**LABORATÓRIO 01** - **CONFIGURAÇÃO DE MÁQUINA ANSIBLE HOST, INSTALAÇÃO DO ANSIBLE, PRIMEIRAS CONFIGURAÇÕES DO ANSIBLE.**

**Requisitos: Configurando o Ansible HOST**

* Um virtualizador VMWare
* ISO Debian server 11.7.0 (Somente a CLI no caso) ou Debian-11-Base25GB OVF
* Máquina virtual com 2GB de RAM + 1 vCPU + 25GB de disco
* Nome da máquina (RECOMENDADO): ansiblehost
* OBS: É necessário que a placa de rede da máquina virtual esteja em modo Bridge (Para que o computador local consiga executar um ssh/ping na VM)
* A OVF já está com apt ajustado e vm-tools e vim instalado.

1. **INSTALAÇÃO DO ANSIBLE**
2. **CRIAR MÁQUINA CLIENTE**
3. **CONFIGURAÇÕES SSH PARA O ANSIBLE**
4. **PRIMEIRAS CONFIGURAÇÕES DO ANSIBLE**
5. **CRIAR ARQUIVO NA MÁQUINA CLIENT**
6. **TRABALHAR COM VÁRIOS HOSTS**
7. **INSTALAÇÃO DO ANSIBLE**

**1 - Acessar como root:**

Mudar para root: **su -**

**2 - Ajustar nome da máquina:**

execute: **hostnamectl set-hostname srvansible**

execute: **bash**

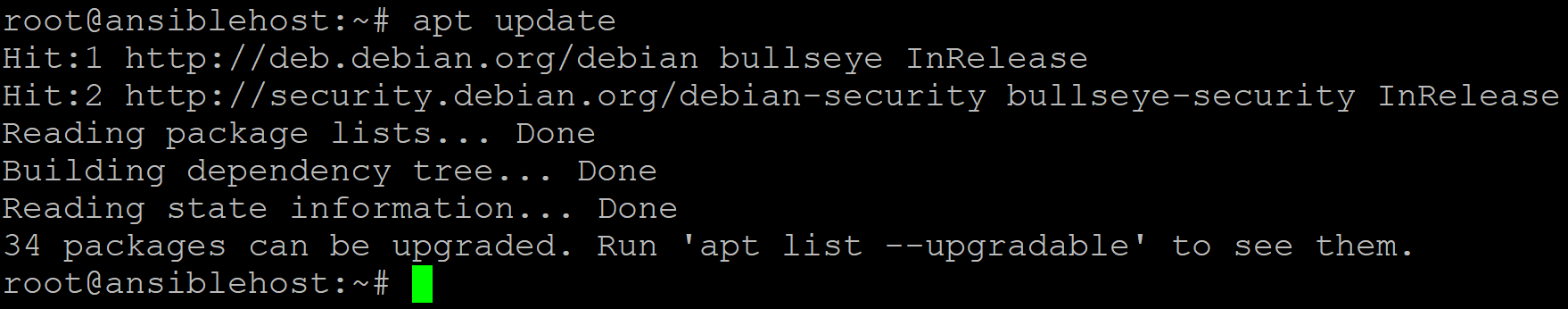
**3 - Ajustar servidor de pacotes do linux - APT:**

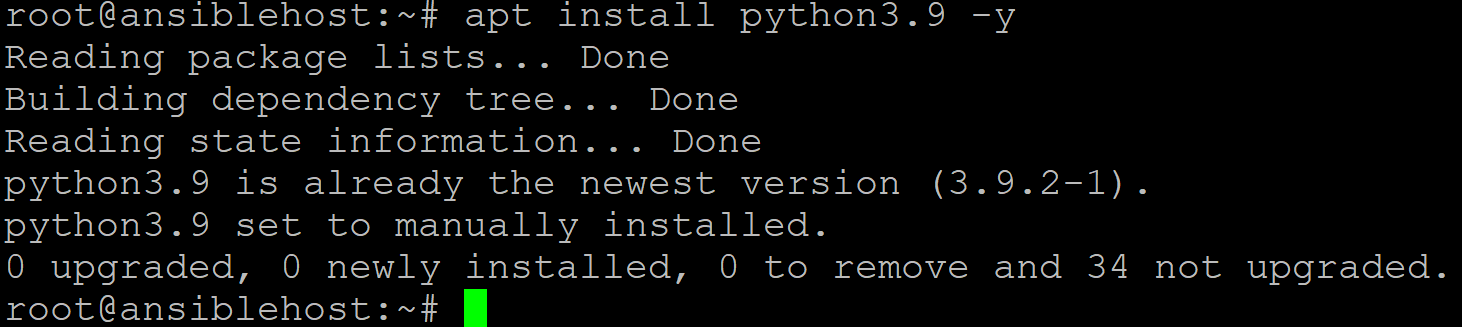
editar o sources.list: **nano /etc/apt/sources.list**

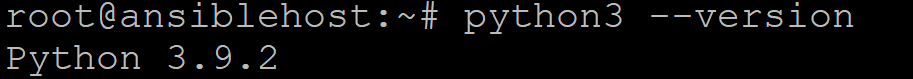
Deixar apenas essa linha:

**deb http://deb.debian.org/debian/ bullseye main**

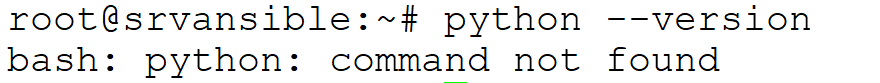
**4 - Após criar e acessar a máquina virtual, vamos atualizar os pacotes da nossa máquina:**

execute: **apt update -y**

**5 - Depois de atualizarmos nossa VM, precisamos instalar o python3 + pip (Caso dê algum erro de pacote no Ansible, essas ferramentas poderam nos ajudar):**  
Verifique se já instalado o python3, execute: **apt install python3.9 -y**

Verifique se o python3 foi instalado corretamente, execute: **python3 --version**

Verifique se o python é um comando reconhecido, execute: **python --version**

****

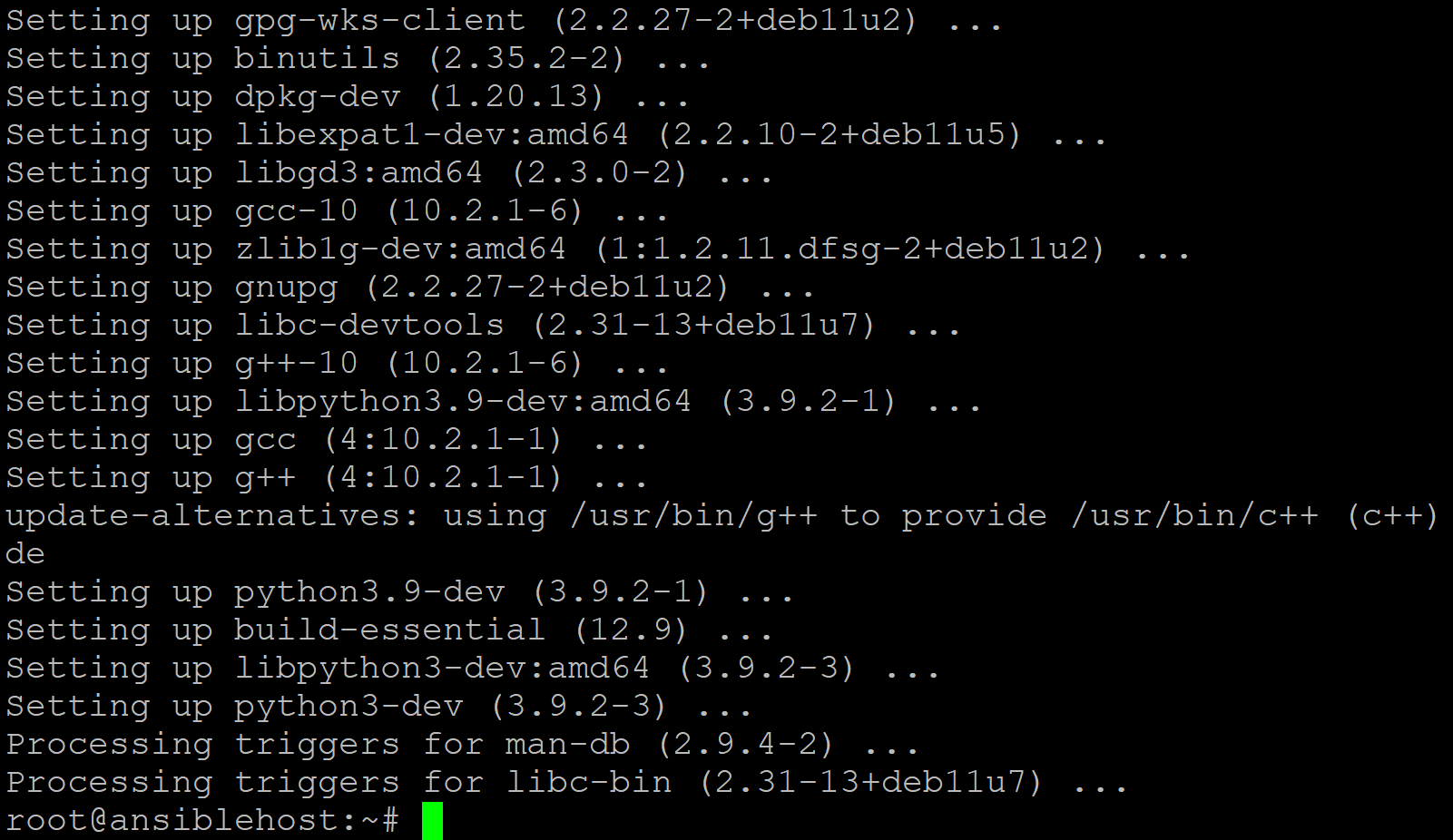
**6 - Vamos fazer com que o programa Python seja igual ao Python3**

execute:

**alias python=python3**

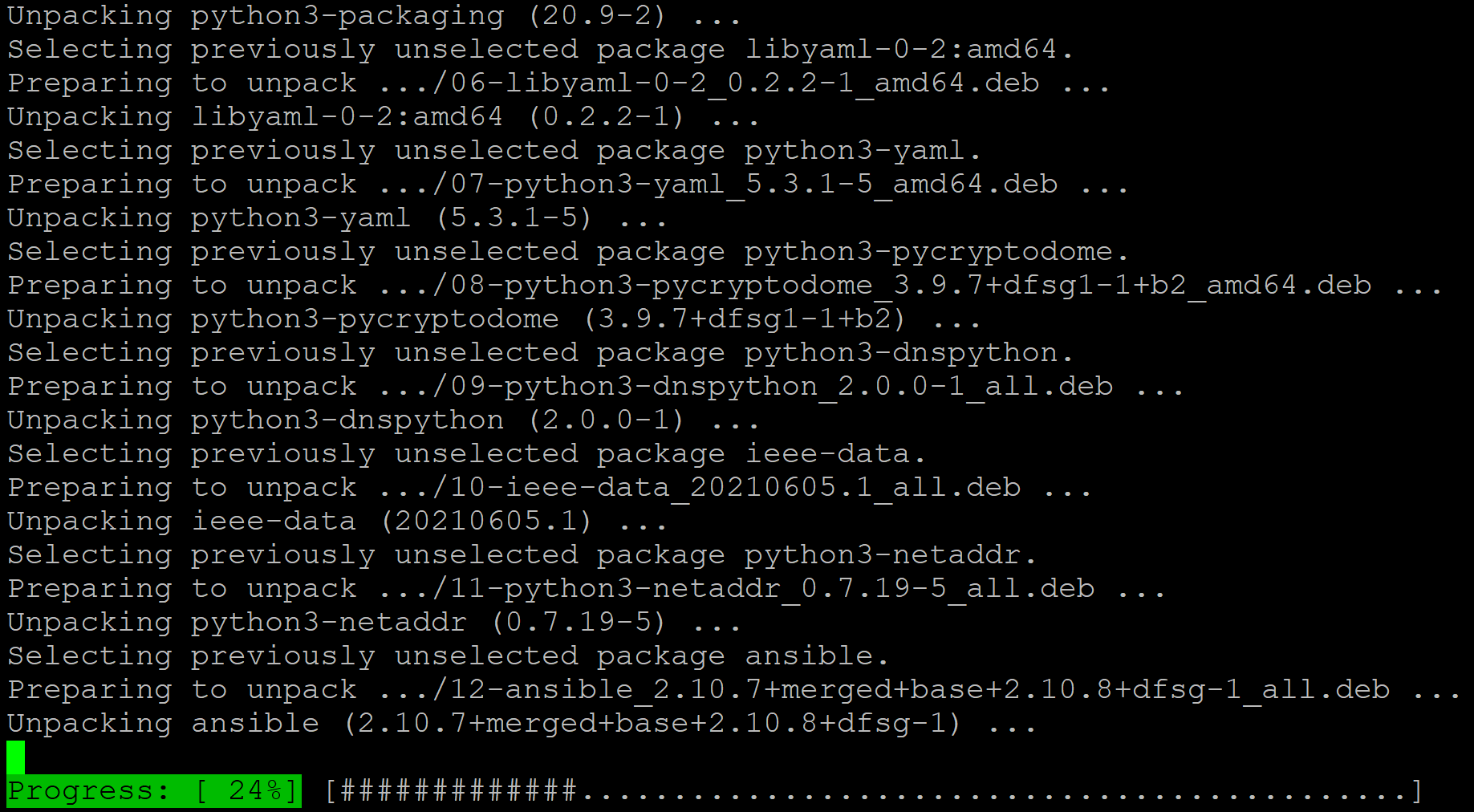
**apt install python-is-python3**

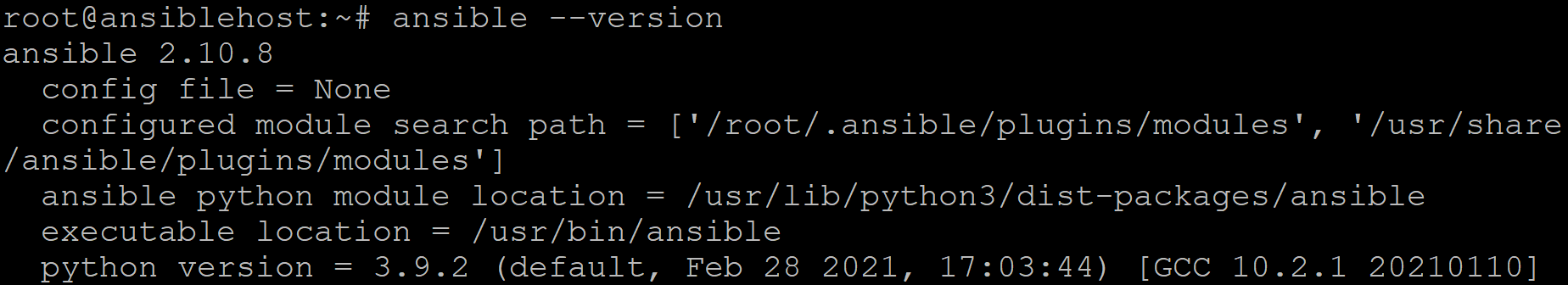
**7 - Vamos instalar a biblioteca PIP para Python3**

Após instalar o python3, instale o pip, execute: **apt install python3-pip -y**

**Já estamos prontos para instalar o Ansible**

**8 - Instale o Ansible:**

execute: **apt install ansible -y**

**9 - Valide se o ansible foi instalado corretamente:**execute: **ansible --version**

1. **CRIE UMA SEGUNDA MÁQUINA VIRTUAL**

**DEBIAN-CLIENT (NÃO EXCEDA LIMITES DE CPU, MEMÓRIA RAM E ARMAZENAMENTO)**

**OBS: Coloque a placa de ambas máquinas virtuais em Bridge**

**1 - Na segunda máquina virtual, vamos visualizar o IP dela:**

execute: **ip -c a**

**OBS: Copie o IP da segunda máquina DEBIAN-CLIENT**

**Ajuste o nome: hostnamectl set-hostname srvweb**

**Ajuste o APT - echo 'deb http://deb.debian.org/debian/ bullseye main' > /etc/apt/sources.list**

**2 - Liberar acesso pelo usuário root na máquina CLIENT:**

editar o arquivo: **nano /etc/ssh/sshd\_config**

editar linha 34, trocar por “**yes**”:

**PermitRootLogin yes**

Reiniciar serviço:

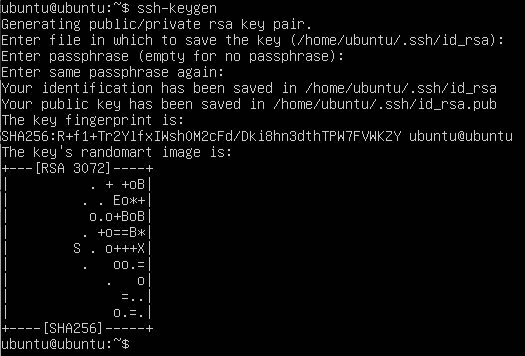
**service sshd restart**

1. **CONFIGURAÇÕES SSH PARA O ANSIBLE-HOST**

**1 - Para o Ansible funcionar corretamente, precisamos definir duas coisas: gerar novas chaves ssh e configurar um host**

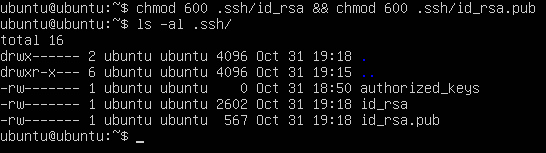
**Dentro da máquina ANSIBLE-HOST e no CLIENT.**

**2 - Gerando novas chaves SSH:**

execute: **ssh-keygen**

**OBS: Aperte a tecla ENTER, até que a chave seja criada (Não configure nenhuma senha passphrase, local que vai salvar e entre outros)**

**3 - Habilite a permissão 600 para as chaves SSH (Permissão para que o proprietário possa ler e escrever no arquivo)**

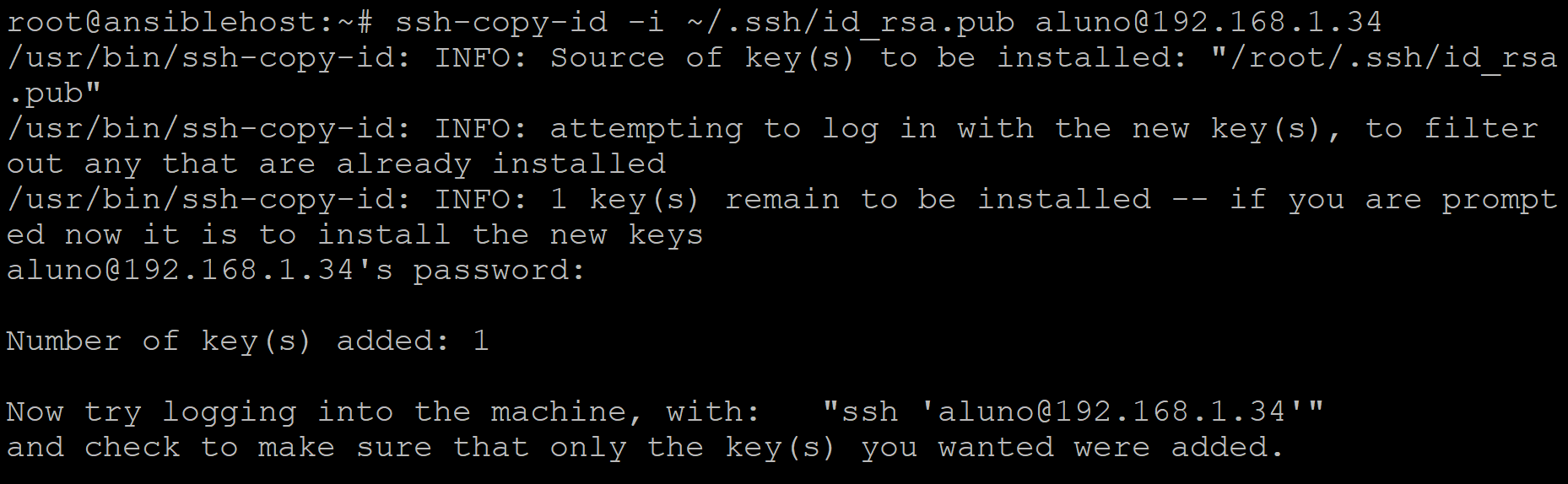
execute: **chmod 600 .ssh/id\_rsa\* **

**5 - Na máquina ANSIBLE execute o comando com o IP da segunda máquina CLIENT, execute o comando abaixo:**

**ssh-copy-id -i/.ssh/id\_rsa.pub root@ip\_da\_sua\_segunda\_maquina\_CLIENT**

**ex:**

**ssh-copy-id -i ~/.ssh/id\_rsa.pub aluno@192.168.1.34**

****

**6 - Faça um teste de conexão SSH**

Não será mais solicitado senha, execute:

**ssh root@172.16.62.232**

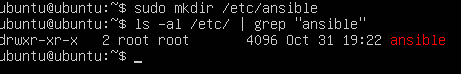
**7 - Sair da conexão SSH**

Pelo amor de Deus, execute:

**exit**

1. **PRIMEIRAS CONFIGURAÇÕES DO ANSIBLE**

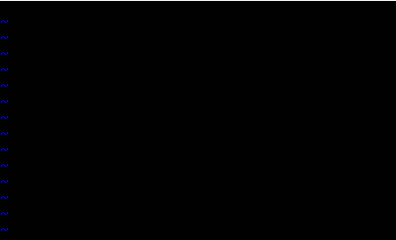
**1 - Agora vamos adicionar uma máquina CLIENT no ANSIBLE, para isso, precisamos criar um diretório chamado ansible, dentro da etc:**

execute: **mkdir /etc/ansible  
**

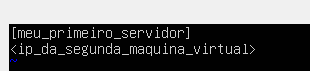
**OBS: Esse diretório é padrão do Ansible, é onde vamos configurar os nossos grupos de servidores, apontando IP, chave ssh.**

**Toda vez que o Ansible for fazer alguma instalação, ele vai acessar esse hosts ou o grupo de servidores para realizar a configuração.**

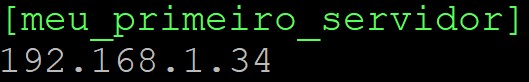
**2 - Precisamos de um arquivo chamado hosts para configurar os o IPs dos CLIENTs:**

execute: **nano /etc/ansible/hosts**

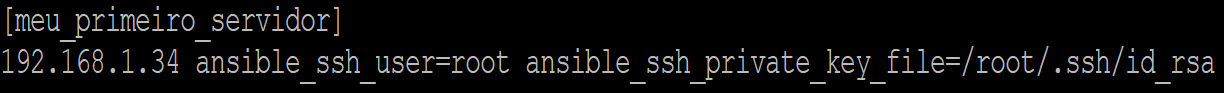
**4 - Dentro do arquivo hosts, configure da seguinte maneira:**

****

**OBS: Substitua *<ip\_da\_segunda\_maquina\_CLIENT>* pelo IP que você copiou anteriormente**

****

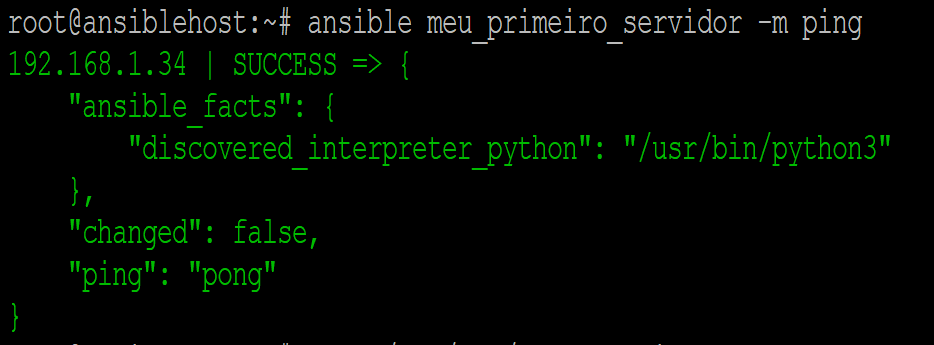
**Vamos configurar acesso SSH (RECOMENDADO):**

Na frente do IP que você copiou anteriormente, digite: **ansible\_ssh\_user=root ansible\_ssh\_private\_key\_file=/root/.ssh/id\_rsa**

**OBS: Após isso, salvar o arquivo**

**5 - Primeiro teste com comando AD-HOC, vamos validar se o Ansible consegue alcançar a nossa máquina:**

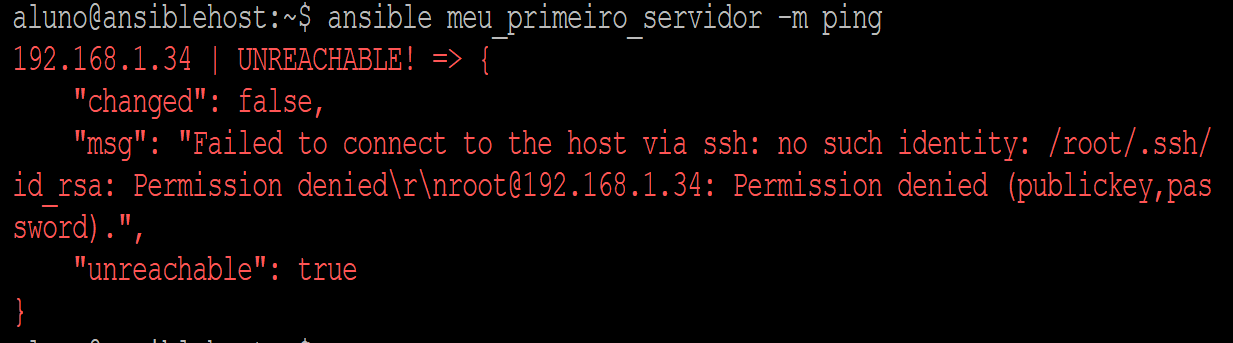
execute: **ansible meu\_primeiro\_servidor -m ping**



**OBS:** O parâmetro -m é o comando que será executado, neste caso ping

**8 - Possivel erro de SSH:**

execute: **ansible meu\_primeiro\_servidor -m ping**

****

**Em vermelho é o HOST que o ping foi executado.**

1. **CRIAR ARQUIVO NA MÁQUINA CLIENT**

**1 - Vamos criar um arquivo chamado sibov\_lindo.txt** **na nossa outra máquina CLIENT via Ansible:**

execute: **ansible meu\_primeiro\_servidor -m copy -a "content='Eu amo ser Nerd' dest=/root/sibov\_lindo.txt"**

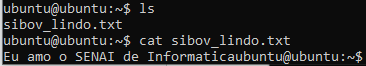
**CHANGED: Significa que o estado do sistema não foi alterado.**

**PING: É o output/saída, do comando executado (ping).**

**O comando ping não faz nada, pois testa somente a conectividade do sistema.**

**2 - Na máquina virtual CLIENT, no diretório do user root:**

execute: **cat sibov\_lindo.txt**

****

**OBS: Se o arquivo não existir, reconecte na VM (SEM DESLIGAR!)**

**3 - Tente ler um arquivo na máquina CLIENT pelo Servidor ANSIBLE**

**ansible all -m shell -a "cat /root/sibov\_lindo.txt"**

1. **TRABALHAR COM VÁRIOS HOSTS**
2. **CRIE UMA TERCEIRA MÁQUINA VIRTUAL**

**DEBIAN-CLIENT (NÃO EXCEDA LIMITES DE CPU, MEMÓRIA RAM E ARMAZENAMENTO)**

**OBS: Coloque a placa de ambas máquinas virtuais em Bridge**

**1 - Na segunda máquina virtual, vamos visualizar o IP dela:**

execute: **ip -c a**

**OBS: Copie o IP da segunda máquina DEBIAN-CLIENT**

**Ajuste o nome: hostnamectl set-hostname srvweb**

**Ajuste o APT - echo 'deb http://deb.debian.org/debian/bullseye main' > /etc/apt/sources.list**

**2 - Liberar acesso pelo usuário root na máquina CLIENT:**

editar o arquivo: **nano /etc/ssh/sshd\_config**

editar linha 34, trocar por “**yes**”:

**PermitRootLogin yes**

Reiniciar serviço:

**service sshd restart**

**3 - Podemos colocar mais de um IP, funcionará como se fosse uma lista.**

**Assim podemos configurar mais de um grupo de hosts, exemplo:**

**No /etc/ansible/hosts, adicionar grupos de servidores ou tipos de equipamentos:**

**[meu\_primeiro\_servidor]**

**PRIMEIRO\_IP\_JÁ\_CRIADO**

**[srvs\_linux]**

**<ip\_srv\_01>**

**<ip\_srv\_02>**

**[roteadores]**

**<ip\_router\_01>**

**<ip\_router\_02>**

**[switchs]**

**<ip\_switch\_03>**

**<ip\_switch\_04>**

**4 - Tente criar e ler arquivo nas máquinas CLIENTs pelo Servidor ANSIBLE ao mesmo tempo**

**ansible srvs\_linux -m copy -a "content='Eu amo o Prof Fofuxo' dest=/root/sibov\_lindo.txt"**

**ansible srvs\_linux -m shell -a "cat /root/sibov\_lindo.txt"**

**Consulta para máquinas que não estão em nenhum grupo: ungrouped**

**ansible ungrouped -m shell -a "cat /root/sibov\_lindo.txt"**