

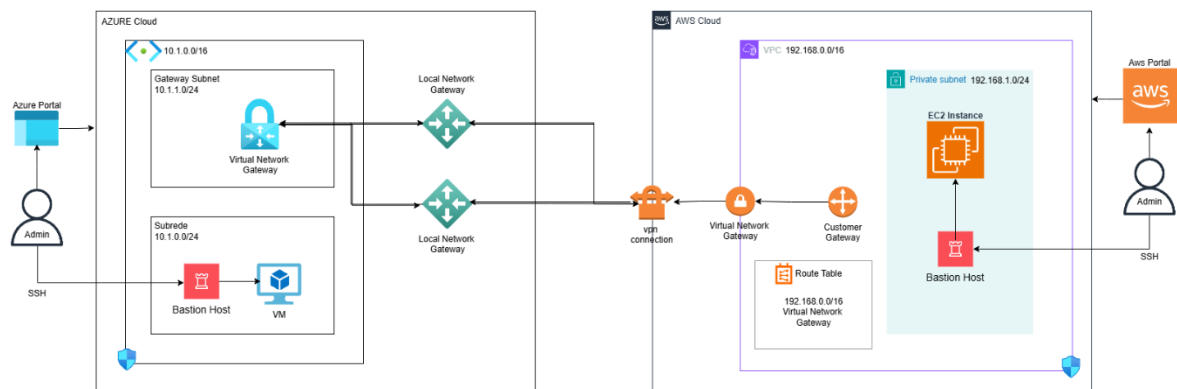
# Lucas Duarte Pires

## Projeto Multi-Cloud Azure & Aws

### Integração Multi-Cloud Azure + AWS via VPN Site-to-Site

Criar um ambiente multi-cloud entre Microsoft Azure e Amazon Web Services (AWS), estabelecendo uma VPN site-to-site que permita a comunicação entre máquinas virtuais usando apenas endereços IP privados.

### Topologia



### Tabela de IPs

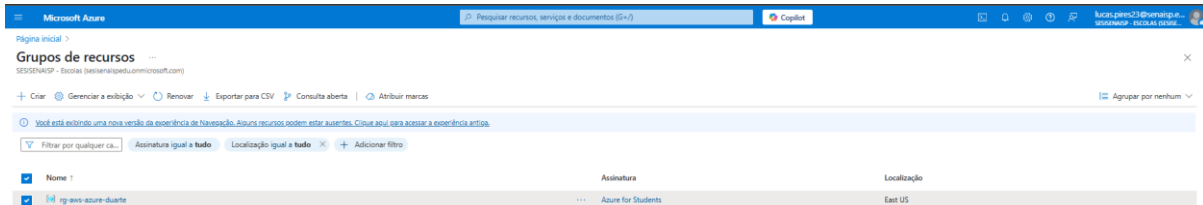
Multi-Cloud Tabela							
Nuvem	Rede	Sub-rede	Nome-VM	IP-Público	IP-Privado	Função-Vm	Tipo-Vm
Azure	10.1.0.0/16	10.1.0.0/24	bastion-host	52.226.121.135	10.1.0.7	Bastion-Host	Standard_B2s
Azure	10.1.0.0/16	10.1.0.0/24	vm-azure	-----	10.1.0.4	Internal VM	Standard_B2s
Aws	192.168.0.0/16	192.168.1.0/24	bastion-host	204.236.248.28	192.168.1.50	Bastion-Host	t2.micro
Aws	192.168.0.0/16	192.168.1.0/24	vm-duarte	-----	192.168.1.116	Internal EC2	t2.micro

Legenda da Tabela	
Nuvem	Nome da provedora de nuvem utilizada (ex: AWS, Azure).
Rede	Endereço da rede principal (ex: 10.1.0.0/16), também chamada de VNet (Azure) ou VPC (AWS).
Sub-rede	Intervalo de IPs atribuído à sub-rede (ex: 10.1.0.0/24), onde a VM está alocada.
Nome-VM	Nome da máquina virtual definido durante sua criação (ex: vm-duarte).
IP-Público	Endereço IP acessível pela internet. Pode mudar se for dinâmico (não estático).
IP-Privado	Endereço IP interno atribuído à VM na rede/sub-rede. Usado para comunicação interna.
Função-VM	Tipo/tamanho da VM, que define seus recursos de hardware (ex: t2.micro, Standard_B2s).
Tipo-VM	Finalidade da máquina na arquitetura.

# Azure

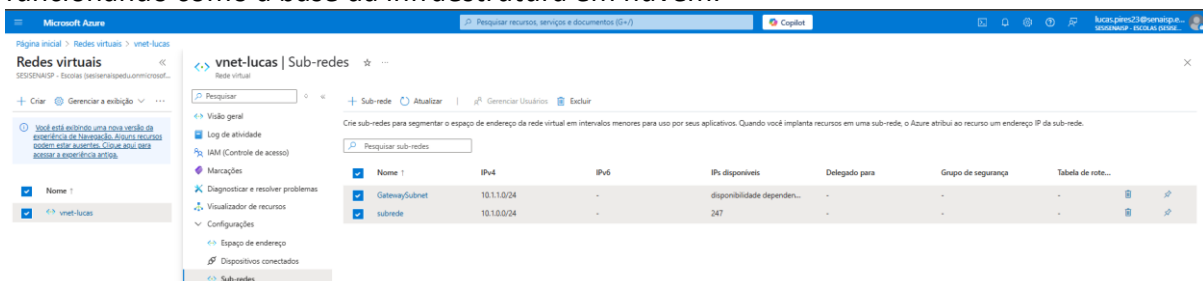
## 1º Passo – Criar o Grupo de Recursos

A criação do grupo de recursos é o ponto de partida para organizar e gerenciar todos os serviços que serão utilizados no Azure. Ele serve como um contêiner lógico para manter os recursos relacionados em um único local.



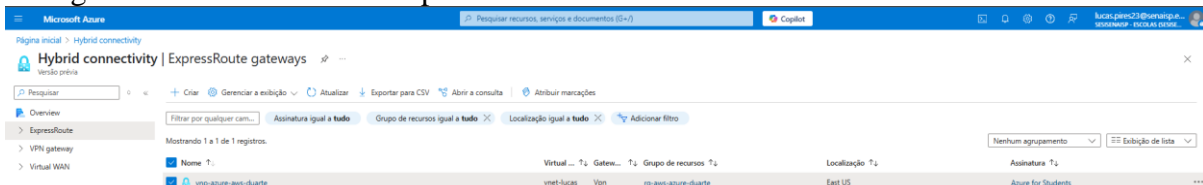
## 2º Passo – Criar a Rede Virtual

A rede virtual (VNet) é fundamental para a comunicação entre os recursos no Azure. Ela permite definir um espaço de endereçamento IP, sub-redes e configurações de segurança, funcionando como a base da infraestrutura em nuvem.



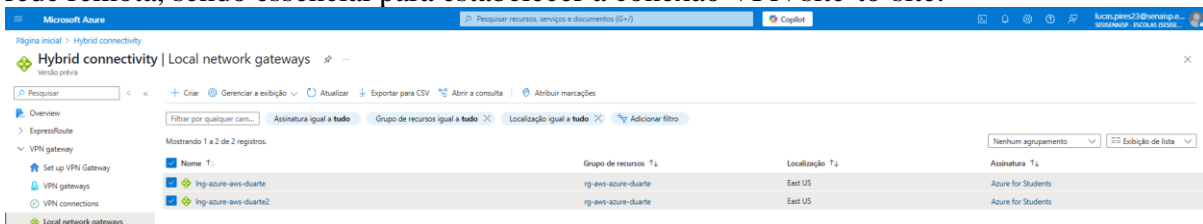
## 3º Passo – Criar o Virtual Network Gateway

O Virtual Network Gateway é responsável por estabelecer a comunicação segura entre a rede virtual do Azure e outras redes, como ambientes locais ou outras VNets. Ele é essencial para configurar conexões VPN ou ExpressRoute.



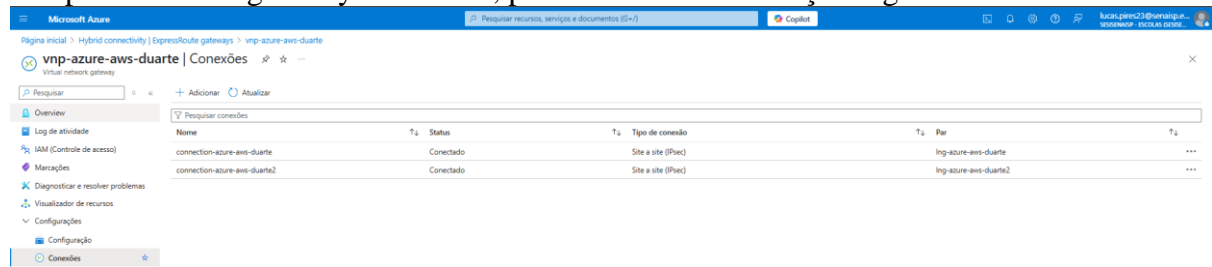
## 4º Passo – Criar o Local Network Gateway

O Local Network Gateway representa a rede local (on-premises) no Azure. Ele armazena informações como o endereço IP público do gateway local e o espaço de endereçamento da rede remota, sendo essencial para estabelecer uma conexão VPN site-to-site.



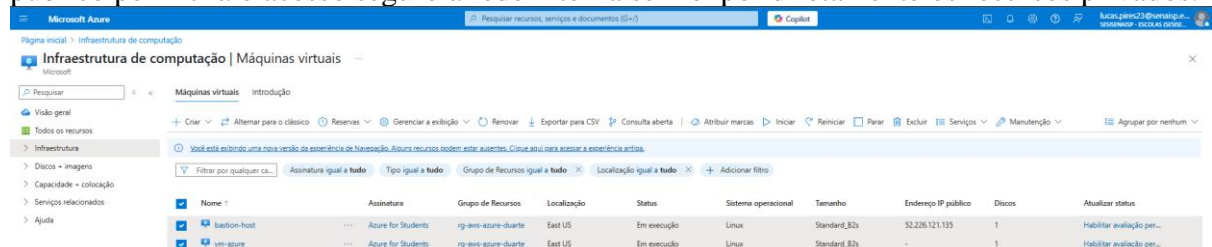
## 5º Passo – Criar a Conexão no Virtual Network Gateway

A conexão no Virtual Network Gateway é o que liga a rede virtual do Azure à rede local. Nessa etapa, são definidas as configurações da VPN, como tipo de conexão, chave compartilhada e o gateway de destino, permitindo a comunicação segura entre os ambientes.



## 6º Passo – Criar Máquinas Virtuais (IP Privado e Bastion Host)

Nesta etapa, serão criadas duas máquinas virtuais: uma com IP privado, que simula um ambiente interno, e outra com IP público, que servirá como Bastion Host. A máquina com IP público permitirá o acesso seguro à rede interna sem expor diretamente os recursos privados.



## 7º Passo – Gerar e Configurar a Chave de Acesso no Bastion Host

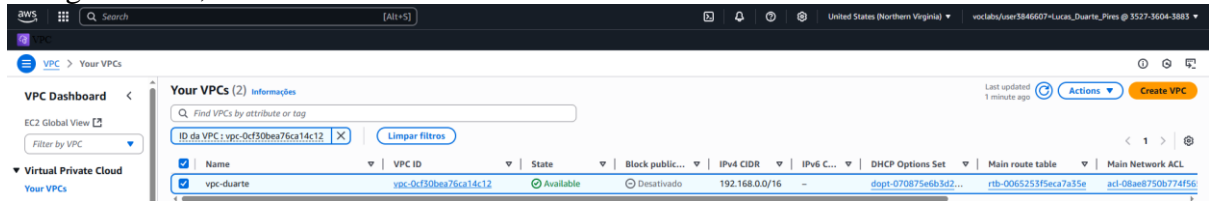
No Bastion Host, será gerada e configurada a chave de acesso necessária para conectar com segurança à máquina virtual privada. Após atualizar os pacotes, criaremos um arquivo para armazenar a chave, aplicaremos as permissões corretas e, por fim, utilizaremos o comando SSH para estabelecer a conexão com a VM interna por meio da chave gerada.

```
azureuser@vm-azure: ~  
azureuser@bastion-host:~$ nano vm-azure_key  
azureuser@bastion-host:~$ chmod 400 vm-azure_key  
azureuser@bastion-host:~$ ssh -i vm-azure_key azureuser@10.1.0.4  
The authenticity of host '10.1.0.4 (10.1.0.4)' can't be established.  
ED25519 key fingerprint is SHA256:sCafML3/odSVL9p2jJZSHY0fYompWYFJvt62pDdwuQg.  
This key is not known by any other names.  
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes  
Warning: Permanently added '10.1.0.4' (ED25519) to the list of known hosts.  
Linux vm-azure 6.1.0-37-cloud-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.140-1 (2025-05-22) x86_64  
  
The programs included with the Debian GNU/Linux system are free software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/*/copyright.  
  
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent  
permitted by applicable law.  
azureuser@vm-azure:~$ |
```

# Aws

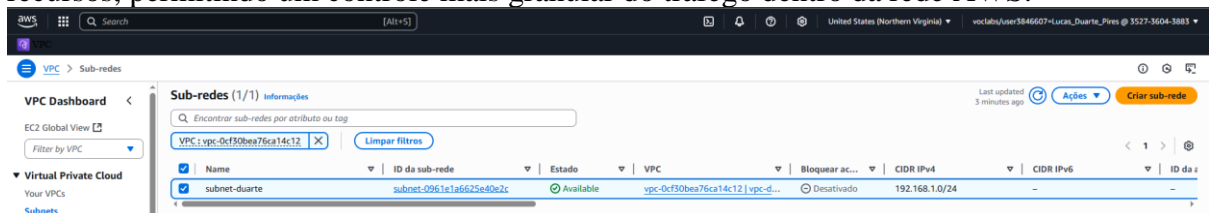
## 1º Passo – Criar a VPC

A VPC (Virtual Private Cloud) é a rede virtual dentro da AWS onde serão implantados os recursos. Ela permite definir um espaço de endereçamento IP personalizado e controlar o tráfego de rede, funcionando como a base da infraestrutura em nuvem.



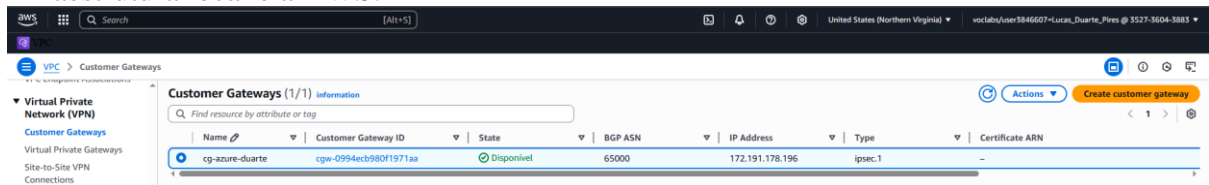
## 2º Passo – Criar uma Subnet

A subnet é uma subdivisão da VPC que segmenta o espaço de IPs para organizar e isolar recursos, permitindo um controle mais granular do tráfego dentro da rede AWS.



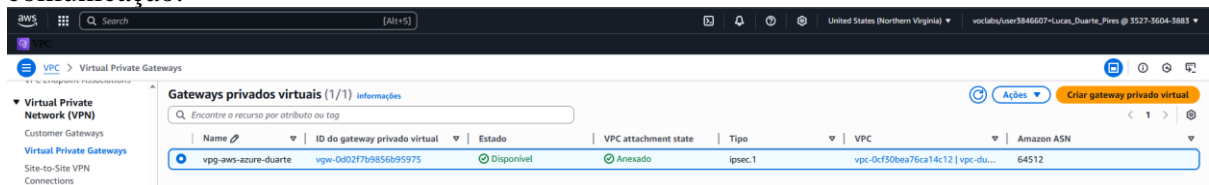
## 3º Passo – Criar o Customer Gateway

O Customer Gateway representa o dispositivo ou a rede local do cliente no ambiente AWS. Ele armazena as informações necessárias para estabelecer a conexão VPN entre a infraestrutura local e a AWS.



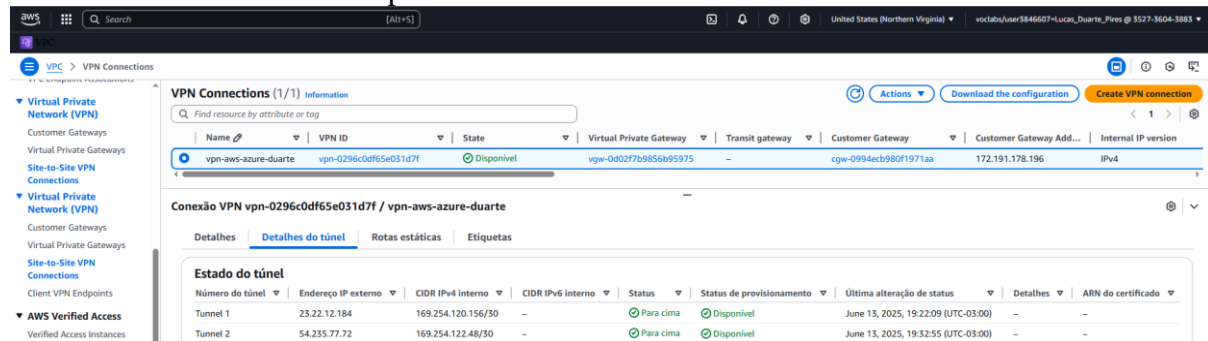
## 4º Passo – Criar a Virtual Private Gateway e Anexar à VPC

A Virtual Private Gateway atua como o ponto de conexão do lado da AWS para a rede privada, permitindo que a VPC se comunique com redes externas, como a infraestrutura local, por meio de VPNs. Após criada, ela deve ser anexada à VPC para estabelecer essa comunicação.



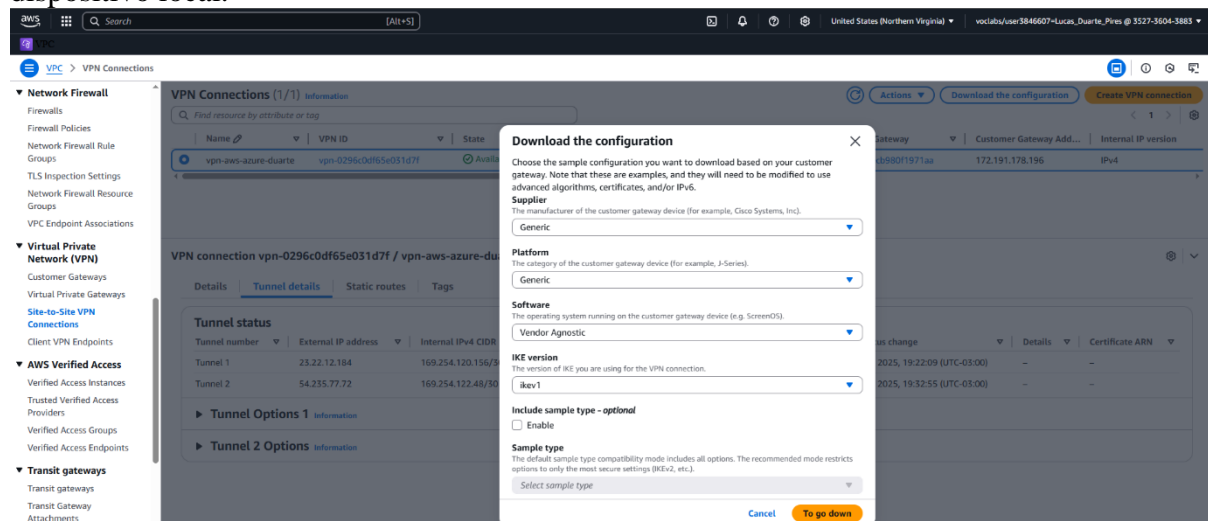
## 5º Passo – Criar a Conexão VPN Site-to-Site

A conexão VPN site-to-site estabelece um túnel seguro entre a rede local do cliente e a AWS, permitindo a comunicação privada e criptografada entre os ambientes, integrando as infraestruturas de forma transparente.



## 6º Passo – Download do Arquivo de Configuração e Dados da Conexão VPN

Após criar a conexão VPN, faça o download do arquivo de configuração gerado pela AWS. Nele, constam informações essenciais para o estabelecimento do túnel seguro, como as chaves pré-compartilhadas (Pre-Shared Keys) e os endereços dos Virtual Private Gateways para cada túnel da conexão. Esses dados são necessários para configurar corretamente o dispositivo local.



## 7º Passo – Criar e Associar a Route Table ao Virtual Private Gateway

Nesta etapa, será criada e configurada a tabela de rotas da VPC para direcionar o tráfego destinado à rede local através do Virtual Private Gateway, garantindo que os pacotes sejam encaminhados corretamente pela conexão VPN estabelecida.

The screenshot shows the AWS VPC console interface. The top navigation bar includes the AWS logo, a search bar, and the user's account information. The left sidebar contains the VPC Dashboard, EC2 Global View, and Virtual Private Cloud sections. The main content area displays the 'Route tables (2)' list. A table lists the route tables, with 'rtb-duarte' selected. Below the table, the 'Routes (3)' section is visible, showing a list of routes. The 'Edit broken' modal is open, allowing the user to modify the routes. The modal includes a table with columns for Destination, Target, Status, and Propagated. The routes listed are: 192.168.0.0/16 to local (Active, No), 10.1.0.0/24 to Virtual Private Gateway (Active, No), and 0.0.0.0/0 to Gateway and Internet (Active, No). The 'Add route' button is at the bottom left of the modal, and 'Cancel', 'Preview', and 'Save changes' buttons are at the bottom right.

Destination	Target	Status	Propagated
192.168.0.0/16	local	Active	No
10.1.0.0/24	Virtual Private Gateway	Active	No
0.0.0.0/0	Gateway and Internet	Active	No

## 8º Passo – Criar Máquinas Virtuais (IP Privado e Bastion Host)

Nesta etapa, serão criadas duas máquinas virtuais: uma com IP privado, que simula um ambiente interno, e outra com IP público, que servirá como Bastion Host. A máquina com IP público permitirá o acesso seguro à rede interna sem expor diretamente os recursos privados.

The screenshot shows the AWS EC2 console interface. The top navigation bar includes the AWS logo, a search bar, and the user's account information. The left sidebar contains the EC2 section. The main content area displays the 'Instances (2/2)' list. A table lists the instances, with 'vm-duarte' and 'bastion-host' selected. Below the table, the 'Edit broken' modal is open, allowing the user to modify the instances. The modal includes a table with columns for Name, Instance ID, Instance status, Instance type, Status check, Alarm status, Availability zone, Public IPv4 DNS, Public IPv4, and Elastic IP. The instances listed are: 1. vm-duarte (Running, t2.micro, 2/2 checks passed, us-east-1a, -, -, -), 2. bastion-host (Running, t2.micro, 2/2 checks passed, us-east-1a, -, 34.238.172.204, -). The 'Add instance' button is at the bottom left of the modal, and 'Cancel', 'Preview', and 'Save changes' buttons are at the bottom right.

Name	Instance ID	Instance status	Instance type	Status check	Alarm status	Availability zone	Public IPv4 DNS	Public IPv4	Elastic IP
vm-duarte	i-039aa0972cc7e0bc	Running	t2.micro	2/2 checks passed	Display alarms	us-east-1a	-	-	-
bastion-host	i-058ffcd29d0787b	Running	t2.micro	2/2 checks passed	Display alarms	us-east-1a	-	34.238.172.204	-

## 9º Passo – Gerar e Configurar a Chave de Acesso no Bastion Host

No Bastion Host, será gerada e configurada a chave de acesso necessária para conectar com segurança à máquina virtual privada. Após atualizar os pacotes, criaremos um arquivo para armazenar a chave, aplicaremos as permissões corretas e, por fim, utilizaremos o comando SSH para estabelecer a conexão com a VM interna por meio da chave gerada.

```
ubuntu@ip-192-168-1-116: ~
ubuntu@ip-192-168-1-50:~$ nano labuser.pem
ubuntu@ip-192-168-1-50:~$ chmod 400 labuser.pem
ubuntu@ip-192-168-1-50:~$ ssh -i labuser.pem ubuntu@192.168.1.116
Welcome to Ubuntu 24.04.2 LTS (GNU/Linux 6.8.0-1029-aws x86_64)

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

System information as of Sat Jun 14 00:45:52 UTC 2025

System load:  0.0           Processes:      105
Usage of /:   25.7% of 6.71GB Users logged in: 0
Memory usage: 21%          IPv4 address for enX0: 192.168.1.116
Swap usage:   0%

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

Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check your Internet connection or proxy settings

Last login: Sat Jun 14 00:43:36 2025 from 192.168.1.50
ubuntu@ip-192-168-1-116:~$
```

## Testes

**Azure:** Realizando teste de Ping para o IP da máquina AWS (192.168.1.116) para verificar a comunicação entre as redes.

```
azureuser@vm-azure: ~
azureuser@vm-azure:~$ ping 192.168.1.116
PING 192.168.1.116 (192.168.1.116) 56(84) bytes of data.
64 bytes from 192.168.1.116: icmp_seq=1 ttl=64 time=7.14 ms
64 bytes from 192.168.1.116: icmp_seq=2 ttl=64 time=10.5 ms
64 bytes from 192.168.1.116: icmp_seq=3 ttl=64 time=5.90 ms
64 bytes from 192.168.1.116: icmp_seq=4 ttl=64 time=5.74 ms
64 bytes from 192.168.1.116: icmp_seq=5 ttl=64 time=7.99 ms
^C
--- 192.168.1.116 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4006ms
rtt min/avg/max/mdev = 5.736/7.456/10.518/1.739 ms
azureuser@vm-azure:~$ |
```

**Aws:** Realizando teste de Ping para o IP da máquina Azure (10.1.0.4) para confirmar a conectividade entre os ambientes.

```
ubuntu@ip-192-168-1-116: ~  
ubuntu@ip-192-168-1-116:~$ ping 10.1.0.4  
PING 10.1.0.4 (10.1.0.4) 56(84) bytes of data.  
64 bytes from 10.1.0.4: icmp_seq=1 ttl=64 time=8.56 ms  
64 bytes from 10.1.0.4: icmp_seq=2 ttl=64 time=5.96 ms  
64 bytes from 10.1.0.4: icmp_seq=3 ttl=64 time=5.46 ms  
64 bytes from 10.1.0.4: icmp_seq=4 ttl=64 time=5.47 ms  
64 bytes from 10.1.0.4: icmp_seq=5 ttl=64 time=6.81 ms  
^C  
--- 10.1.0.4 ping statistics ---  
5 packets transmitted, 5 received, 0% packet loss, time 4004ms  
rtt min/avg/max/mdev = 5.460/6.453/8.556/1.160 ms  
ubuntu@ip-192-168-1-116:~$
```