OBLIGATORIO taller de administracion de servidores linux

Trabajo para entregar obligatorio de forma online

**IUniversidad Ort**

**Carrera Analista en Infraestructura informática**

**Materia Taller de Servidores Linux**

**Integrantes del grupo: Mauro Diaz** – **N° de estudiante 170909**

**Mattías Sánchez 230382**

INDICE

[2 – Configurar un archivo de inventario de Ansible 3](#_Toc205684793)

[Crear un archivo de inventario (inventory.ini) con los grupos Ubuntu y Centos y al menos un host por grupo 3](#_Toc205684794)

[Debe haber un grupo Linux con los subgrupos Ubuntu y Centos y un grupo Webserver, con el servidor Centos 3](#_Toc205684795)

[Una variable definida ansible\_user para acceso SSH 3](#_Toc205684796)

[En caso de necesitar una variable ansible\_host para cada uno con la IP 3](#_Toc205684797)

[Proporcionar una captura de prueba de pantalla de una conexión exitosa usando los comandos: 3](#_Toc205684798)

[$ ansible -inventory - inventory.ini - - list 3](#_Toc205684799)

[$ ansible all -I inventory.ini -m ping 3](#_Toc205684800)

[3- Ejecutar comandos ad-doc 3](#_Toc205684801)

[Ejecutar y documentar los siguientes comandos ad-doc: 4](#_Toc205684802)

[Listar todos los usuarios en servidor Ubuntu 4](#_Toc205684803)

[Mostrar el uso de memoria en todos los servidores 4](#_Toc205684804)

[Que el servicio chrony este instalado y funcionando en Servidor Centos 4](#_Toc205684805)

[4- Crear y ejecutar playbook de Ansible 4](#_Toc205684806)

[Crea un playbook (nfs\_setup.yml) para CentOS que permita obtener el siguiente resultado: 5](#_Toc205684807)

[El servidor NFS este instalado 5](#_Toc205684808)

[Se asegure que el servicio NFS este iniciado y funcionando 5](#_Toc205684809)

[- El firewall permita conexiones al puerto 2049 5](#_Toc205684810)

[Exista el directorio / var/nfs\_shared que pertenece al usuario/ grupo nobody/ nobody y tiene permisos 777 5](#_Toc205684811)

[El directorio esta compartido por NFS 5](#_Toc205684812)

[Debe haber un handler que actualice y relea el archivo / etc/exports si este cambia bloqueando todo el trafico entrante y que permita solo SSH 5](#_Toc205684813)

[Crear un playbok (hardening.yml) que se va a ejecutar en los servidores Ubuntu que realice lo siguiente: 5](#_Toc205684814)

[Actualizar todos los paquetes 5](#_Toc205684815)

[-Que este habilitado ufw bloqueando todo el trafico entrante y permitiendo solo SSH 5](#_Toc205684816)

[-Que solo se pueda hacer login con clave publica y que root no pueda hacer login 5](#_Toc205684817)

[-Que este instaldo fail2ban y bloquee intentos fallidos de conexión SSH. El servicio debe quedar habilitado y activado. 5](#_Toc205684818)

[- Debe haber un handler que reinicie el sistema si se actualizan paquetes 5](#_Toc205684819)

[- Debe haber un handler que reinicie SSH si cambia la configuración 5](#_Toc205684820)

[5- Responder las siguientes preguntas 5](#_Toc205684821)

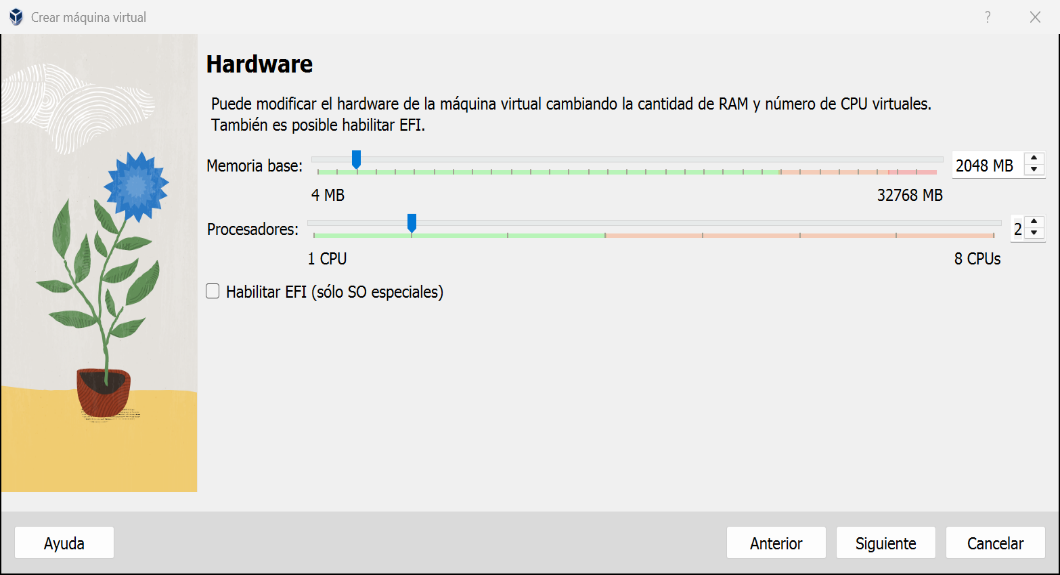
[¿ Que es Ansible? Mencione dos actividades que se puedan hacer con Ansible 6](#_Toc205684822)

[¿ Que es un playbook de Ansible? 6](#_Toc205684823)

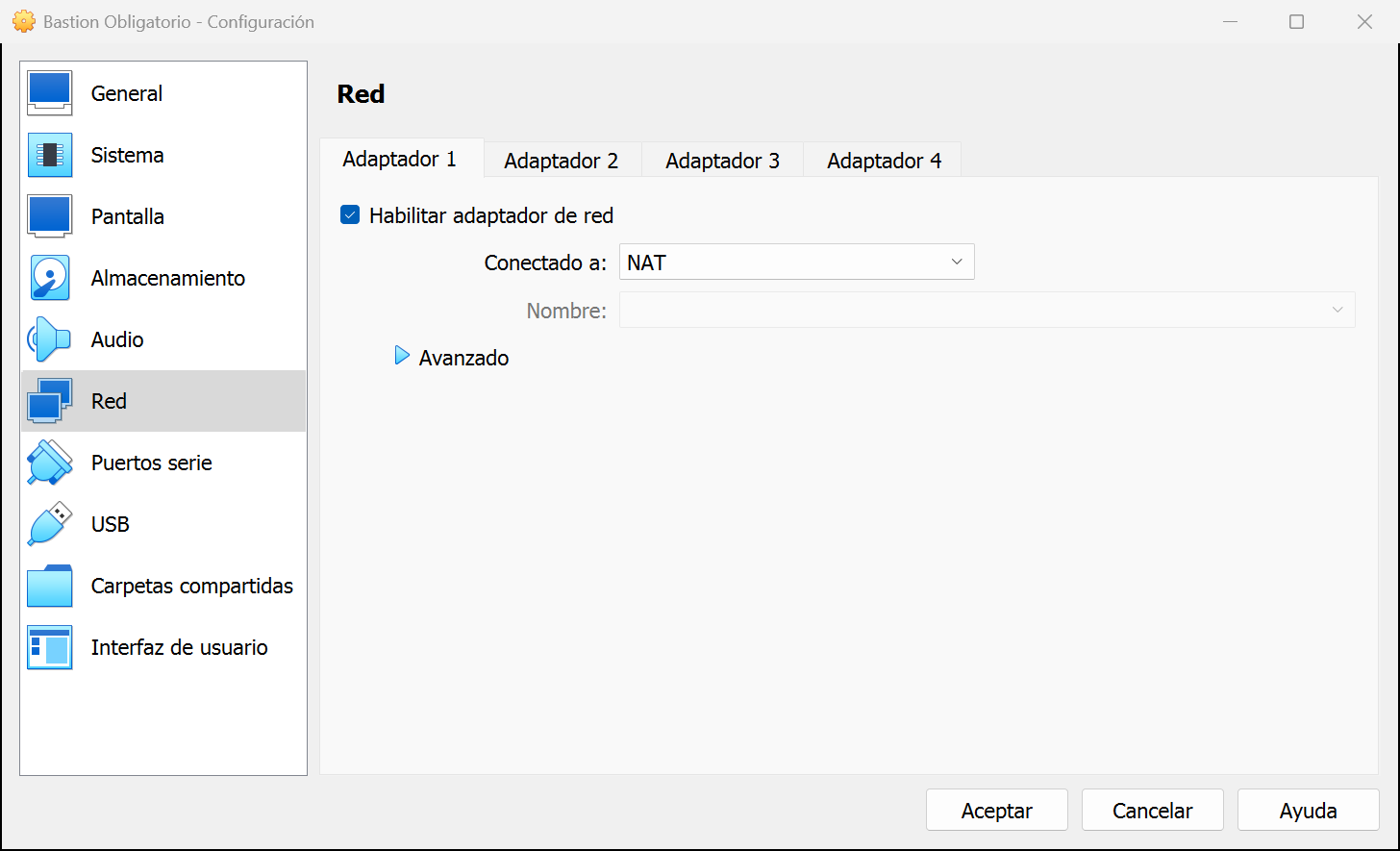
[¿ Que información contiene un inventario de Ansible? 6](#_Toc205684824)

[Explique que es un modulo de Ansible y de un ejemplo 6](#_Toc205684825)

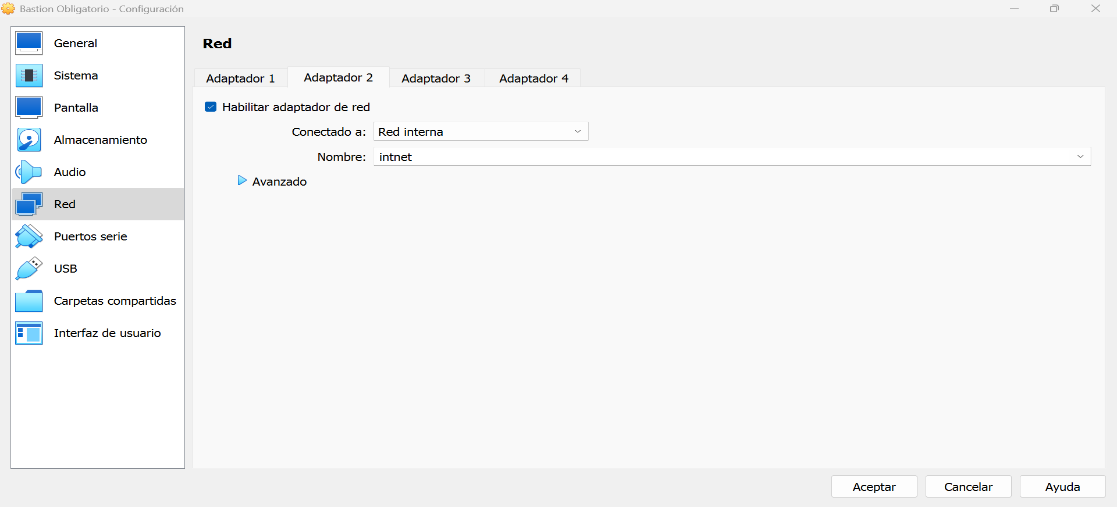
[¿ Que ventajas tiene Ansible sobre otros métodos de atomatizacion? 6](#_Toc205684826)

* 

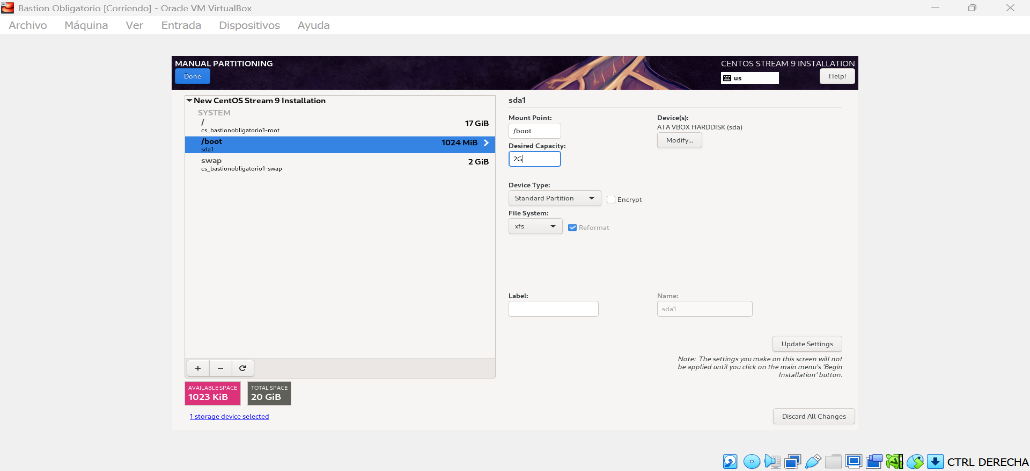
Se crea bastión obligatorio en CentOS Stream como server con 2 GB de ram y una CPU



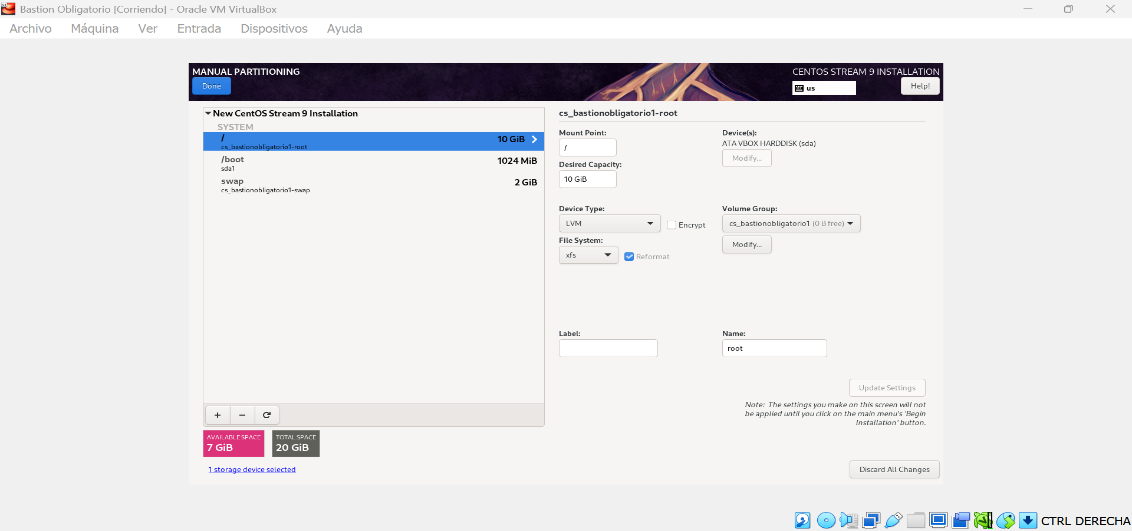
Para configurar la red del Bastion colocamos el adaptador 1 conectado a Nat



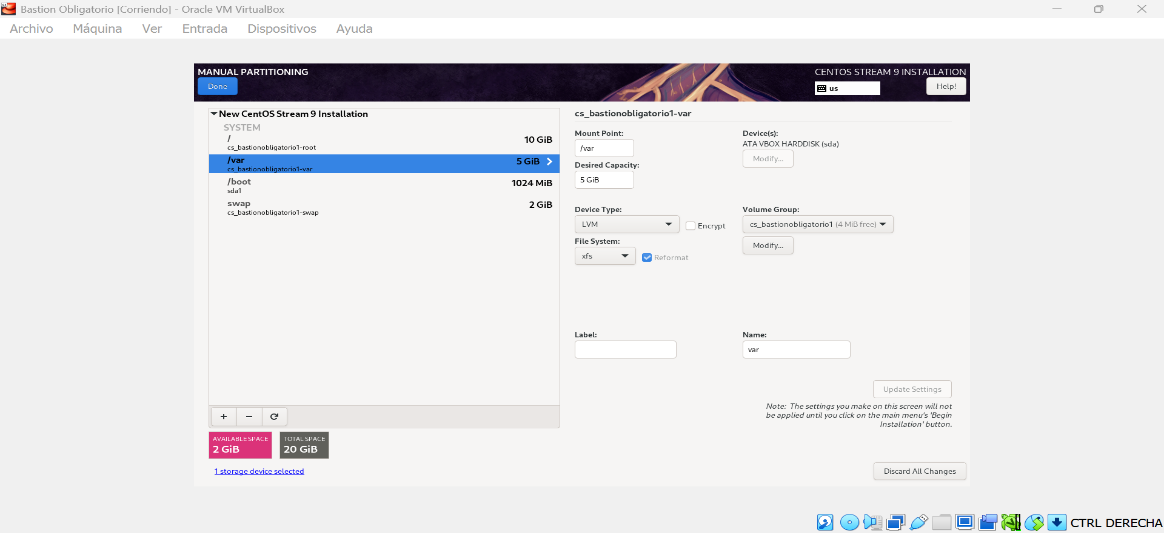
Se configura la red con Adaptador 2 con la red interna para que puedan ser accedidos por el bastion o controller



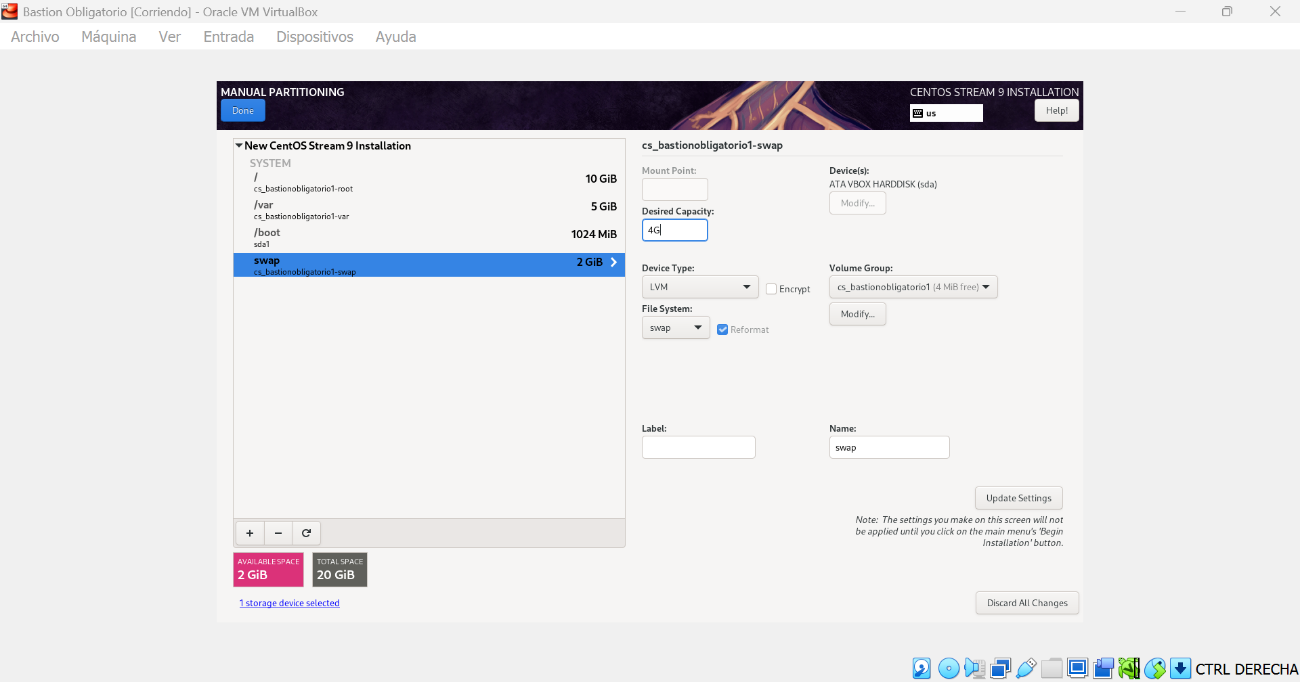
El fylesistem / boot queda configurado con 2 GB



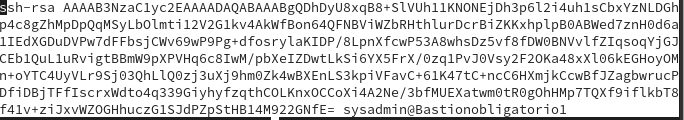
El fylesistem / quedo configurado con 10 GB



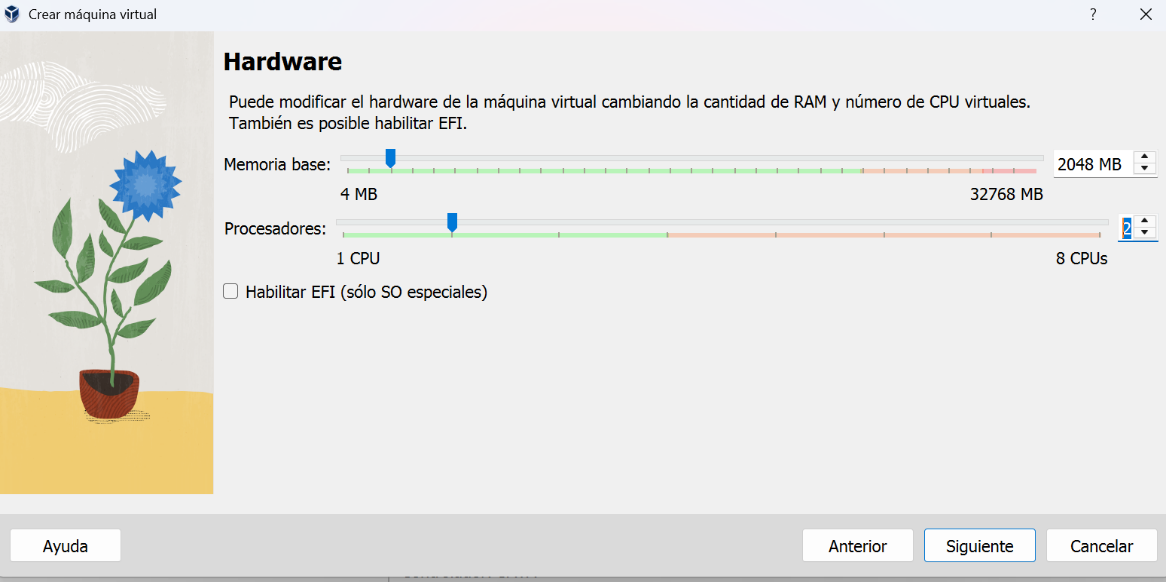
El fylesistem / var quedo configurado con 5G



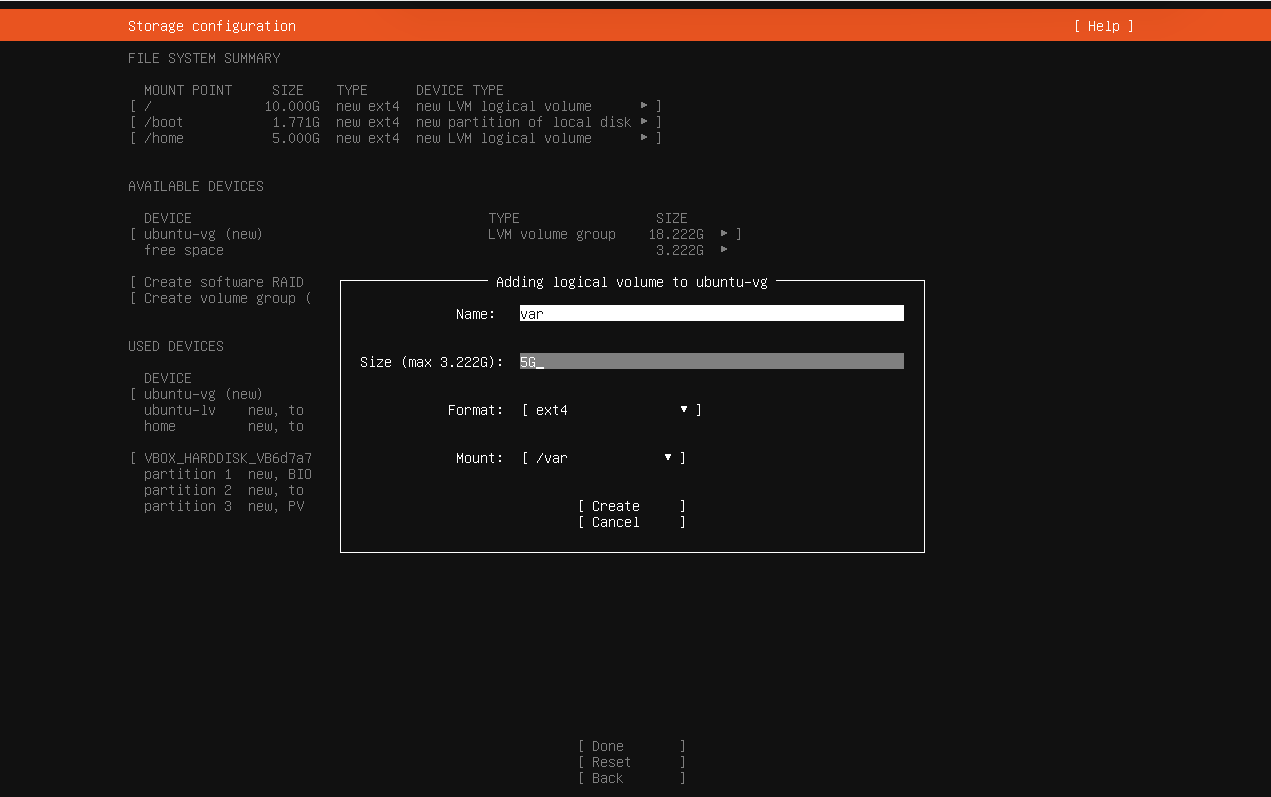
El fylesistem para swap quedo configurado con 4G



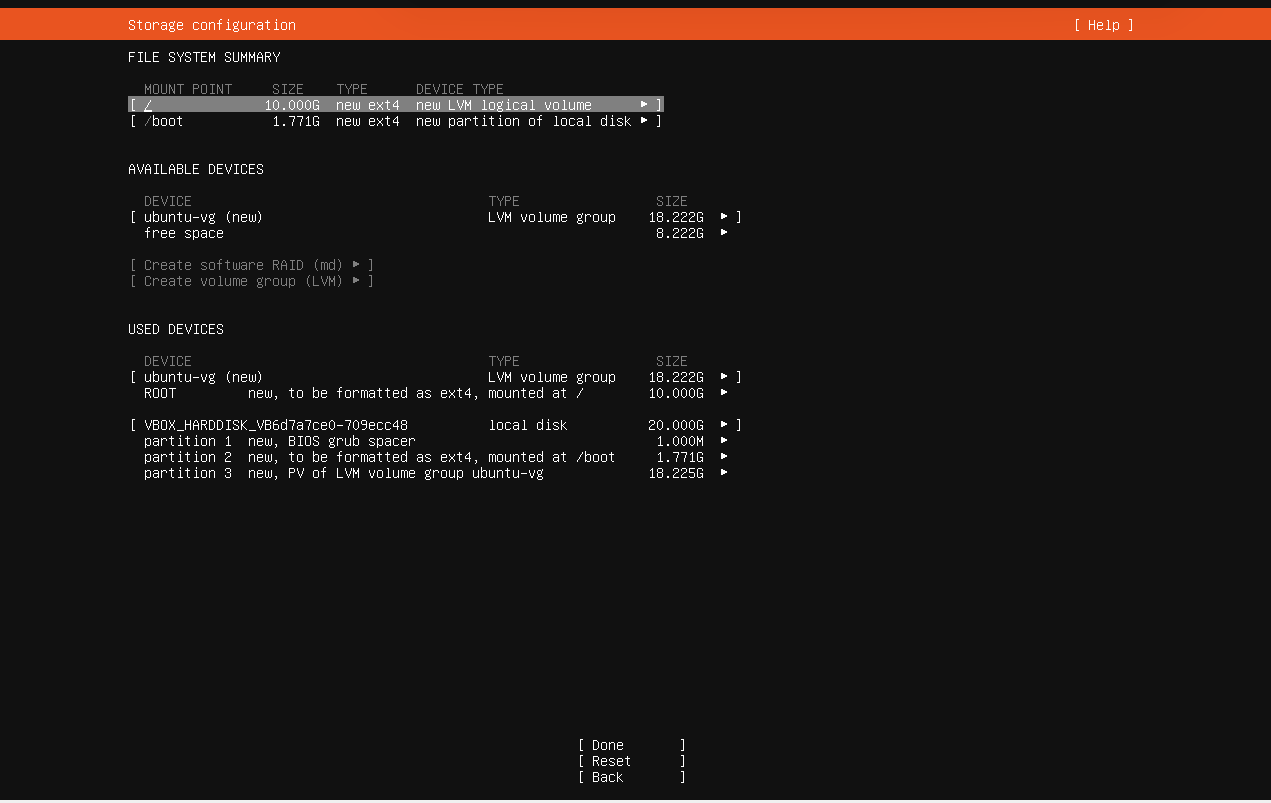
Se crea una clave publica desde el servidor Bastion Obligatorio



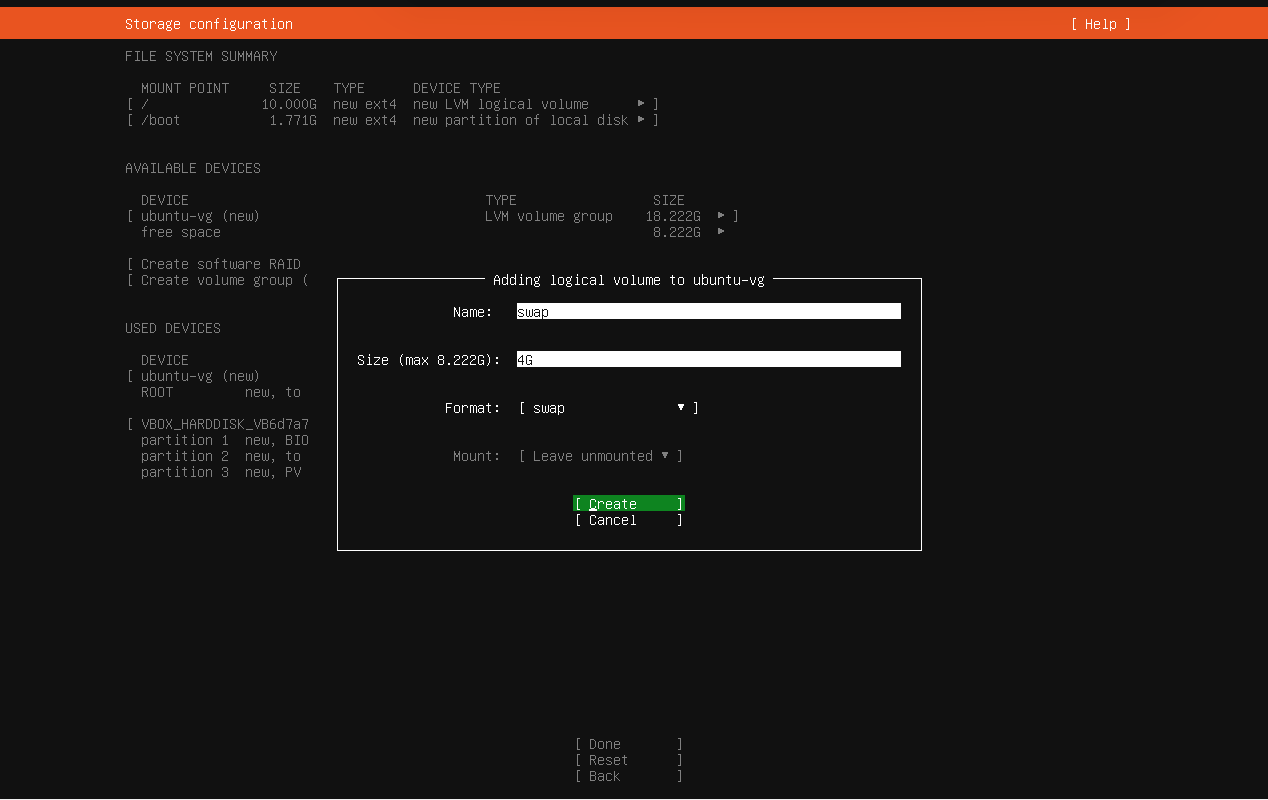
Se crea el servidor Ubuntu Obligatorio con un cpu



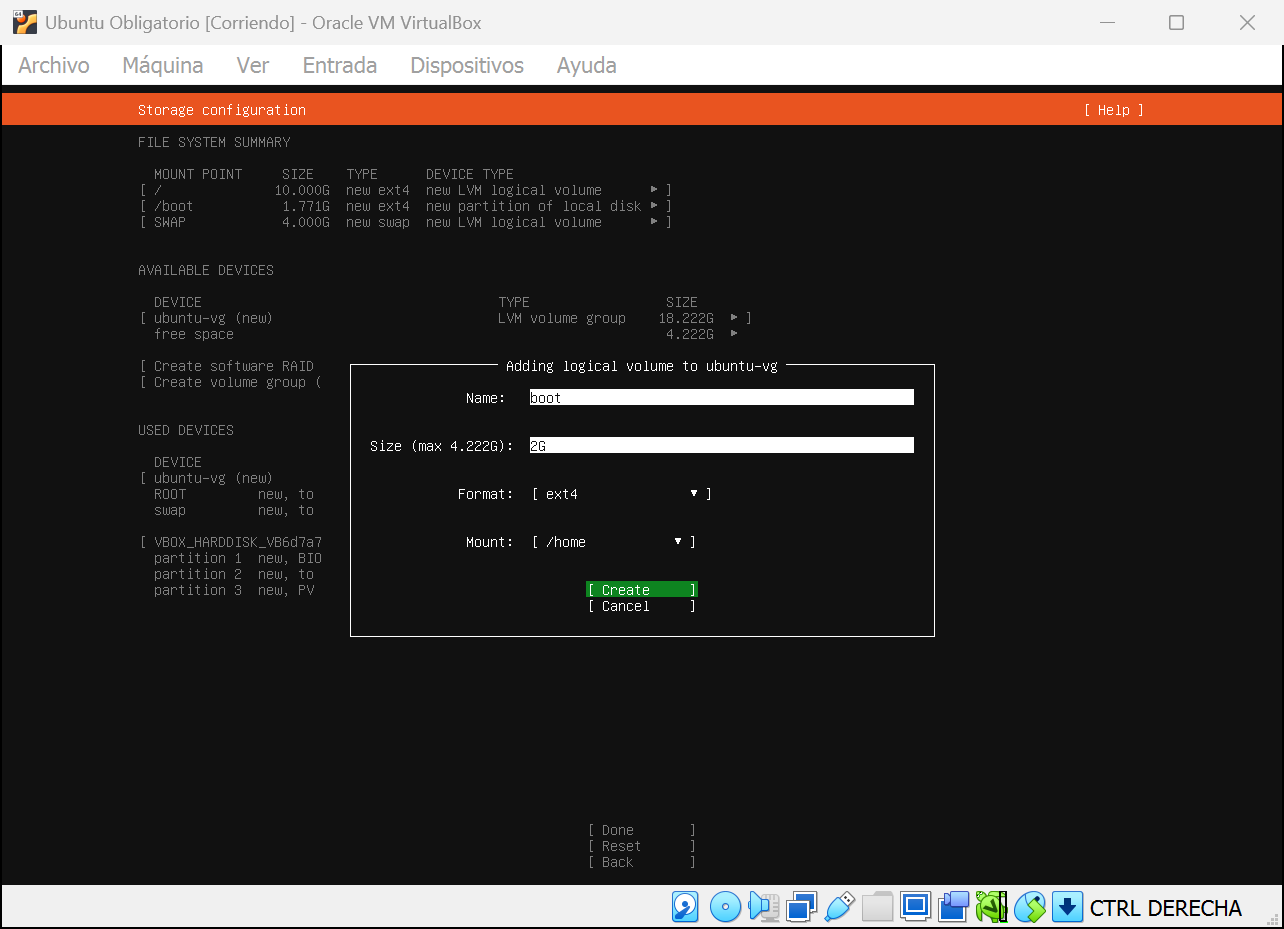
S e crea la partición / var en el fylesistem del servidor Ubuntu con 5G



Se crea la partición / en el fylesistem del servidor Ubuntu con 10G

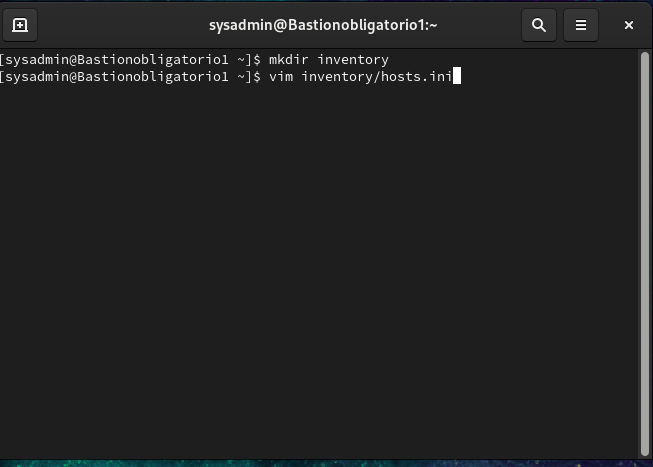


Se crea swap en el servidor Ubuntu con 4G

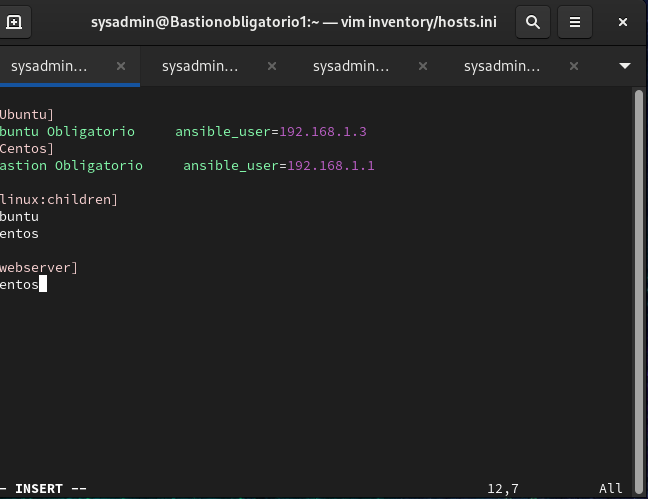


Se crea la partición / boot para el fylesistem en el servidor Ubuntu con 2G

Parte 2: Configurar un archivo de inventario de Ansible



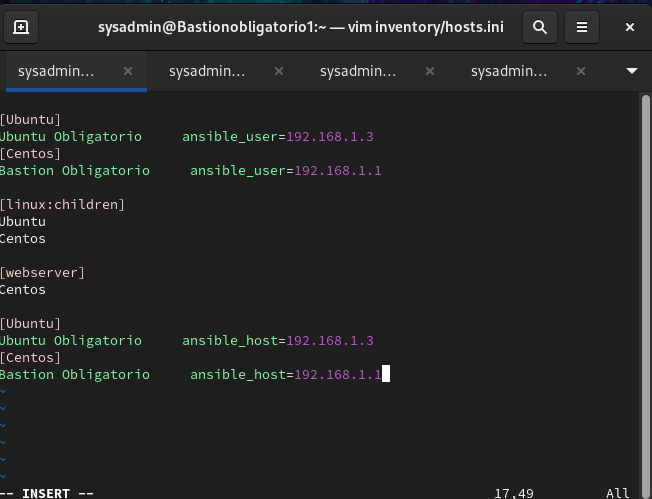
Se crea el archivo de inventario inventory.ini



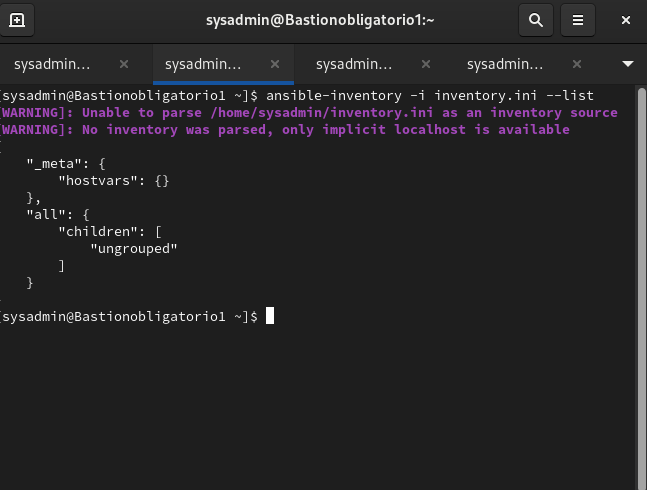
Se crea el grupo Ubuntu y Centos con un host por cada grupo.

También se crea un super grupo Linux con los grupos Centos y Ubuntu.

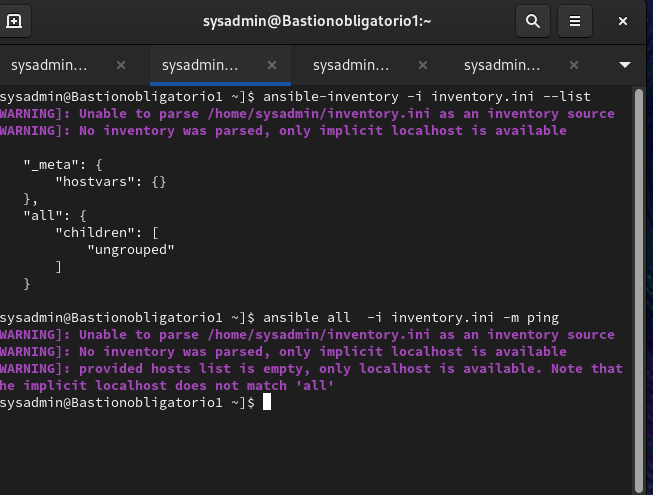
Se crea el grupo webserver que incluye al grupo Centos.



Se crea la variable ansible\_user y la variable ansible\_host para los hosts Ubuntu Obligatorio y Centos Obligatorio con sus respectivas IP.

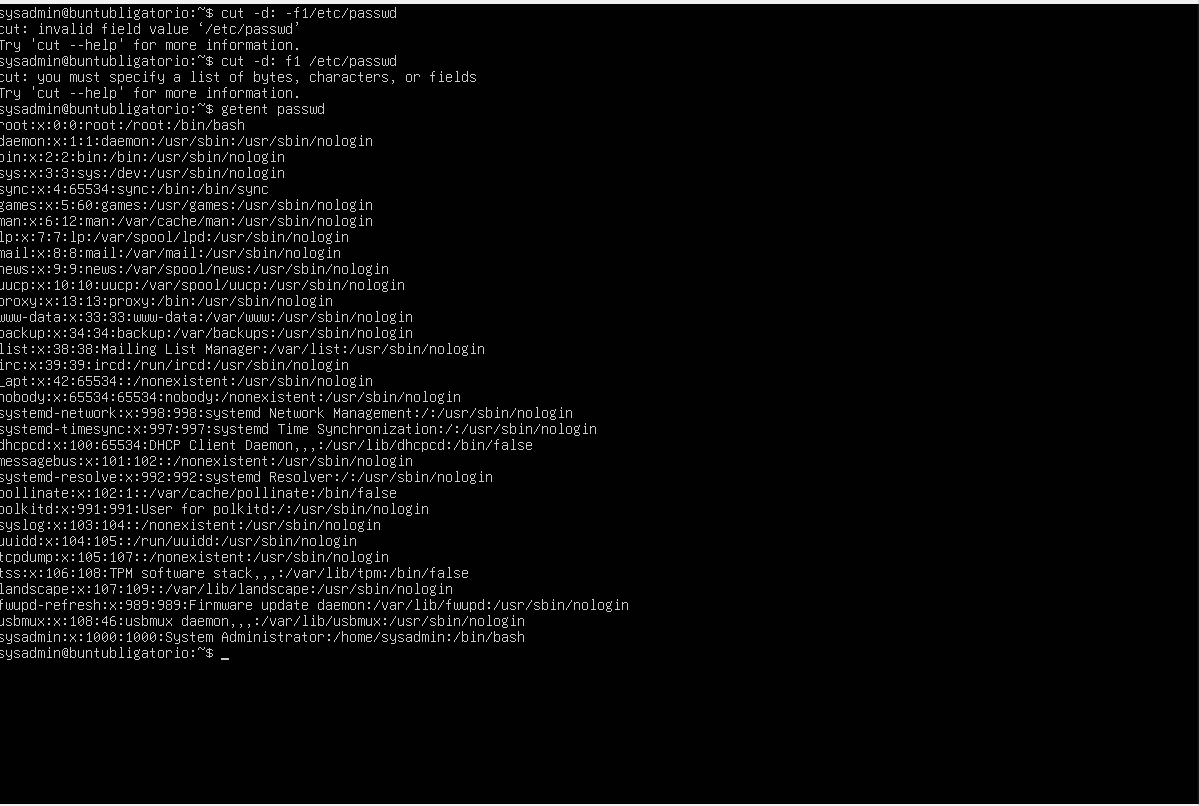


Ingresando el comando ansible-inventory -i inventory.ini –list se muestra la conexión exitosa



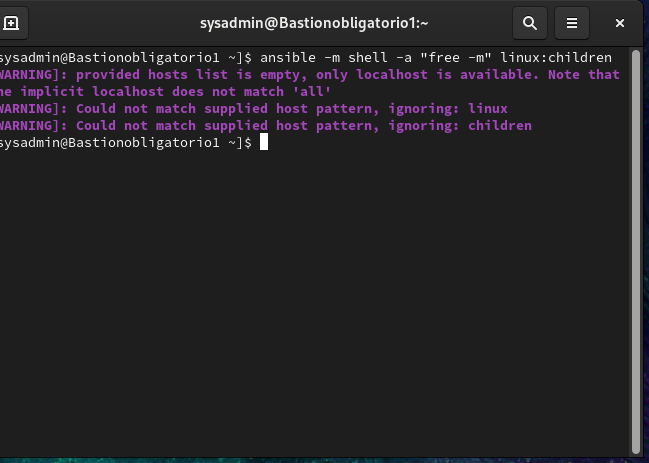
Con el segundo comando ingresado también se muestra la conexión exitosa

Parte 3: Ejecución y documentación de comandos ad-doc

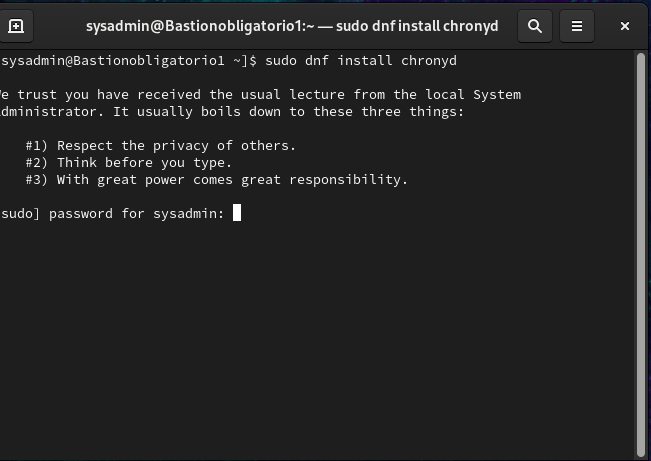


Con el comando getent passwd me lista todos los usuarios de la base de datos incluyendo usuarios del sistema y usuarios regulares

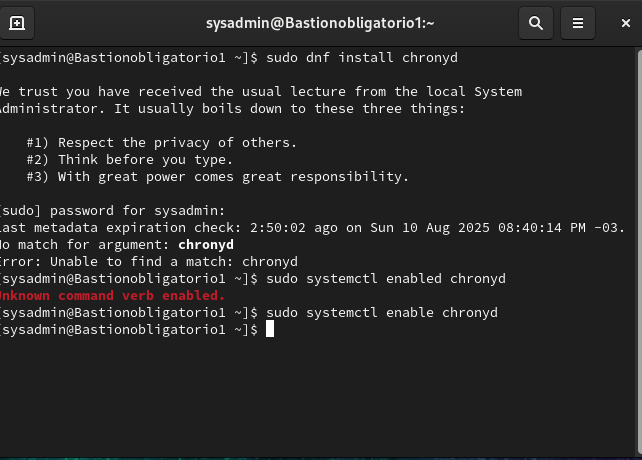
Tambien el comando cat /etc/ passwd me sirve para consultar lo mismo pero el comando getent passwd es más recomendable porque consulta todas las bases de datos configuradas



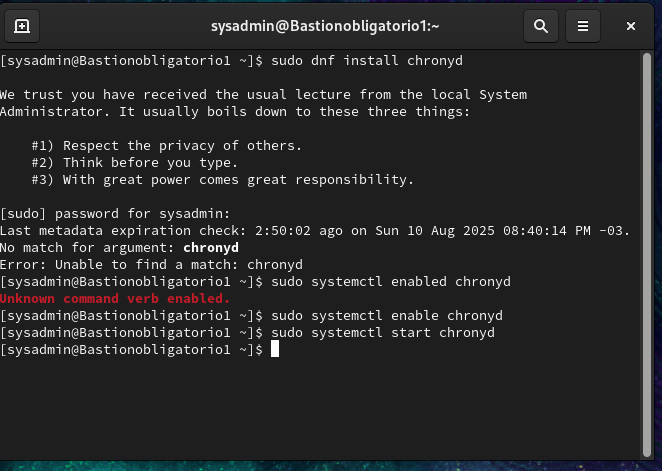
Este comando muestra la memoria en los servidores instalados conectados por ssh.



Queda instalado el servicio chronyd en el servidor Centos

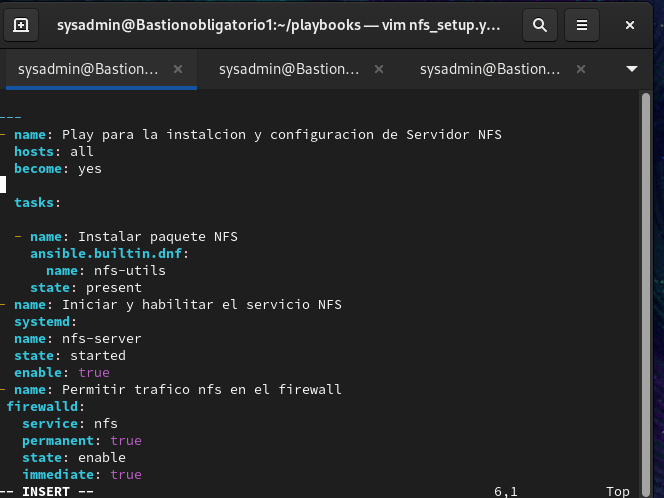


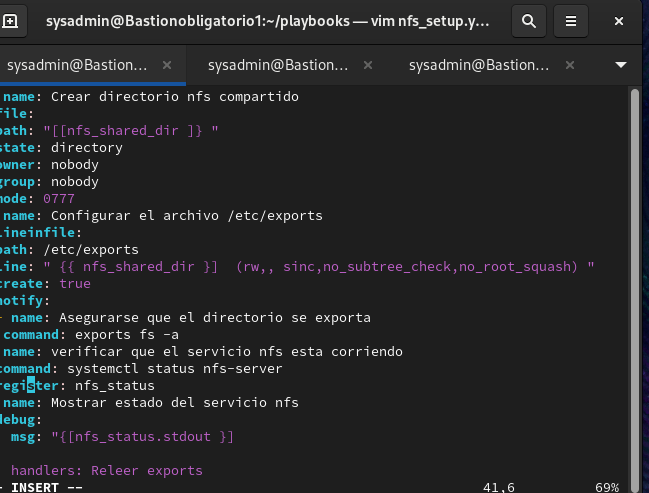
Con este comando habilito el servicio en el servidor CentOS



Con el comando systemctl start chronyd inicio el servicio en el servidor Centos

Parte 4: Crear y ejecutar playbook de Ansible









Parte 5: Cuestionario

1 ¿Que es Ansible? Mencione dos actividades que se puedan realizar con realizar con Ansible

Ansible es muy utilizada en sistemas operativos Linux. en Linux. Es una herramienta de automatización de código abierto que nos facilita la configuración, implementación y gestión de sistemas informáticos. Principalmente se utiliza para tareas como la gestión de la configuración, la implementación de aplicaciones y la orquestación de flujos de trabajo complejo.

Entre las actividades que se pueden realizar con Ansible se pueden destacar dos características importantes: en primer lugar, permite la automatización de la configuración del sistema.

Ansible nos permite definir el estado deseado de un sistema y aplicar de forma automática los cambios necesarios para alcanzarlo como instalar paquetes, configurar archivos, gestionar usuarios y servicios etc.

En segundo lugar, Ansible nos permite la implementación de aplicaciones. Ansible nos facilita la implementación de aplicaciones en múltiples servidores de manera consistente y repetible, automatizando tareas como la copia de archivos la configuración de servicios web y también nos permite ejecutar scripts de inicialización.

Otro elemento para destacar es que nos permite automatizar la creación y configuración de máquinas virtuales, contenedores y otros recursos de infraestructura.

2 ¿Que es un playbook de Ansible?

Un playbook de Ansible es un archivo YAML que define una serie de tareas automatizadas que Ansible va a ejecutar en hosts administrados. Es un conjunto de instrucciones que Ansible sigue para poner a los sistemas en un estado deseado.Los playbooks son la base de la automatización con Ansible y permiten a los usuarios definir configuraciones, coordinar pasos a seguir y ejecutar tareas de forma consistente y repetitiva.

Los playbooks se escriben en el lenguaje YAML que es fácil de leer y entender tanto para los humanos como para las maquinas. Un playbook define que acciones se deben realizar y en que orden utilizando módulos de Ansible para realizar distintas tareas como instalar paquetes, configurar archivos, gestionar usuarios etc.

Los playbooks nos permiten nos permiten administrar los hosts.

Apuntan a hosts o grupos de hosts específicos donde las tareas se ejecutaran.

Los playbooks también se pueden reutilizar en diferentes entornos y situaciones, lo cual garantiza la consistencia en la configuración de los sistemas.

Las tareas que se ejecutan dentro de un playbook se hacen en un orden especifico lo que permite controlar el orden de la automatización.

Básicamente se puede decir que un playbook de Ansible es un documento YAML que define una serie de tareas automatizadas apuntando a hosts específicos con el objetivo de llevarlos a un estado deseado de forma consistente y repetible.

3 ¿ Qué información contiene un inventario de Ansible?

Un inventario de Ansible contiene información sobre los hosts que Ansible gestiona, incluyendo sus nombres direcciones IP y grupos al os que pertenecen. Tambien puede contener variables específicas para cada host o grupo permitiendo a Ansible personalizar la ejecución de tareas de acuerdo a las características de cada máquina. Un inventario de Ansible puede contener información detallada como por ejemplo de los hosts.

Genera una lista de máquinas con las que Ansible interactúa. Pueden ser identificadas por nombres de host o direcciones IP.

Tambien los hosts pueden ser administrados en grupos para facilitar la gestión. Por ejemplo se pueden crear grupos como webservers o dbservers.

Los inventarios de Ansible manejan variables que es una información adicional que se puede asociar a hosts o grupos. Estas variables nos permiten personalizar la ejecución de tareas de Ansible como por ejemplo especificar la versión de software a instalar o configurar diferentes parámetros de red.

Tambien nos permite tener detalles de conexión a los hosts. Detalles por ejemplo como el usuario y la clave SSH necesarios para acceder a los hosts.

Además nos permite obtener inventarios dinámicos. En determinados casos el inventario puede ser dinámico obteniendo información de fuentes externas como proveedores de nube por ejemplo AWS o Azure.

Básicamente se puede decir que el inventario es un archivo que actúa como un directorio de los recursos que Ansible gestiona, permitiendo la automatización de tareas de manera eficiente y escalable.

4 ¿Explique que es un módulo de Ansible y de un ejemplo?

En Ansible un módulo es un pequeño script o programa que realiza una tarea especifica en un host gestionado. El host puede ser tanto local como remoto.

Un módulo son las herramientas que Ansible utiliza para ejecutar acciones concretas como por ejemplo instalar software, configurar servicios o manipular archivos.

Cada tarea de Ansible es ejecutada a través de un módulo.

Un ejemplo puede ser como se vería una tarea en un playbook de Ansible para instalar Apache:

Código

---  
- name: Instalar Apache  
 hosts: servidor\_remoto *# Reemplaza con la dirección de tu servidor*

tasks:  
 - name: Instalar el paquete Apache  
 apt:  
 name: apache2  
 state: present # *Asegura que el paquete esté instalado*

become: true # Necesario para ejecutar con privilegios de superusuario

Explicación:

* name: Instalar Apache: Es una descripción de la tarea, para facilitar la lectura.
* hosts: servidor\_remoto: Especifica en qué máquina o grupo de máquinas se ejecutará la tarea.
* tasks:: Define una lista de tareas a ejecutar.
* name: Instalar el paquete Apache: Una descripción de la tarea específica.
* apt: El módulo utilizado (en este caso, apt para sistemas basados en Debian/Ubuntu).
* name: apache2: El nombre del paquete a instalar.
* state: present: Indica que el paquete debe estar instalado.
* become: true: Indica que la tarea necesita ser ejecutada con privilegios de superusuario (sudo) para instalar el paquete.

En este ejemplo queda demostrado como un módulo en este caso apt se utiliza para realizar una acción determinada en este caso instalar Apache en un host gestionado dentro del contexto de un playbook de Ansible. Los módulos son la base de la automatización en Ansible

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

Ansible se destaca sobre otros métodos de automatización por su simplicidad, arquitectura sin agentes y un lenguaje que esta basado en YAML lo cual facilita la gestión y configuración de estructuras complejas. Una de las diferencias que tiene con otras herramientas es que Ansible no requiere la instalación de agentes en los nodos gestionados utilizando SSH o WinRM para la comunicación lo cual simplifica la implementación y reduce la sobrecarga. Por otra parte su lenguaje YAML permite una configuración declarativa lo cual facilita la comprensión y mantenimiento de los flujos de trabajo de automatización.

Una de las ventajas importantes de Ansible es que se caracteriza por tener una interfaz sencilla y su lenguaje YAML legible lo que facilita su aprendizaje y adopción. Es ideal para usuarios con poca experiencia en programación.

Al no requerir agentes en los nodos gestionados, Ansible reduce significativamente la complejidad y los riesgos de seguridad asociados con la instalación y gestión de software adicional.

Ansible facilita la orquestación de tareas complejas, permitiendo la automatización de flujos de trabajo que involucran múltiples sistemas y aplicaciones.

Otra de las ventajas que tiene utilizar Ansible es que se puede implementar rápidamente en distintas plataformas y entornos incluyendo nubes públicas y privadas lo cual lo convierte en una herramienta flexible y adaptable.

Otra de las ventajas que tiene es que tiene un desarrollo y despliegue ágil de las aplicaciones de esta forma asegura la estabilidad y coherencia de los entornos.

Ansible fomenta la colaboración entre equipos proporcionando una plataforma común para compartir conocimientos y practicas recomendadas.

Ansible posee información valiosa para identificar y resolver problemas operativos y de seguridad además facilita el análisis y la gestión de la automatización a gran escala.