



TOCANTINS
GOVERNO DO ESTADO



03/2025

Aluno: Gabriel Mussatto Silva

Professor: Mailson Oliveira

UNIVERSIDADE ESTADUAL DO TOCANTINS
SISTEMAS DE INFORMAÇÃO
COMPUTAÇÃO ORIENTADA A SERVIÇOS

Relatório Laboratório ANSIBLE

Palmas/TO

2025

03/2025

Sumário

1	Introdução.....	3
2	Ambiente e Preparação	3
2.1	Criação da Conta e Instâncias na AWS	3
2.2	Configuração Local	6
3	Instalação e Configuração do Ansible	7
3.1	Instalação	7
3.2	Configuração do Inventário.....	8
4	Criação e Execução dos Playbooks	9
4.1	Playbook para instalar o Nginx.....	9
4.2	Playbook para instalar o Apache.....	10
5	Resultados	11
6	Conclusão	11

03/2025

1 Introdução

O presente relatório tem como objetivo documentar o procedimento de configuração de servidores na AWS utilizando o Ansible para automação de tarefas. Através da criação de instâncias, definição de chaves de acesso e execução de comandos remotos, foi possível automatizar a instalação de serviços essenciais, como Nginx e Apache. Este documento apresenta, de forma detalhada, cada etapa do processo, desde a preparação do ambiente até a execução dos playbooks, oferecendo uma visão completa dos procedimentos adotados e dos resultados alcançados.

2 Ambiente e Preparação

2.1 Criação da Conta e Instâncias na AWS

Para a criação da conta na AWS acesse o site: <https://aws.amazon.com/pt/>, e configure a conta conforme desejado. Em seguida na tela inicial entre na sessão “EC2” para poder criar as instâncias (servidores virtuais) necessárias.

Abrindo

a

seguinte

tela:

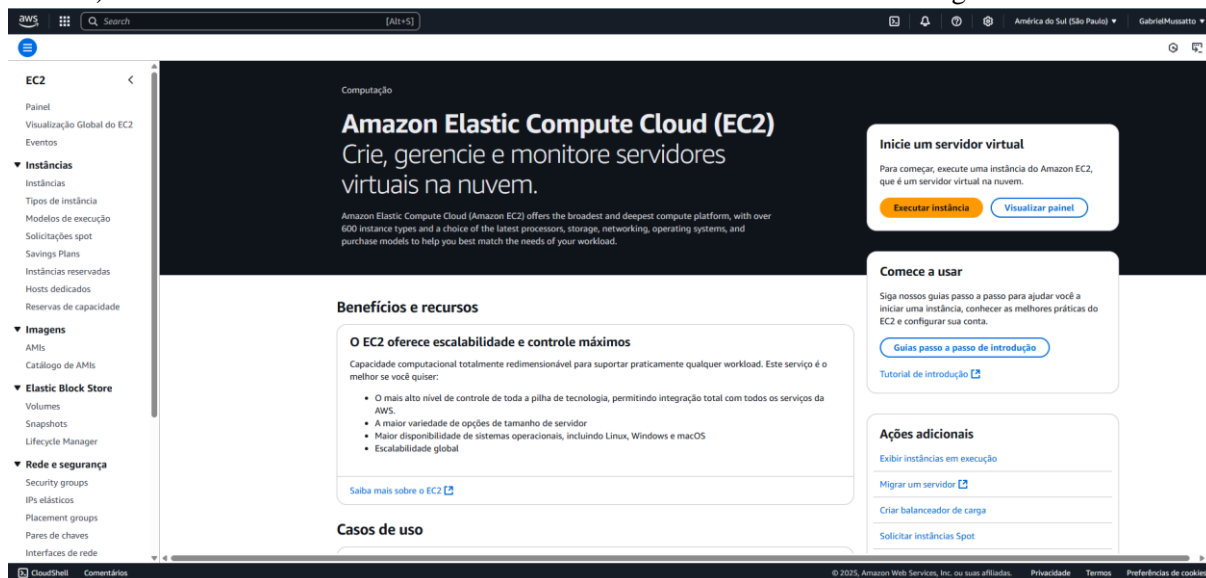
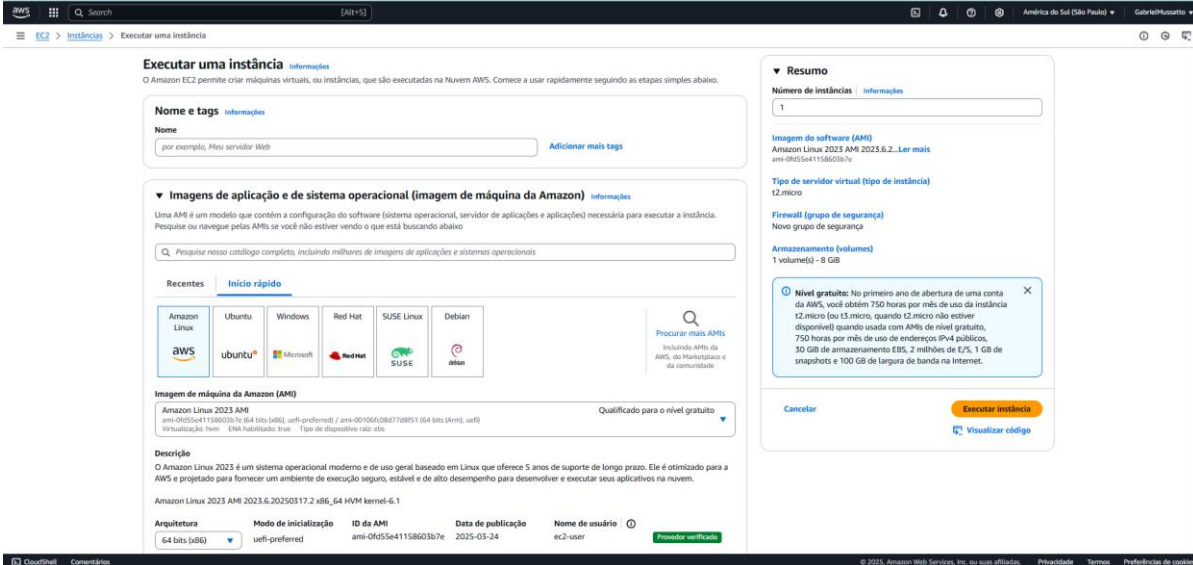


Figura 1 - Tela AWS EC2

Clicando no botão laranja “Executar Instância” o site direciona para a tela de criação das instâncias.

03/2025



Executar uma instância [Informações](#)

O Amazon EC2 permite criar máquinas virtuais, ou instâncias, que são executadas na Nuvem AWS. Comece a usar rapidamente seguindo as etapas simples abaixo.

Nome e tags [Informações](#)

Nome [Adicionar mais tags](#)

Imagens de aplicação e de sistema operacional (imagem de máquina da Amazon) [Informações](#)

Uma AMI é um modelo que contém a configuração do software (sistema operacional, servidor de aplicações e aplicações) necessária para executar a instância. Pesquise ou navegue pelas AMIs se você não estiver vendo o que está buscando abaixo.

Recentes **Início rápido**

Amazon Linux Ubuntu Windows Red Hat SUSE Linux Debian

Imagem de máquina da Amazon (AMI)

Amazon Linux 2023 AMI Qualificado para o nível gratuito

ami-0f55e41158603b7e (64 bits (x86), uefi-preferred) / ami-001064c0867788951 (64 bits (ARM), uefi)
Virtualização: hvm - ENA habilitado: true - Tipo de dispositivo: iatd: ebs

Descrição

O Amazon Linux 2023 é um sistema operacional moderno e de uso geral baseado em Linux que oferece 5 anos de suporte de longo prazo. Ele é otimizado para a AWS e projetado para fornecer um ambiente de execução seguro, estável e de alto desempenho para desenvolver e executar seus aplicativos na nuvem.

Amazon Linux 2023 AMI 2023.6.20250317.2 x86_64 HVM kernel-6.1

Arquitetura **Modo de inicialização** **ID da AMI** **Data de publicação** **Nome de usuário** **Provedor verificado**

64 bits (x86) uefi-preferred ami-0f55e41158603b7e 2025-03-24 ec2-user Provedor verificado

Resumo

Número de instâncias [Informações](#)

1

Imagem do software (AMI)

Amazon Linux 2023 AMI 2023.6.2...Ler mais
ami-0f55e41158603b7e

Tipo de servidor virtual (tipo de instância)

t2.micro

Firewall (grupo de segurança)

Novo grupo de segurança

Armazenamento (volumes)

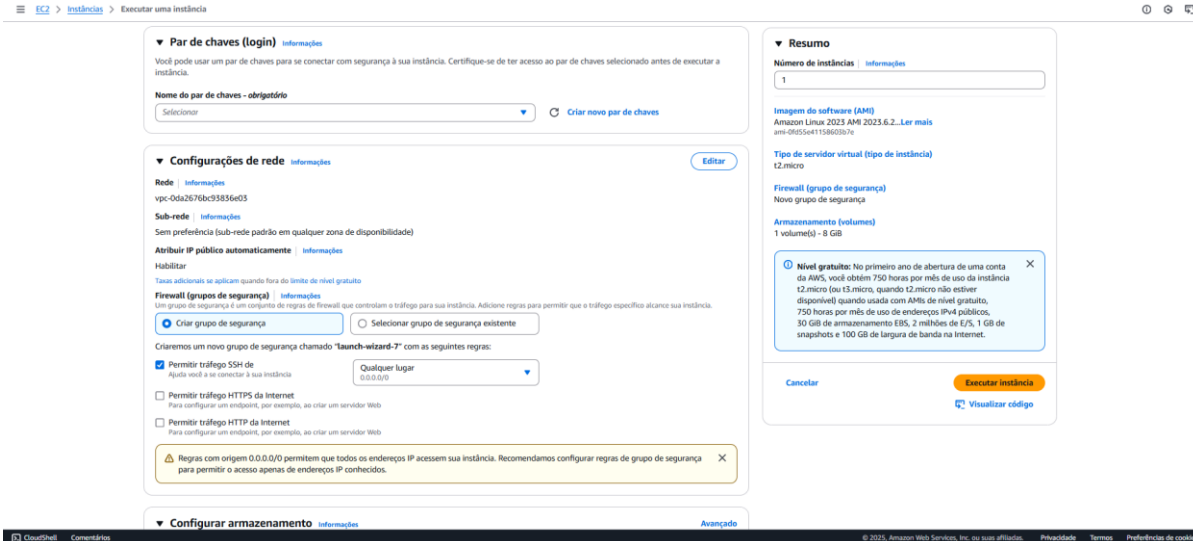
1 volume(s) - 8 GB

Nível gratuito: No primeiro ano de abertura de uma conta da AWS, você obtém 750 horas por mês de uso da instância t2.micro (ou t3.micro, quando t2.micro não estiver disponível) quando usada com AMIs de nível gratuito, 750 horas por mês de uso de endereços IPv4 públicos, 30 GB de armazenamento EBS, 2 milhões de IOPS, 1 GB de snapshots e 100 GB de largura de banda na Internet.

[Cancelar](#) [Executar instância](#) [Visualizar código](#)

Figura 2 - Executar instância

No campo “Nome e tags” foi declare o nome da máquina, em seguida selecione a imagem da máquina da sua preferência. Descendo a página teremos os outros campos:



Par de chaves (login) [Informações](#)

Você pode usar um par de chaves para se conectar com segurança à sua instância. Certifique-se de ter acesso ao par de chaves selecionado antes de executar a instância.

Nome do par de chaves - obrigatório

[Selecionar](#) ☐ [Criar novo par de chaves](#)

Configurações de rede [Informações](#) [Editar](#)

Rede [Informações](#)

vpc-0da267db-c38356e03

Sub-rede [Informações](#)

Sem preferência (sub-rede padrão em qualquer zona de disponibilidade)

Atribuir IP público automaticamente [Informações](#)

Habilitar

Firewall (grupos de segurança) [Informações](#)

Um grupo de segurança é um conjunto de regras de firewall que controlam o tráfego para sua instância. Adicione regras para permitir que o tráfego específico alcance sua instância.

☒ [Criar grupo de segurança](#) ☐ [Selecionar grupo de segurança existente](#)

Criaremos um novo grupo de segurança chamado "launch-wizard-2" com as seguintes regras:

☒ Permitir tráfego SSH de [Ajude você a se conectar à sua instância](#)

☐ Permitir tráfego HTTPS da Internet

☐ Permitir tráfego HTTP da Internet

Regras com origem 0.0.0.0/0 permitem que todos os endereços IP acessem sua instância. Recomendamos configurar regras de grupo de segurança para permitir o acesso apenas de endereços IP conhecidos.

Configurar armazenamento [Informações](#) [Avançado](#)

Resumo

Número de instâncias [Informações](#)

1

Imagem do software (AMI)

Amazon Linux 2023 AMI 2023.6.2...Ler mais
ami-0f55e41158603b7e

Tipo de servidor virtual (tipo de instância)

t2.micro

Firewall (grupo de segurança)

Novo grupo de segurança

Armazenamento (volumes)

1 volume(s) - 8 GB

Nível gratuito: No primeiro ano de abertura de uma conta da AWS, você obtém 750 horas por mês de uso da instância t2.micro (ou t3.micro, quando t2.micro não estiver disponível) quando usada com AMIs de nível gratuito, 750 horas por mês de uso de endereços IPv4 públicos, 30 GB de armazenamento EBS, 2 milhões de IOPS, 1 GB de snapshots e 100 GB de largura de banda na Internet.

[Cancelar](#) [Executar instância](#) [Visualizar código](#)

Figura 3 - Executar Instância (2)

Para gerar o par de chaves para a máquina, clique em “criar novo par de chaves”, onde deverá abrir o campo:

03/2025

Criar par de chaves

Nome do par de chaves

Os pares de chaves permitem que você se conecte à sua instância com segurança.

Insira o nome do par de chaves

O nome pode incluir até 255 caracteres ASCII. Ele não pode incluir espaços iniciais ou finais.

Tipo de par de chaves

☒ RSA
Par de chaves públicas e privadas criptografadas por RSA

☐ ED25519
Par de chaves ED25519 públicas e privadas criptografadas

Formato de arquivo de chave privada

☒ .pem
Para uso com OpenSSH

☐ .ppk
Para uso com PuTTY

Quando solicitado, armazene a chave privada em um local seguro e acessível no seu computador. Você precisará dele mais tarde para se conectar à sua instância. Saiba mais

Cancelar

Criar par de chaves

Figura 4 - Criar par de chaves

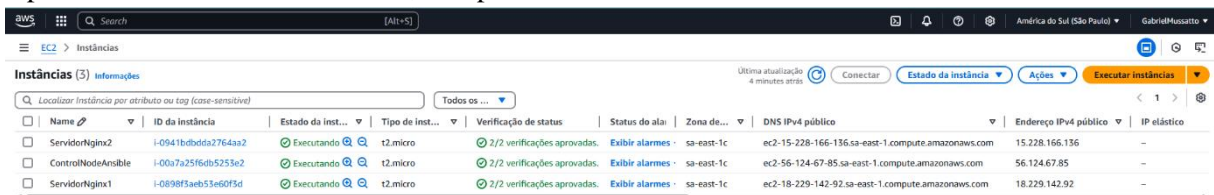
Coloque o nome da sua preferência, e crie o par de chaves.

Em seguida você poderá clicar em “Executar Instância” como mostra na figura 2.

A configuração das instâncias seguiu o seguinte esquema:

- **VM01**
 - Nome: ControlNodeAnsible
 - Imagem: Ubuntu
 - Nome par de chaves: chaveControlNode
- **VM02**
 - Nome: ServidorNginx1
 - Imagem: Ubuntu
 - Nome par de chaves: ServidorNginx1
- **VM03**
 - Nome: ServidorNginx2
 - Imagem: Ubuntu
 - Nome par de chaves: ServidorNginx2

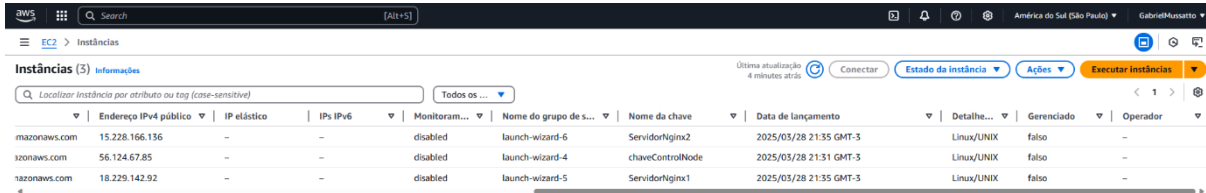
Após ter criado todas as instâncias, elas podem ser visualizadas no site.



	Name	ID da instância	Estado da inst...	Tipo de inst...	Verificação de status	Status do alai	Zona de...	DNS IPv4 público	Endereço IPv4 público	IP elástico
<input type="checkbox"/>	ServidorNginx2	i-0941b0bda2764aa2	Executando	t2.micro	2/2 verificações aprovadas.	Exibir alarmes	sa-east-1c	ec2-15-228-166-136.sa-east-1.compute.amazonaws.com	15.228.166.136	-
<input type="checkbox"/>	ControlNodeAnsible	i-00a7a25f6db5253e2	Executando	t2.micro	2/2 verificações aprovadas.	Exibir alarmes	sa-east-1c	ec2-56-124-67-85.sa-east-1.compute.amazonaws.com	56.124.67.85	-
<input type="checkbox"/>	ServidorNginx1	i-0898f3aeb53e60f5d	Executando	t2.micro	2/2 verificações aprovadas.	Exibir alarmes	sa-east-1c	ec2-18-229-142-92.sa-east-1.compute.amazonaws.com	18.229.142.92	-

Figura 5 - Instâncias criadas

03/2025



	Endereço IPv4 público	IP elástico	IPs IPv6	Monitoram...	Nome do grupo de s...	Nome da chave	Data de lançamento	Detalhe...	Gerenciado	Operador
mazonaws.com	15.228.166.136	-	-	disabled	launch-wizard-6	ServidorNginx2	2025/03/28 21:35 GMT-3	Linux/UNIX	falso	-
azonaws.com	56.124.67.85	-	-	disabled	launch-wizard-4	chaveControlNode	2025/03/28 21:31 GMT-3	Linux/UNIX	falso	-
azonaws.com	18.229.142.92	-	-	disabled	launch-wizard-5	ServidorNginx1	2025/03/28 21:35 GMT-3	Linux/UNIX	falso	-

Figura 6 - Instâncias criadas (2)

Podendo ver as informações relacionadas a cada instância.

2.2 Configuração Local

No ambiente local, as chaves de acesso foram transferidas para a instância ControlNodeAnsibel por meio do comando scp.

```
C:\Windows\System32> scp -i "C:\Users\gabriel\cos\chaveControlNode.pem" "C:\Users\gabriel\cos\ServidorNginx1.pem" ubuntu@ec2-56-124-67-85.sa-east-1.compute.amazonaws.com:/home/ubuntu
ServidorNginx1.pem
100% 1674 48.1KB/s 00:00

C:\Windows\System32> scp -i "C:\Users\gabriel\cos\chaveControlNode.pem" "C:\Users\gabriel\cos\ServidorNginx2.pem" ubuntu@ec2-56-124-67-85.sa-east-1.compute.amazonaws.com:/home/ubuntu
ServidorNginx2.pem
100% 1678 48.2KB/s 00:00
```

Figura 7 - Transferindo chaves para ControlNodeAnsible

A transferência foi realizada utilizando o CMD como administrador. Posteriormente, foi estabelecida a conexão com o ControlNodeAnsible via SSH utilizando o comando:

```
PS C:\Users\gabriel\cos> ssh -i "chaveControlNode.pem" ubuntu@ec2-56-124-67-85.sa-east-1.compute.amazonaws.com
Welcome to Ubuntu 24.04.2 LTS (GNU/Linux 6.8.0-1024-aws x86_64)

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

System information as of Sat Mar 29 02:36:05 UTC 2025

System load:  0.0          Processes:      108
Usage of /:   25.4% of 6.71GB Users logged in: 0
Memory usage: 21%         IPv4 address for enX0: 172.31.32.128
Swap usage:   0%

 * Ubuntu Pro delivers the most comprehensive open source security and
   compliance features.

   https://ubuntu.com/aws/pro

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

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

Last login: Sat Mar 29 02:21:41 2025 from 18.228.70.37
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-32-128:~$ |
```

Figura 8 - Conexão com ControlNodeAnsible

03/2025

3 Instalação e Configuração do Ansible

3.1 Instalação

Dentro da instância ControlNodeAnsible, os seguintes comandos foram executados para atualizar o sistema e instalar as dependências necessárias. Seguindo a ordem:

```
ubuntu@ip-172-31-32-128:~$ sudo apt update && sudo apt upgrade -y
Hit:1 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Get:2 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Get:3 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]
Get:4 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Packages [15.0 MB]
Get:5 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]
Get:6 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe Translation-en [5982 kB]
Get:7 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Components [3871 kB]
Get:8 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 c-n-f Metadata [301 kB]
Get:9 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 Packages [269 kB]
Get:10 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse Translation-en [118 kB]
Get:11 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 Components [35.0 kB]
Get:12 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 c-n-f Metadata [8328 B]
Get:13 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 Packages [961 kB]
Get:14 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main Translation-en [213 kB]
Get:15 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 Components [151 kB]
Get:16 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 c-n-f Metadata [13.5 kB]
Get:17 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 Packages [1044 kB]
Get:18 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe Translation-en [263 kB]
Get:19 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 Components [365 kB]
Get:20 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 c-n-f Metadata [26.0 kB]
Get:21 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/restricted amd64 Packages [842 kB]
Get:22 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/restricted Translation-en [170 kB]
Get:23 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/restricted amd64 Components [212 B]
Get:24 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/restricted amd64 c-n-f Metadata [492 B]
Get:25 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/multiverse amd64 Packages [21.5 kB]
Get:26 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/multiverse Translation-en [4788 B]
Get:27 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/multiverse amd64 Components [940 B]
Get:28 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/multiverse amd64 c-n-f Metadata [592 B]
Get:29 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/main amd64 Packages [39.1 kB]
Get:30 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/main Translation-en [8676 B]
Get:31 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/main amd64 Components [7056 B]
Get:32 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/main amd64 c-n-f Metadata [272 B]
Get:33 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/universe amd64 Packages [26.4 kB]
Get:34 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/universe Translation-en [16.3 kB]
Get:35 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/universe amd64 Components [15.7 kB]
Get:36 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/universe amd64 c-n-f Metadata [1304 B]
Get:37 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/restricted amd64 Components [216 B]
Get:38 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/restricted amd64 c-n-f Metadata [116 B]
Get:39 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/multiverse amd64 Components [212 B]
Get:40 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/multiverse amd64 c-n-f Metadata [116 B]
Get:41 http://security.ubuntu.com/ubuntu noble-security/main amd64 Packages [711 kB]
Get:42 http://security.ubuntu.com/ubuntu noble-security/main Translation-en [136 kB]
Get:43 http://security.ubuntu.com/ubuntu noble-security/main amd64 Components [8968 B]
```

Figura 9 - Comando "sudo apt update && sudo apt upgrade -y"

```
ubuntu@ip-172-31-32-128:~$ sudo apt update
Hit:1 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Hit:2 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease
Hit:3 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu noble-security InRelease
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
All packages are up to date.
ubuntu@ip-172-31-32-128:~$ sudo apt install software-properties-common
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
software-properties-common is already the newest version (0.99.49.1).
software-properties-common set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
ubuntu@ip-172-31-32-128:~$ sudo add-apt-repository --yes --update ppa:ansible/ansible
Repository: 'Types: deb
URIs: https://ppa.launchpadcontent.net/ansible/ansible/ubuntu/
Suites: noble
Components: main
'
Description:
Ansible is a radically simple IT automation platform that makes your applications and systems easier to deploy. Avoid writing scripts or custom
code to deploy and update your applications- automate in a language that approaches plain English, using SSH, with no agents to install on rem
ote systems.

http://ansible.com/

If you face any issues while installing Ansible PPA, file an issue here:
https://github.com/ansible-community/ppa/issues
More info: https://launchpad.net/~ansible/+archive/ubuntu/ansible
Adding repository.
Hit:1 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Hit:2 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease
Hit:3 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu noble-security InRelease
Get:5 https://ppa.launchpadcontent.net/ansible/ansible/ubuntu noble InRelease [17.8 kB]
Get:6 https://ppa.launchpadcontent.net/ansible/ansible/ubuntu noble/main amd64 Packages [776 B]
Get:7 https://ppa.launchpadcontent.net/ansible/ansible/ubuntu noble/main Translation-en [472 B]
Fetched 19.1 kB in 2s (10.0 kB/s)
Reading package lists... Done
ubuntu@ip-172-31-32-128:~$
```

Figura 10 - Comandos

03/2025

```
ubuntu@ip-172-31-32-128:~$ sudo apt install ansible
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  ansible-core python3-kerberos python3-nacl python3-ntlm-auth python3-paramiko python3-requests-ntlm python3-resolvelib python3-winrm
  python3-xlrd python3-xlsxwriter python3-yaml python3-zipp
Suggested packages:
  python-nacl-doc python3-gssapi python3-invoke
The following NEW packages will be installed:
  ansible ansible-core python3-kerberos python3-nacl python3-ntlm-auth python3-paramiko python3-requests-ntlm python3-resolvelib
  python3-winrm python3-xlrd python3-xlsxwriter python3-yaml python3-zipp
0 upgraded, 11 newly installed, 0 to remove and 0 not upgraded.
Need to get 19.2 MB of archives.
After this operation, 213 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 python3-resolvelib all 1.0.1-1 [25.7 kB]
Get:2 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 python3-kerberos amd64 1.1.14-3.1build9 [21.2 kB]
Get:3 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 python3-nacl amd64 1.5.0-4build1 [57.9 kB]
Get:4 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 python3-ntlm-auth all 1.5.0-1 [21.3 kB]
Get:5 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 python3-paramiko all 2.12.0-2ubuntu4.1 [137 kB]
Get:6 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 python3-requests-ntlm all 1.1.0-3 [6308 B]
Get:7 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 python3-xlrd all 0.13.0-1 [13.4 kB]
Get:8 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 python3-xlsxwriter all 0.4.3-2 [31.9 kB]
Get:9 http://sa-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 python3-yaml amd64 1.09-1 [11.7 kB]
Get:10 https://ppa.launchpadcontent.net/ansible/ansible/ubuntu noble/main amd64 ansible-core all 2.17.10-1ppa-noble [1016 kB]
Get:11 https://ppa.launchpadcontent.net/ansible/ansible/ubuntu noble/main amd64 ansible all 10.7.0-1ppa-noble [17.8 MB]
35% [11 ansible 2658 kB/17.8 MB 15%] 187 kB/s 1min 20s
```

Figura 11 - Comando para instalar o Ansible

Após concluir a instalação do Ansible, foi verificada a versão pelo comando:

```
ubuntu@ip-172-31-32-128:~$ ansible --version
ansible [core 2.17.10]
  config file = /etc/ansible/ansible.cfg
  configured module search path = ['/home/ubuntu/.ansible/plugins/modules', '/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python3/dist-packages/ansible
  ansible collection location = /home/ubuntu/.ansible/collections:/usr/share/ansible/collections
  executable location = /usr/bin/ansible
  python version = 3.12.3 (main, Feb  4 2025, 14:48:35) [GCC 13.3.0] (/usr/bin/python3)
  jinja version = 3.1.2
  libyaml = True
ubuntu@ip-172-31-32-128:~$
```

Figura 12 - Comando "ansible --version"

3.2 Configuração do Inventário

```
ubuntu@ip-172-31-32-128:~$ cd /etc/ansible
ubuntu@ip-172-31-32-128:/etc/ansible$ ls
ansible.cfg  hosts  roles
ubuntu@ip-172-31-32-128:/etc/ansible$
```

Figura 13 - Visualizando pastas

O arquivo de inventário `/etc/ansible/hosts` foi configurado com dois grupos de servidores:

- **[servidores]:** Inclui os endereços das instâncias destinadas à instalação do Nginx.
- **[apache]:** Grupo específico para o servidor que receberá a instalação do Apache.

Cada entrada no inventário especifica o usuário (ubuntu), o caminho da chave privada e o interpretador Python. Para acessar o inventário execute o comando “`sudo nano /etc/ansible/hosts`”.

```
GNU nano 7.2 /etc/ansible/hosts
## [dbservers]
## db01.intranet.mydomain.net
## db02.intranet.mydomain.net
## 10.25.1.56
## 10.25.1.57

# Ex4: Multiple hosts arranged into groups such as 'Debian' and 'openSUSE':

## [Debian]
## alpha.example.org
## beta.example.org

## [openSUSE]
## green.example.com
## blue.example.com

[servidores]
ec2-18-229-142-92.sa-east-1.compute.amazonaws.com ansible_user=ubuntu ansible_ssh_private_key_file=/home/ubuntu/ServidorNginx1.pem ansible_python_interpreter=/usr/bin/python3
ec2-15-228-166-136.sa-east-1.compute.amazonaws.com ansible_user=ubuntu ansible_ssh_private_key_file=/home/ubuntu/ServidorNginx2.pem ansible_python_interpreter=/usr/bin/python3

[apache]
ec2-15-228-166-136.sa-east-1.compute.amazonaws.com ansible_user=ubuntu ansible_ssh_private_key_file=/home/ubuntu/ServidorNginx2.pem ansible_python_interpreter=/usr/bin/python3

^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location   ^U Undo       ^M Set Mark   ^] To Bracket  ^_ Previous   ^R Back
^X Exit      ^R Read File  ^E Replace    ^J Paste      ^_ Justify    ^G Go To Line ^I Redo       ^H Copy       ^F Where Was  ^N Next       ^S Forward
```

Figura 14 - Arquivo de Inventário

03/2025

Após configurar o inventário, foi testada a conexão com os servidores pelos comandos:

```
ubuntu@ip-172-31-32-128:~$ ansible servidores -m ping
ec2-15-228-166-136.sa-east-1.compute.amazonaws.com | SUCCESS => {
  "changed": false,
  "ping": "pong"
}
ec2-18-229-142-92.sa-east-1.compute.amazonaws.com | SUCCESS => {
  "changed": false,
  "ping": "pong"
}
ubuntu@ip-172-31-32-128:~$ |
```

Figura 15 - Testando conexões com servidores

```
ubuntu@ip-172-31-32-128:~$ ansible servidores -a "uptime"
ec2-18-229-142-92.sa-east-1.compute.amazonaws.com | CHANGED | rc=0 >>
  04:28:13 up  3:53,  1 user,  load average: 0.00, 0.00, 0.00
ec2-15-228-166-136.sa-east-1.compute.amazonaws.com | CHANGED | rc=0 >>
  04:28:13 up  3:52,  1 user,  load average: 0.08, 0.02, 0.01
ubuntu@ip-172-31-32-128:~$ |
```

Figura 16 - Comando "ansible servidores -a "uptime""

```
ubuntu@ip-172-31-32-128:~$ ansible servidores -a "ip a"
ec2-15-228-166-136.sa-east-1.compute.amazonaws.com | CHANGED | rc=0 >>
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: enX0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 9001 qdisc fq_codel state UP group default qlen 1000
    link/ether 0a:fa:f3:2c:fa:9f brd ff:ff:ff:ff:ff:ff
    inet 172.31.39.109/20 metric 100 brd 172.31.47.255 scope global dynamic enX0
        valid_lft 2091sec preferred_lft 2091sec
    inet6 fe80::8fa:f3ff:fe2c:fa9f/64 scope link
        valid_lft forever preferred_lft forever
ec2-18-229-142-92.sa-east-1.compute.amazonaws.com | CHANGED | rc=0 >>
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: enX0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 9001 qdisc fq_codel state UP group default qlen 1000
    link/ether 0a:55:85:f9:ac:17 brd ff:ff:ff:ff:ff:ff
    inet 172.31.40.191/20 metric 100 brd 172.31.47.255 scope global dynamic enX0
        valid_lft 2051sec preferred_lft 2051sec
    inet6 fe80::855:85ff:fef9:ac17/64 scope link
        valid_lft forever preferred_lft forever
ubuntu@ip-172-31-32-128:~$ |
```

Figura 17 - Comando "ansible servidores -a "ip a""

4 Criação e Execução dos Playbooks

4.1 Playbook para instalar o Nginx

Para criar o arquivo playbook, execute o comando: "sudo nano playbook-nginx.yml". Dentro do arquivo coloque o seguinte código:

03/2025

```
ubuntu@ip-172-31-32-128: /el  X  +  v
GNU nano 7.2                                     playbook-nginx.yml
---
- name: Instalar o nginx
  hosts: servidores
  become: true
  tasks:
    - name: Instalar nginx
      apt: # Usando apt para sistemas baseadas em Debian (como Ubuntu)
        name: nginx
        state: present
```

Figura 18 - Código Playbook Nginx

Lembrando que para os arquivos .yaml ou .yml o código deve estar bem indentado para a leitura correta, caso contrário pode acontecer erros na execução.

Para executar o playbook execute o comando:

```
ubuntu@ip-172-31-32-128:/etc/ansible$ sudo nano playbook-nginx.yml
ubuntu@ip-172-31-32-128:/etc/ansible$ ansible-playbook playbook-nginx.yml

PLAY [Instalar o nginx] *****

TASK [Gathering Facts] *****
ok: [ec2-15-228-166-136.sa-east-1.compute.amazonaws.com]
ok: [ec2-18-229-142-92.sa-east-1.compute.amazonaws.com]

TASK [Instalar nginx] *****
changed: [ec2-18-229-142-92.sa-east-1.compute.amazonaws.com]
changed: [ec2-15-228-166-136.sa-east-1.compute.amazonaws.com]

PLAY RECAP *****
ec2-15-228-166-136.sa-east-1.compute.amazonaws.com : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
ec2-18-229-142-92.sa-east-1.compute.amazonaws.com : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

ubuntu@ip-172-31-32-128:/etc/ansible$ |
```

Figura 19 - Executando Playbook Nginx

4.2 Playbook para instalar o Apache

Para criar o arquivo playbook, execute o comando: “sudo nano playbook-apache.yml”. Dentro do arquivo coloque o seguinte código:

```
ubuntu@ip-172-31-32-128: /el  X  +  v
GNU nano 7.2                                     playbook-apache.yml *
---
- name: Instalar o apache
  hosts: apache
  become: true
  tasks:
    - name: Instalar apache
      apt: # Usando apt para instalar o apache no sistema
        name: apache2
        update_cache: yes
        state: latest|
```

Figura 20 - Código Playbook Apache

Para executar o playbook execute o comando:

```
ubuntu@ip-172-31-32-128:/etc/ansible$ ansible-playbook playbook-apache.yml

PLAY [Instalar o apache] *****

TASK [Gathering Facts] *****
ok: [ec2-15-228-166-136.sa-east-1.compute.amazonaws.com]

TASK [Instalar apache] *****
changed: [ec2-15-228-166-136.sa-east-1.compute.amazonaws.com]

PLAY RECAP *****
ec2-15-228-166-136.sa-east-1.compute.amazonaws.com : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

ubuntu@ip-172-31-32-128:/etc/ansible$ |
```

Figura 21 - Executando Playbook Apache

5 Resultados

Após a execução dos playbooks, foram observadas as seguintes validações:

- Resposta positiva dos módulos Ansible (por meio dos comandos de verificação como ping e uptime)
- Instalação bem-sucedida dos pacotes Nginx e Apache nas instâncias correspondentes
- Logs e prints evidenciam que os servidores responderam conforme o esperado, garantindo a eficácia da automação e a configuração correta do ambiente.

6 Conclusão

O relatório evidencia que o processo de criação e automação dos servidores na AWS utilizando Ansible foi concluído com sucesso. Através da configuração adequada do inventário, transferência segura das chaves e execução dos playbooks, foi possível automatizar a instalação dos serviços web. Este procedimento não só demonstrou a viabilidade da utilização do Ansible em ambientes cloud, mas também reforçou a importância de uma documentação detalhada para futuras manutenções e expansões do sistema.