

Obligatorio

Taller de servidores Linux – Agosto 2025

Michael Candia N.º 305769

Inti López N.º 343404

Grupo: N3.5A

Docente: Enrique Verdes

Índice

Índice.....	1
1 - Declaración de autoría	2
2 – Configuración de disco de Servidores CentOS y Ubuntu.....	3
3 – Tarea 5: Responder las siguientes preguntas	4
4 – Comandos Tarea 2:	5
5 – Comandos ad-hoc.....	6
6 – Ejecuciones de Playbooks	7
7 – Desafíos encontrados	9
8 – Bibliografía.	9

1 - Declaración de autoría

Nosotros, Inti López y Michael Candia, declaramos que el trabajo que se presenta en esa obra es de nuestra propia mano. Podemos asegurar que:

- La obra fue producida en su totalidad mientras realizábamos la materia Taller de Servidores Linux;
- Cuando hemos consultado el trabajo publicado por otros, lo hemos atribuido con claridad;
- Cuando hemos citado obras de otros, hemos indicado las fuentes. Con excepción de estas citas, la obra es enteramente nuestra;
- En la obra, hemos acusado recibo de las ayudas recibidas;
- Cuando la obra se basa en trabajo realizado conjuntamente con otros, hemos explicado claramente qué fue contribuido por otros, y qué fue contribuido por nosotros;
- Ninguna parte de este trabajo ha sido publicada previamente a su entrega, excepto donde se han realizado las aclaraciones correspondientes.

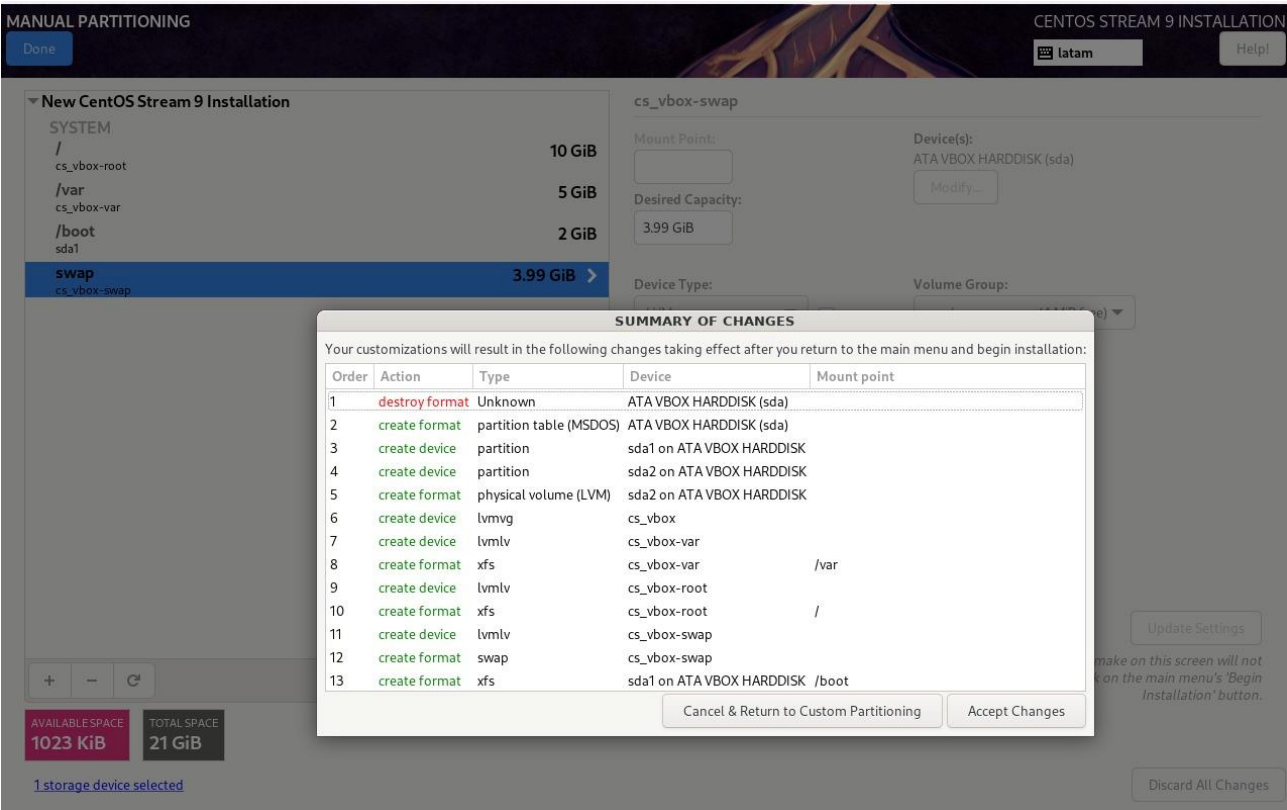


Inti López
N° 343404



Michael Candia
N° 305769

2 – Configuración de disco de Servidores CentOS y Ubuntu



Dirección IP servidor Ubuntu: 192.168.1.3/24
Dirección IP servidor CentOS: 192.168.1.2/24

3 – Tarea 5: Responder las siguientes preguntas

1. ¿Qué es Ansible? Mencione dos actividades que se puedan hacer con Ansible

Es una herramienta que se utiliza para automatizar procesos informáticos como la preparación de infraestructura, gestión de configuración, implementación de las aplicaciones y organización de sistemas.

2. ¿Qué es un playbook de Ansible?

Es un paso a paso que indica que tareas realizar y en que dispositivo realizarlas, se utiliza para automatizar trabajos que generalmente tomarían mucho tiempo si se hiciera individualmente como, por ejemplo, instalación de programas, configuraciones de equipos, etc, con este método Ansible ejecuta paso a paso las tareas en donde sea necesario de forma automática pudiéndose guardar y reutilizar las mismas.

3. ¿Qué información contiene un inventario de Ansible?

Contiene la información de los host en los que se pueden realizar tareas permitiéndole a ansible saber a qué equipo conectarse, organizarlos en grupos, aplicar configuraciones específicas y automatizar tareas.

4. Explique que es un módulo de Ansible y dé un ejemplo.

Es un script o programa que realiza una tarea específica en los dispositivos que se ejecuta como instalar software, configurar servicios, etc.

Ejemplo:

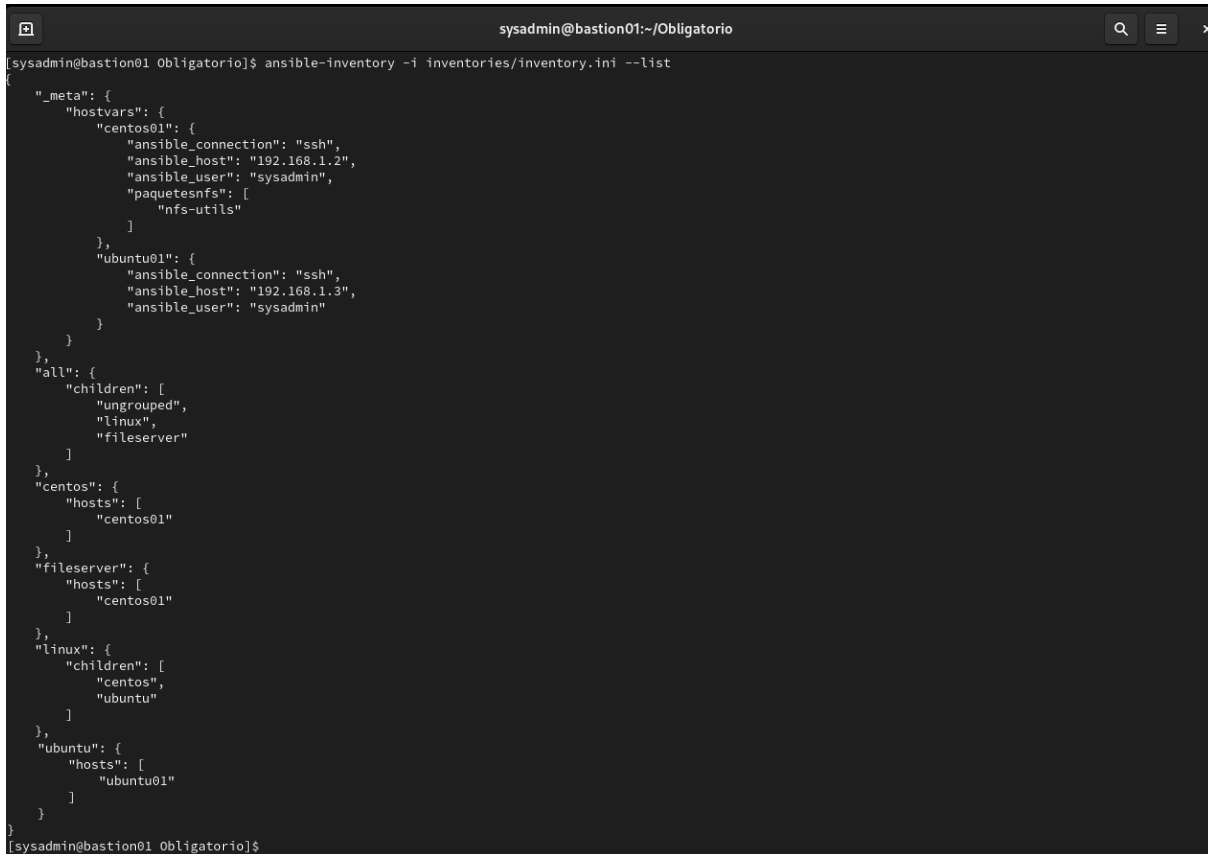
ansible.builtin apt: se utiliza para instalar paquetes como httpd, etc.

5. ¿Qué ventajas tiene Ansible sobre otros métodos de automatización?

- Permite realizar tareas complejas de forma sencilla, eficiente y rápida.
- Utiliza yaml, que es un lenguaje sencillo y legible para el usuario.
- Posee documentación con muchos módulos que permiten automatizar diversas tareas.
- Utiliza conexiones SSH permitiendo conectarse de forma remota desde otro equipo minimizando el riesgo de seguridad con métodos de autenticación específicos.

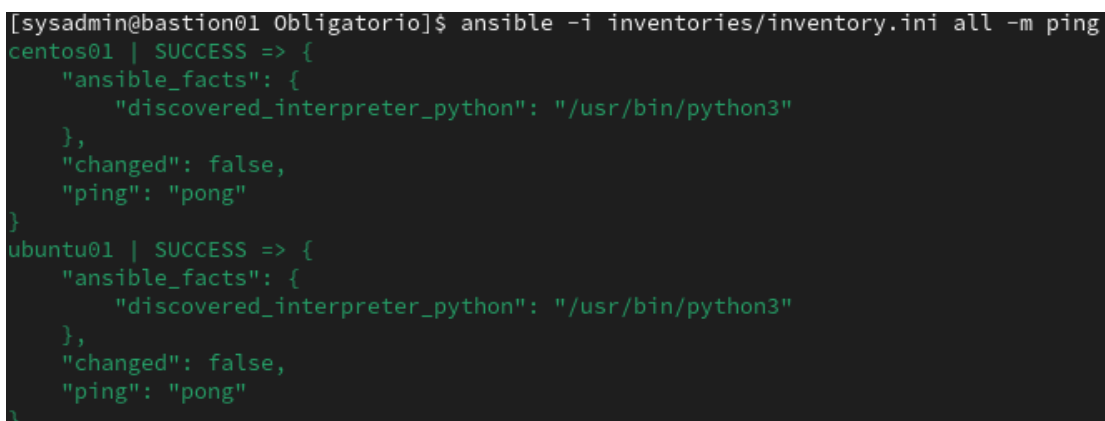
4 – Comandos Tarea 2:

`ansible-inventory -i inventories/inventory.ini --list`

A terminal window titled 'sysadmin@bastion01:~/Obligatorio' showing the command 'ansible-inventory -i inventories/inventory.ini --list' and its output. The output is a JSON structure representing the inventory. It includes a '_meta' section with 'hostvars' for 'centos01' and 'ubuntu01', an 'all' section with 'children' (ungrouped, linux, fileserver), and individual sections for 'centos', 'fileserver', 'linux', and 'ubuntu', each containing a list of hosts. The terminal shows the command being executed and the resulting JSON output.

```
[sysadmin@bastion01 Obligatorio]$ ansible-inventory -i inventories/inventory.ini --list
{
  "_meta": {
    "hostvars": {
      "centos01": {
        "ansible_connection": "ssh",
        "ansible_host": "192.168.1.2",
        "ansible_user": "sysadmin",
        "paquetes nfs": [
          "nfs-utils"
        ]
      },
      "ubuntu01": {
        "ansible_connection": "ssh",
        "ansible_host": "192.168.1.3",
        "ansible_user": "sysadmin"
      }
    }
  },
  "all": {
    "children": [
      "ungrouped",
      "linux",
      "fileserver"
    ]
  },
  "centos": {
    "hosts": [
      "centos01"
    ]
  },
  "fileserver": {
    "hosts": [
      "centos01"
    ]
  },
  "linux": {
    "children": [
      "centos",
      "ubuntu"
    ]
  },
  "ubuntu": {
    "hosts": [
      "ubuntu01"
    ]
  }
}
```

`ansible -i inventories/inventory.ini all -m ping`

A terminal window showing the command 'ansible -i inventories/inventory.ini all -m ping' and its output. The output shows the command being executed and the resulting JSON output for each host. The output is a JSON structure representing the results of the ping command. It includes a 'centos01' section with 'SUCCESS' and 'ping' result, and an 'ubuntu01' section with 'SUCCESS' and 'ping' result. The terminal shows the command being executed and the resulting JSON output.

```
[sysadmin@bastion01 Obligatorio]$ ansible -i inventories/inventory.ini all -m ping
centos01 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3"
  },
  "changed": false,
  "ping": "pong"
}
ubuntu01 | SUCCESS => {
  "ansible_facts": {
    "discovered_interpreter_python": "/usr/bin/python3"
  },
  "changed": false,
  "ping": "pong"
}
```

5 – Comandos ad-hoc

- 1- Listar todos los usuarios en servidor Ubuntu:

```
ansible ubuntu -m command -a "cut -d: -f1 /etc/passwd"
```

```
[sysadmin@bastion01 Obligatorio]$ ansible ubuntu -m command -a "cut -d: -f1 /etc/passwd"
ubuntu01 | 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
syslog
uuid
tcpdump
tss
landscape
fwupd-refresh
usbmux
sshd
sysadmin
```

- 2- Mostrar el uso de memoria en todos los servidores:

```
ansible all -m command -a "free -h"
```

```
[sysadmin@bastion01 Obligatorio]$ ansible all -m command -a "free -h"
centos01 | CHANGED | rc=0 >>
      total        used        free      shared  buff/cache   available
Mem:    1.7Gi       368Mi     1.1Gi       9.0Mi       346Mi       1.3Gi
Swap:   4.0Gi          0B       4.0Gi
ubuntu01 | CHANGED | rc=0 >>
      total        used        free      shared  buff/cache   available
Mem:    1.9Gi       342Mi     1.5Gi       1.1Mi       206Mi       1.6Gi
Swap:   4.0Gi          0B       4.0Gi
```

3- Que el servicio chrony ese instalado y funcionando en servidor Centos

ansible centos -m shell -a "dnf install -y chrony && systemctl enable --now chronyd" --become --ask-become-pass

```
[sysadmin@bastion01 Obligatorio]$ ansible centos -m shell -a "dnf install -y chrony && systemctl enable --now chronyd" --become --ask-become-pass
BECOME password:
centos01 | CHANGED | rc=0 >>
Last metadata expiration check: 5:36:17 ago on Mon 11 Aug 2025 08:59:58 PM -03.
Package chrony-4.6.1-2.el9.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
[sysadmin@bastion01 Obligatorio]$
```

6 – Ejecuciones de Playbooks

1- ansible-playbook playbooks/nfs_shared.yml -K

```
[sysadmin@bastion01 Obligatorio]$ ansible-playbook playbooks/nfs_shared.yml -K
BECOME password:

PLAY [Intalacion y configuracion de NFS] *****

TASK [Gathering Facts] *****
ok: [centos01]

TASK [Instalar paquetes NFS] *****
ok: [centos01]

TASK [Servicio NFS iniciado y enabled] *****
ok: [centos01]

TASK [Permitir conexiones al puerto 2049] *****
ok: [centos01]

TASK [Exista el directorio en /var] *****
ok: [centos01]

TASK [Declarar carpeta compartida en export] *****
ok: [centos01]

TASK [Recargar export "NFS"] *****
changed: [centos01]

PLAY RECAP *****
centos01 : ok=7 changed=1 unreachable=0 failed=0 skipped=0 rescued=0 ignore
d=0

[sysadmin@bastion01 Obligatorio]$
```


2- ansible-playbook playbooks/hardening.yml -K

```
[sysadmin@vbox Obligatorio]$ ansible-playbook playbooks/hardening.yml -K
BECOME password:

PLAY [ubuntu] *****

TASK [Gathering Facts] *****
ok: [ubuntu01]

TASK [Actualizador de paquetes] *****
ok: [ubuntu01]

TASK [Habilitar ufw] *****
ok: [ubuntu01]

TASK [Permite ssh] *****
ok: [ubuntu01]

TASK [No permitir entrada] *****
ok: [ubuntu01]

TASK [Deshabilitar root login] *****
ok: [ubuntu01]

TASK [Eliminar archivo PasswordAuth] *****
ok: [ubuntu01]

TASK [Deshabilitar login por contraseña] *****
ok: [ubuntu01]

TASK [Habilitar ingreso por clave Publica] *****
ok: [ubuntu01]

TASK [Instalar fail2ban] *****
ok: [ubuntu01]

TASK [Configurar archivo de bloqueo ssh] *****
changed: [ubuntu01]

RUNNING HANDLER [Reiniciar Fail2ban] *****
changed: [ubuntu01]

PLAY RECAP *****
ubuntu01 : ok=12  changed=2  unreachable=0  failed=0  skipped=0  rescued=0  ignored=0

[sysadmin@vbox Obligatorio]$
```

7 – Desafíos encontrados

Durante el desarrollo de las actividades se presentaron ciertas dificultades, como el poco tiempo de realización del trabajo debiendo organizar nuestros horarios de rutina diaria para poder cumplir con los requerimientos, otras dificultades surgieron en la realización de las tareas y resolución de errores, donde tuvimos que enfrentarlas investigando en la documentación de Ansible. Estos desafíos permitieron fortalecer nuestro aprendizaje para lograr cumplir con todos los requerimientos.

8 – Bibliografía.

1- Documentación de Ansible para módulos:

All modules — Ansible Documentation. (s. f.).

https://docs.ansible.com/ansible/2.9/modules/list_of_all_modules.html

2- Material de Clase:

Entrar al sitio / Aulas. (s. f.). <https://aulas.ort.edu.uy/>