sysadmin@ub01 ~> hostname && cat /proc/cpuinfo | grep "cpu cores" && free -h && df -h && ip a
ub01
cpu cores       : 1
               total        used        free      shared  buff/cache   available
Mem:           1.9Gi       278Mi       1.6Gi       768Ki       232Mi       1.6Gi
Swap:          4.0Gi          0B       4.0Gi
Filesystem      Size  Used Avail Use% Mounted on
tmpfs           197M  756K  197M   1% /run
/dev/sda2       9.8G  1.8G  7.5G  19% /
tmpfs           985M     0  985M   0% /dev/shm
tmpfs           5.0M     0  5.0M   0% /run/lock
/dev/sda3       2.0G   99M  1.7G   6% /boot
/dev/sda4       4.9G  373M  4.3G   8% /var
tmpfs           197M   12K  197M   1% /run/user/1000
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
       valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:05:f7:99 brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 metric 100 brd 10.0.2.255 scope global dynamic enp0s3
       valid_lft 86249sec preferred_lft 86249sec
    inet6 fe80::a00:27ff:fe05:f799/64 scope link
       valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:b2:d0:66 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.20/24 brd 192.168.1.255 scope global enp0s8
       valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:feb2:d066/64 scope link
       valid_lft forever preferred_lft forever
sysadmin@ub01 ~> _
```

SSH-key:

```
sysadmin@Bastion ~> ssh-keygen
Generating public/private ed25519 key pair.
Enter file in which to save the key (/home/sysadmin/.ssh/id_ed25519):
Created directory '/home/sysadmin/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/sysadmin/.ssh/id_ed25519
Your public key has been saved in /home/sysadmin/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:6z2j98apIHsi+B3iuhuZ4ubXgeySevKYinhCBNF/OPw sysadmin@Bastion
The key's randomart image is:
+--[ED25519 256]--+
|.o               |
|. .              |
|.  o .           |
| .  = .          |
|. . .+  S        |
| . = .E  .       |
|o *.o + o  . .   |
|*B+=.+.=.o+ +    |
|%OB=o.ooooo*.    |
+----[SHA256]-----+
sysadmin@Bastion ~> ssh-copy-id 192.168.1.20
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/sysadmin/.ssh/id_ed25519.pub"
The authenticity of host '192.168.1.20 (192.168.1.20)' can't be established.
ED25519 key fingerprint is SHA256:6AB61iznmNpvAkiY4ztJhHsTzOJd5xnoP1c1k5HUp8g.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
/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@192.168.1.20's password:

Number of key(s) added: 1

Now try logging into the machine, with:   "ssh '192.168.1.20'"
and check to make sure that only the key(s) you wanted were added.

sysadmin@Bastion ~> _
```

Centos01

Nombre: centos01

CPUs: 1

RAM: 2.0GB

FileSystem: / 10GB

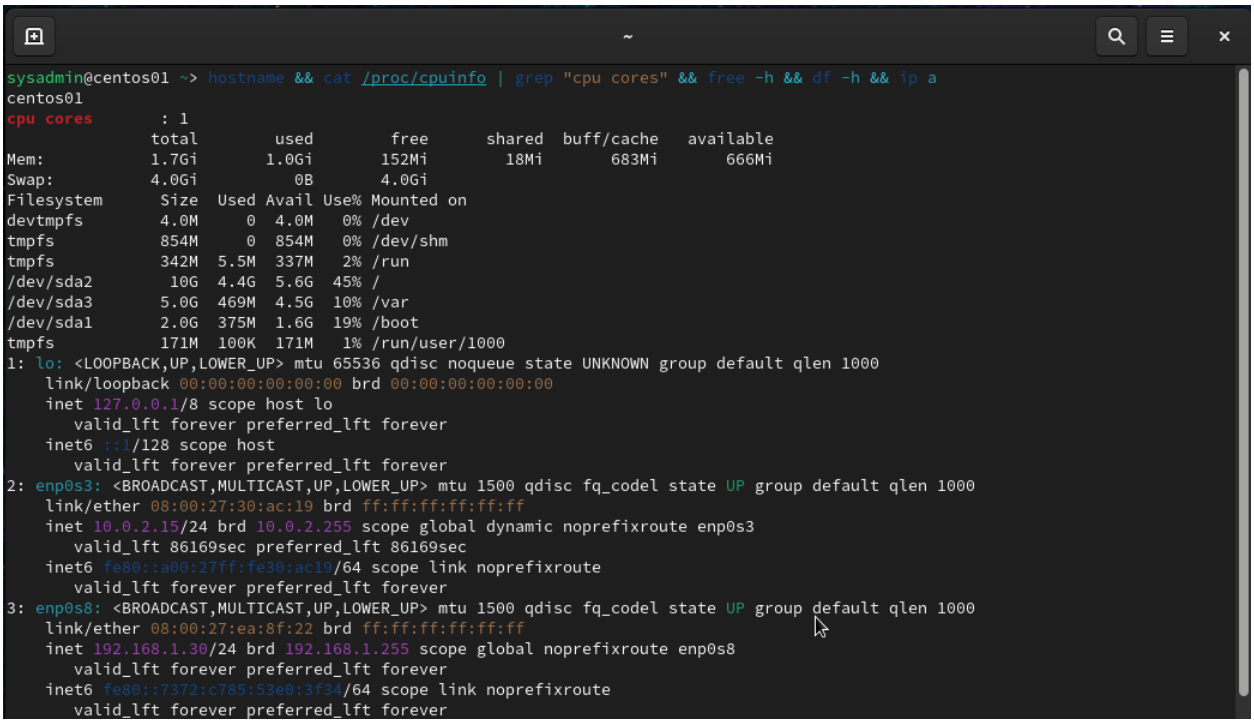
/boot: 2GB

/var 5GB

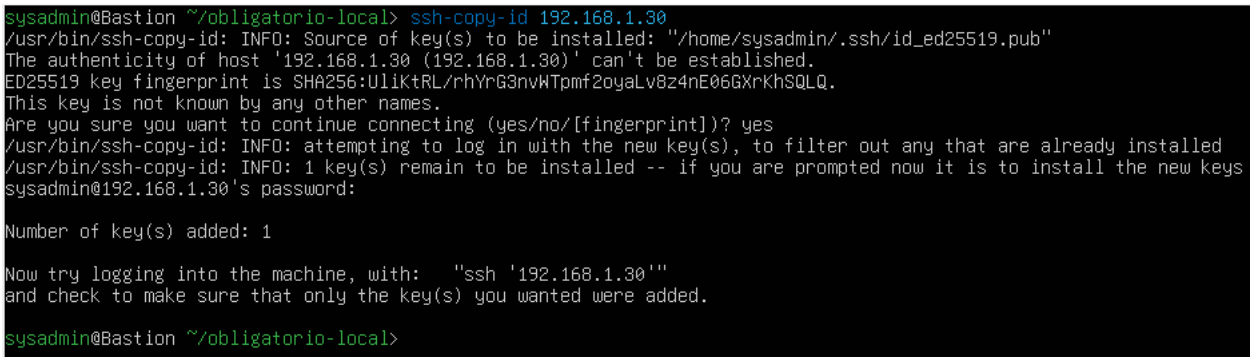
SWAP 4GB

Red: Adaptador1: NAT (enp0s3-DHCP)

Adaptador2: Red Interna (enp0s8-192.168.1.30)


~
sysadmin@centos01 ~> hostname && cat /proc/cpuinfo | grep "cpu cores" && free -h && df -h && ip a
centos01
cpu cores : 1
total used free shared buff/cache available
Mem: 1.7Gi 1.0Gi 152Mi 18Mi 683Mi 666Mi
Swap: 4.0Gi 0B 4.0Gi
Filesystem Size Used Avail Use% Mounted on
devtmpfs 4.0M 0 4.0M 0% /dev
tmpfs 854M 0 854M 0% /dev/shm
tmpfs 342M 5.5M 337M 2% /run
/dev/sda2 10G 4.4G 5.6G 45% /
/dev/sda3 5.0G 469M 4.5G 10% /var
/dev/sda1 2.0G 375M 1.6G 19% /boot
tmpfs 171M 100K 171M 1% /run/user/1000
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
inet 127.0.0.1/8 scope host lo
valid_lft forever preferred_lft forever
inet6 ::1/128 scope host
valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
link/ether 08:00:27:30:ac:19 brd ff:ff:ff:ff:ff:ff
inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic noprefixroute enp0s3
valid_lft 86169sec preferred_lft 86169sec
inet6 fe80::a00:27ff:fe30:ac19/64 scope link noprefixroute
valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
link/ether 08:00:27:ea:8f:22 brd ff:ff:ff:ff:ff:ff
inet 192.168.1.30/24 brd 192.168.1.255 scope global noprefixroute enp0s8
valid_lft forever preferred_lft forever
inet6 fe80::7372:c785:53e0:3f34/64 scope link noprefixroute
valid_lft forever preferred_lft forever

SSH-Key


sysadmin@Bastion ~/obligatorio-local> ssh-copy-id 192.168.1.30
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/sysadmin/.ssh/id_ed25519.pub"
The authenticity of host '192.168.1.30 (192.168.1.30)' can't be established.
ED25519 key fingerprint is SHA256:UliKtRL/rhYrG3nvWTpmf2oyaLv8z4nE06GXrKhSQLQ.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
/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@192.168.1.30's password:
Number of key(s) added: 1
Now try logging into the machine, with: "ssh '192.168.1.30'"
and check to make sure that only the key(s) you wanted were added.
sysadmin@Bastion ~/obligatorio-local>

Inventario

Inventario.ini

```
root@Bastion ~/obligatorio (master)# ls
README.md  ansible_basics.txt  hardening.yml  inventario.ini  nfs_setup.yml  pruebaAnsible-list.txt
root@Bastion ~/obligatorio (master)# cat inventario.ini
ub00 ansible_host=192.168.1.10 ansible_user=sysadmin
ub01 ansible_host=192.168.1.20 ansible_user=sysadmin
centos01 ansible_host=192.168.1.30 ansible_user=sysadmin

[ubuntu]
ub00
ub01

[ubuntu:vars]
ssh_service=ssh
apache_service=apache2

[centos]
centos01

[centos:vars]
ssh_service=sshd
apache_service=httpd

[workstation]
ub00

[webserver]
centos01

[hardening]
ub01

[linux:children]
ubuntu
centos
root@Bastion ~/obligatorio (master)#
```

Pruebas

Ansible-inventory –i inventario.ini --list

```
{
    "_meta": {
        "hostvars": {
            "centos01": {
                "ansible_host": "192.168.1.30",
                "ansible_user": "sysadmin",
                "apache_service": "httpd",
                "ssh_service": "sshd"
            },
            "ub00": {
                "ansible_host": "192.168.1.10",
                "ansible_user": "sysadmin",
                "apache_service": "apache2",
                "ssh_service": "ssh"
            },
            "ub01": {
                "ansible_host": "192.168.1.20",
                "ansible_user": "sysadmin",
                "apache_service": "apache2",
                "ssh_service": "ssh"
            }
        },
        "profile": "inventory_legacy"
    },
    "all": {
        "children": [
            "ungrouped",
            "workstation",
            "webserver",
            "hardening",
            "linux"
        ]
    },
    "centos": {
        "hosts": [
            "centos01"
        ]
    },
    "hardening": {
        "hosts": [
            "ub01"
        ]
    },
    "linux": {
        "children": [
            "ubuntu",
            "centos"
        ]
    },
}
```


Ansible –i inevntario.ini all -m ping


```
sysadmin@Bastion ~/obligatorio-local> ansible -i inventario.ini all -m ping
[WARNING]: Host 'ub01' is using the discovered Python interpreter at '/usr/bin/python3.12', but future installation of another Python interpreter could cause a
different interpreter to be discovered. See https://docs.ansible.com/ansible-core/2.19/reference_appendices/interpreter_discovery.html for more information.
ub01 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3.12"
    },
    "changed": false,
    "ping": "pong"
}
[WARNING]: Host 'centos01' is using the discovered Python interpreter at '/usr/bin/python3.9', but future installation of another Python interpreter could cause
 a different interpreter to be discovered. See https://docs.ansible.com/ansible-core/2.19/reference_appendices/interpreter_discovery.html for more information.
centos01 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3.9"
    },
    "changed": false,
    "ping": "pong"
}
sysadmin@Bastion ~/obligatorio-local> _
```


Comandos ad-hoc

Listar usuarios (Ubuntu)

Comando: ansible –i inventario.ini ubuntu –m shell –a “cat /etc/passwd | cut –d: -f1”

```
sysadmin@Bastion ~/obligatorio-local> ansible -i inventario.ini ubuntu -m shell -a "cat /etc/passwd | cut -d: -f1"
[WARNING]: Host 'ub01' is using the discovered Python interpreter at '/usr/bin/python3.12', but future installation of another Python interpreter could cause a
different interpreter to be discovered. See https://docs.ansible.com/ansible-core/2.19/reference_appendices/interpreter_discovery.html for more information.
ub01 | CHANGED | rc=0 >>
root
daemon
bin
sys
sync
games
man
lp
mail
news
uucp
proxy
www-data
backup
list
irc
_apt
nobody
systemd-network
systemd-timesync
dhcpcd
messagebus
systemd-resolve
pollinate
polkitd
usbmux
sshd
sysadmin
sysadmin@Bastion ~/obligatorio-local> _
```

Uso de memoria

Ansible –i inventario.ini all –m shell -a “df -h”

```
sysadmin@Bastion ~/obligatorio-local> ansible -i inventario.ini all -m shell -a "df -h"
[WARNING]: Host 'ub01' is using the discovered Python interpreter at '/usr/bin/python3.12', but future installation of another Python interpreter could cause a
different interpreter to be discovered. See https://docs.ansible.com/ansible-core/2.19/reference_appendices/interpreter_discovery.html for more information.
ub01 | CHANGED | rc=0 >>
Filesystem      Size  Used Avail Use% Mounted on
tmpfs           197M  768K  197M   1% /run
/dev/sda2       9.8G  1.8G  7.5G  19% /
tmpfs           985M     0  985M   0% /dev/shm
tmpfs           5.0M     0  5.0M   0% /run/lock
/dev/sda3       2.0G   99M  1.7G   6% /boot
/dev/sda4       4.9G  373M  4.3G   8% /var
tmpfs           197M   12K  197M   1% /run/user/1000
[WARNING]: Host 'centos01' is using the discovered Python interpreter at '/usr/bin/python3.9', but future installation of another Python interpreter could cause
 a different interpreter to be discovered. See https://docs.ansible.com/ansible-core/2.19/reference_appendices/interpreter_discovery.html for more information.
centos01 | CHANGED | rc=0 >>
Filesystem      Size  Used Avail Use% Mounted on
devtmpfs        4.0M     0  4.0M   0% /dev
tmpfs           854M     0  854M   0% /dev/shm
tmpfs           342M  5.5M  337M   2% /run
/dev/sda2        10G  4.4G  5.6G  45% /
/dev/sda3       5.0G  469M  4.5G  10% /var
/dev/sda1       2.0G  375M  1.6G  19% /boot
tmpfs           171M  104K  171M   1% /run/user/1000
sysadmin@Bastion ~/obligatorio-local> _
```

Chorny (Instalacion y servicio)

Ansible –i inventario.ini centos –m dnf –a “name=chrony state=latest” --become

Ansible –i inventario.ini centos –m shell –a “systemctl status chronyd”

```
sysadmin@Bastion ~/obligatorio-local> ansible -i inventario.ini centos -m dnf -a "name=chrony state=latest" --become
[WARNING]: Host 'centos01' is using the discovered Python interpreter at '/usr/bin/python3.9', but future installation of another Python interpreter could cause
 a different interpreter to be discovered. See https://docs.ansible.com/ansible-core/2.19/reference_appendices/interpreter_discovery.html for more information.
centos01 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3.9"
    },
    "changed": false,
    "msg": "Nothing to do",
    "rc": 0,
    "results": []
}
sysadmin@Bastion ~/obligatorio-local> ansible -i inventario.ini centos -m shell -a "systemctl status chronyd"
[WARNING]: Host 'centos01' is using the discovered Python interpreter at '/usr/bin/python3.9', but future installation of another Python interpreter could cause
 a different interpreter to be discovered. See https://docs.ansible.com/ansible-core/2.19/reference_appendices/interpreter_discovery.html for more information.
centos01 | CHANGED | rc=0 >>
● chronyd.service - NTP client/server
     Loaded: loaded (/usr/lib/systemd/system/chronyd.service; enabled; preset: enabled)
     Active: active (running) since Sun 2025-08-10 13:53:57 -03; 2h 51min ago
       Docs: man:chronyd(8)
             man:chrony.conf(5)
    Process: 614 ExecStart=/usr/sbin/chronyd $OPTIONS (code=exited, status=0/SUCCESS)
   Main PID: 644 (chronyd)
      Tasks: 1 (limit: 10526)
     Memory: 4.0M
        CPU: 129ms
     CGroup: /system.slice/chronyd.service
             └─644 /usr/sbin/chronyd -F 2

Aug 10 14:33:00 centos01 chronyd[644]: Jitter of 190.64.134.53 (2.centos.pool.ntp.org) exceeds maxjitter of 1.000 seconds
Aug 10 14:33:01 centos01 chronyd[644]: Selected source 164.73.232.34 (2.centos.pool.ntp.org)
Aug 10 14:33:01 centos01 chronyd[644]: System clock wrong by 172.756071 seconds
Aug 10 14:34:01 centos01 chronyd[644]: Root distance of 190.64.134.53 (2.centos.pool.ntp.org) exceeds maxdistance of 3.000 seconds
Aug 10 14:34:01 centos01 chronyd[644]: Root distance of 164.73.225.4 (2.centos.pool.ntp.org) exceeds maxdistance of 3.000 seconds
Aug 10 14:34:01 centos01 chronyd[644]: Jitter of 200.40.115.74 (2.centos.pool.ntp.org) exceeds maxjitter of 1.000 seconds
Aug 10 14:34:02 centos01 chronyd[644]: Jitter of 164.73.225.4 (2.centos.pool.ntp.org) exceeds maxjitter of 1.000 seconds
Aug 10 14:34:04 centos01 chronyd[644]: Jitter of 190.64.134.53 (2.centos.pool.ntp.org) exceeds maxjitter of 1.000 seconds
Aug 10 14:35:06 centos01 chronyd[644]: Selected source 200.40.115.74 (2.centos.pool.ntp.org)
Aug 10 14:35:07 centos01 chronyd[644]: Root distance of 190.64.134.53 (2.centos.pool.ntp.org) exceeds maxdistance of 3.000 seconds
sysadmin@Bastion ~/obligatorio-local>
```

Playbook

Nfs_setup.yml

```
 GNU nano 7.2                                                nfs_setup.yml
- name: NFS
  hosts: centos
  become: true
  vars:
    directorio: "/var/nfs_shared"
    opciones: "192.168.1.30/24(rw,sync,root_squash)"
    permisos: '0777'
    #creamos variables que se usaran para mejorar la implementacion

  tasks:

    - name: instalacion de NFS y actualizacion
      ansible.builtin.dnf:
        name:
          - nfs-utils
        state: latest
        #instalamos NFS y si este ya esta instalado es actualizado


    - name: creacion de directorio NFS o probar que existe
      ansible.builtin.file:
        path: "{{ directorio }}"
        state: directory
        mode: "{{ permisos }}"
        owner: nobody
        group: nobody
        #se crear el directorio y se lo adjudica a nobody y se le da permisos


    - name: compartir en /etc/exports
      ansible.builtin.lineinfile:
        path: /etc/exports
        state: present
        line: '{{ directorio }} {{ opciones }}'
      notify: reinicio NFS server
      #creamos a export para darle las opciones al directorio a compartir y mencionamos cambios al handler


    - name: compartir export
      ansible.builtin.command: "exportfs -rav"
      #porque no encotramos modulo para mantener tabla de directorios compartidos lo harmos por comando
      #r actualizara /var/lib/nfs/etab con el contenido de /etc/exports
      #a exporta directorios
      #v explica vervosamente, por si susede un error

    - name: permitir en firewall
```

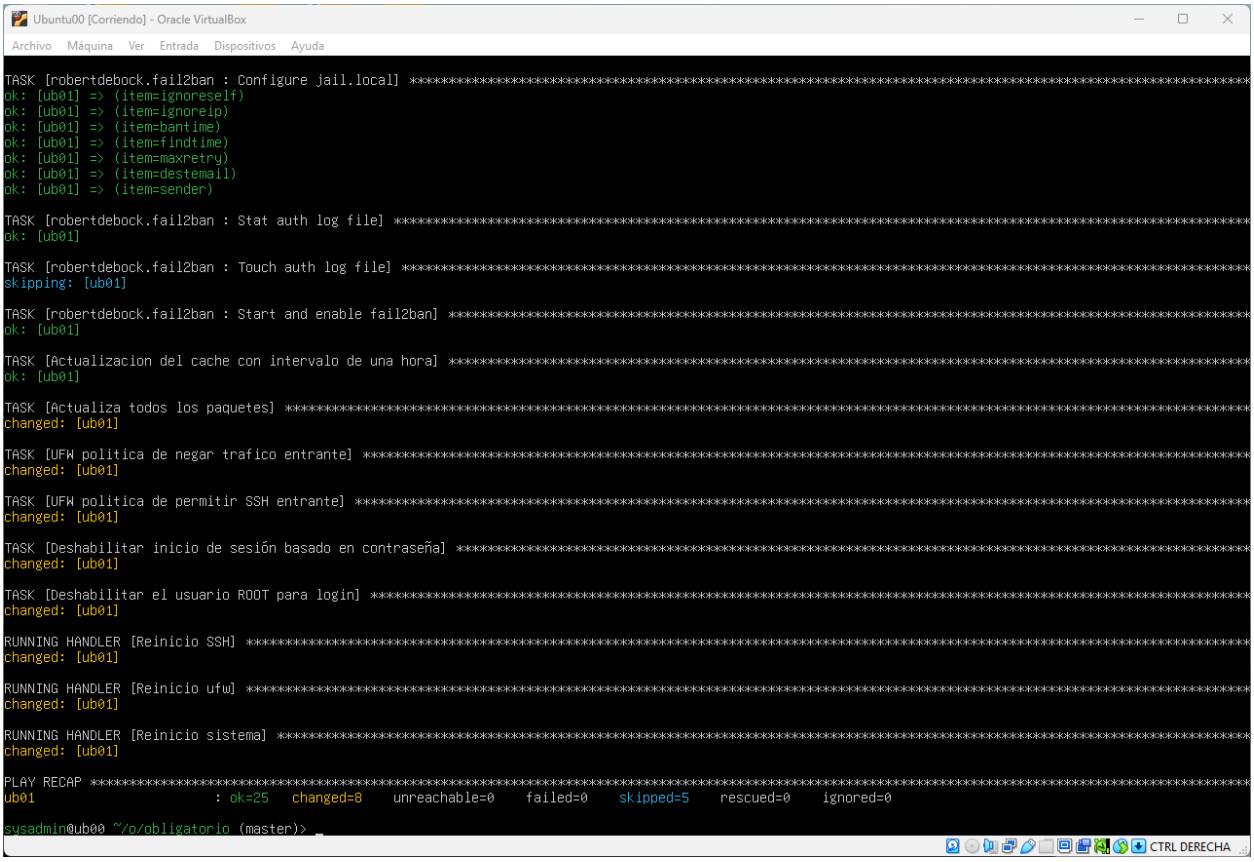
Ansible –i inventario.ini nfs_setup.yml

```
sysadmin@Bastion ~/obligatorio> ansible-playbook -i inventario.ini nfs_setup.yml

PLAY [NFS] *********************************************************************************************************************************************************************************

TASK [Gathering Facts] *********************************************************************************************************************************************************************
[WARNING]: Host 'centos01' is using the discovered Python interpreter at '/usr/bin/python3.9', but future installation of another Python interpreter could cause
 a different interpreter to be discovered. See https://docs.ansible.com/ansible-core/2.19/reference_appendices/interpreter_discovery.html for more information.
ok: [centos01]

TASK [instalacion de NFS y actualizacion] **************************************************************************************************************************************************
changed: [centos01]

TASK [creacion de directorio NFS o probar que existe] **************************************************************************************************************************************
changed: [centos01]

TASK [compartir en /etc/exports] ***********************************************************************************************************************************************************
changed: [centos01]

TASK [compartir export] ********************************************************************************************************************************************************************
changed: [centos01]

TASK [permitir en firewall] ****************************************************************************************************************************************************************
changed: [centos01] => (item=nfs)
changed: [centos01] => (item=mountd)

RUNNING HANDLER [reinicio NFS server] ******************************************************************************************************************************************************
changed: [centos01]

PLAY RECAP *********************************************************************************************************************************************************************************
centos01                   : ok=7    changed=6    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

sysadmin@Bastion ~/obligatorio>
```

Hardening.yml

```
  GNU nano 7.2                                                  hardening.yml
- name: Hardening
  hosts: hardening
  user: "{{ ansible_user }}"
  become: yes
  gather_facts: yes

  vars:
    directorio: "/etc/fail2ban"
    permisos: '0644'
    #Definimos variables a usar

  tasks:

    - name: Actualizacion del cache con intervalo de una hora
      ansible.builtin.apt:
        update_cache: yes
        cache_valid_time: 3600
      #Actualizo cache y que este no se actualize por una hora


    - name: Actualiza todos los paquetes
      ansible.builtin.apt:
       name: "*"
       state: latest
      notify: Reinicio sistema
      #Actualizo paquetes y reinicio


    - name: Instalacion de fail2ban
      ansible.builtin.apt:
        name: fail2ban
        state: latest
        #Instalamos a fail2ban

    - name: Copiar el archivo de configuracion de fail2ban
      ansible.builtin.copy:
        src: "{{ directorio }}/fail2ban.conf"
        dest: "{{ directorio }}/fail2ban.local"
        remote_src: yes
        owner: root
        group: root
        mode: "{{ permisos }}"
        #Copiamos el archivo de configuracion ya que este se posible que se cambiara al actualizarse


    - name: Deshabilitar inicio de sesión basado en contraseña
      ansible.builtin.lineinfile:
```

Ansible –i inventario.ini hardening.yml



Preguntas

¿Qué es Ansible?

Ansible es una tecnología útil para la automatización de configuraciones en entornos de infraestructura, haciendo que este proceso sea mucho mas eficiente. Permite gestionar equipos remotos centralizándolos desde un bastión central. La ventaja de emplear este sistema radica en la posibilidad que brinda de administrar varios equipos a la vez de forma generalizada.

Ansible cuenta con ciertos componentes claves, tales como:

- Inventario

- Playbooks
- Módulos
- Roles

Esta tecnología es compatible con entornos de **Linux** y **Windows**.

Funciones:

- Automatización de configuraciones
- Ejecución de comandos ad-hoc
- Creación de ambientes específicos y optimizados

¿Que es un playbook de Ansible?

Los playbooks de ansible son archivos con extension **.yml** los cuales contienen instrucciones a modo de pseudocódigo que serán ejecutadas por el programa. Estas tareas definen las configuraciones que serán llevadas a cabo en los servidores que están siendo administrados bajo el control del bastión.

Dentro de los playbooks se incorporan varios de los elementos que componen a la herramienta; por ejemplo se define el inventario sobre el que va a actuar y los módulos que va a emplear para hacer la gestión sobre los hosts.

¿Que información contiene un inventario de Ansible?

Dentro de un inventario de ansible se encuentra todo el ecosistema sobre el que la herramienta es capaz de actuar, es posible detallar nombres de hosts asilados o formar grupos de equipos para estandarizar configuraciones; también se pueden establecer sub grupos, que son grupos dentro de otros grupos, y definir variables que van a actuar en ciertos sistemas.

Funciones:

- Organizar
- Ejecutar tareas en ciertos hosts en concreto

- Agrupar y generalizar configuraciones

¿Que es un módulo de ansible?

En ansible, los módulos son las herramientas a través de las cuales esta tecnología es capaz de interactuar con los equipos que gestiona. Existen varios módulos para llevar a cabo diferentes tareas especificas; algunos de ellos son propios de la aplicación, es decir, **BuiltIn**; mientras que otros son externos y deben ser integrados de forma manual. A menudo estos módulos externos son creados por la comunidad para suplir las necesidades de realizar ciertas tareas concretas las cuales no pueden ser ejecutadas mediante los módulos autóctonos de Ansible.

Es posible añadir módulos a ansible mediante el uso del componente **ansible-galaxy**. Estos módulos pueden ser hechos en **bash** o **python** y se distribuyen a treves de colecciones.

Módulos **BuiltIn**:

- Apt/dnf-yum
- shell
- Lineinfile

Módulos externos:

- Community.general.ufw
- Community.mysql.mysql_user
- Azure.collection.azure_rm_vm

¿Que ventajas tiene Ansible sobre otros métodos de automatización?

- Es **agent-less**, no es necesario instalar componentes en los equipos que van a ser gestionados por la herramienta; únicamente deben contar con **SSH**.
- Es escalable, permite la adaptación frente a el crecimiento de la infraestructura.

- Es compatible con **GIT**, lo que ayuda a la creación de configuraciones en grupo.
- Cuenta con una comunidad activa sobre la cual es posible apoyarse preguntando y colaborando sobre diferentes temas y proyectos.
- Esta soportado por **Red Hat**.