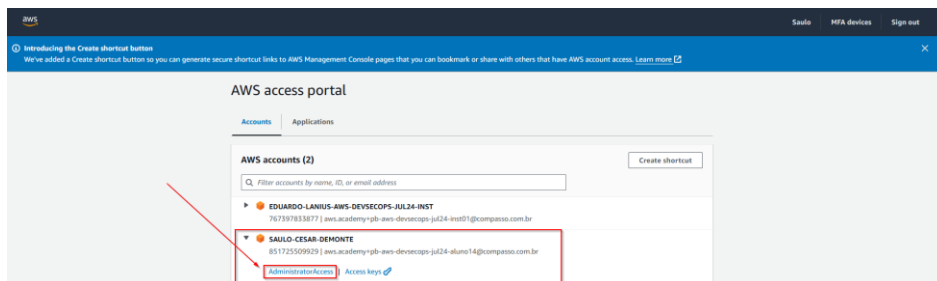
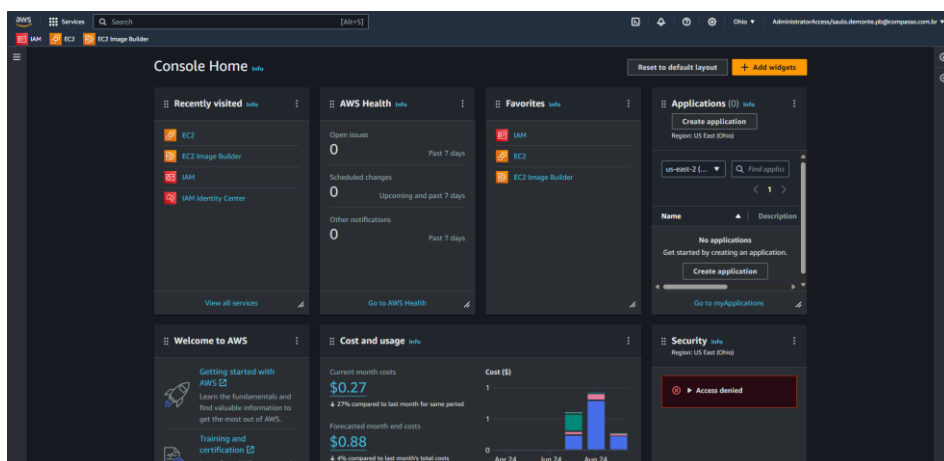


## #Acessando o Console AWS

1. Acessar o console [AWS access portal \(awsapps.com\)](https://awsapps.com) e fazer o login com a conta Compass;
2. Ao acessar o console AWS entre em sua conta;



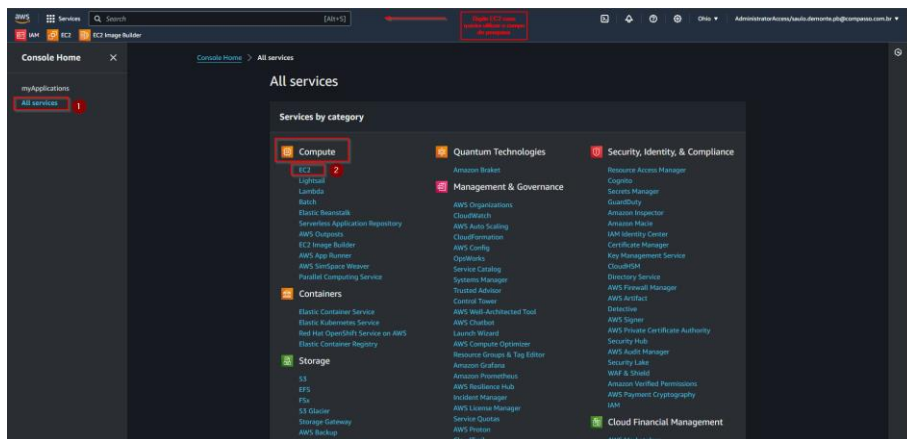
3. A próxima tela será a página Home do console que pode variar dependendo de sua personalização;



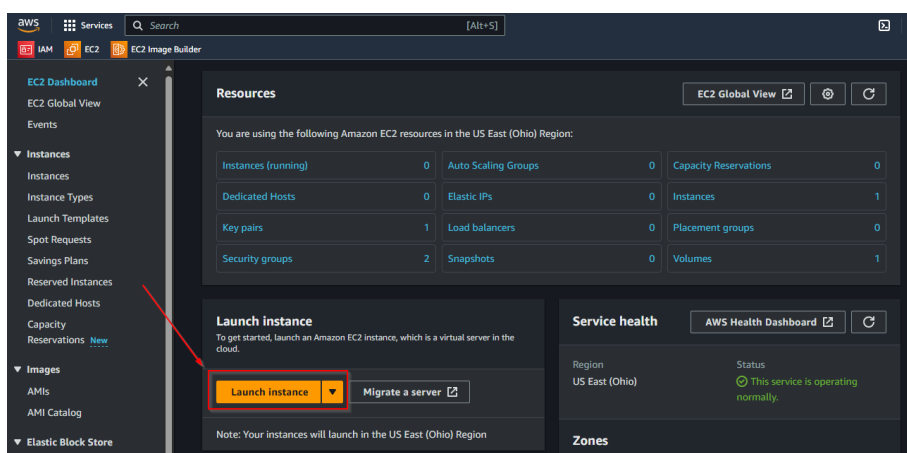
## #Criando uma instância EC2

Para criar uma instância EC2 (Elastic Compute Cloud) na AWS, você pode seguir o passo a passo abaixo:

1. No console, vá para o serviço EC2. Para isso, clique em *All Services* no canto superior esquerdo e, em seguida, escolha *EC2* na seção *Compute*. Ou utilize o campo de pesquisa na parte superior da tela digitando *EC2*



2. No painel do EC2, clique no botão *Launch Instance*.

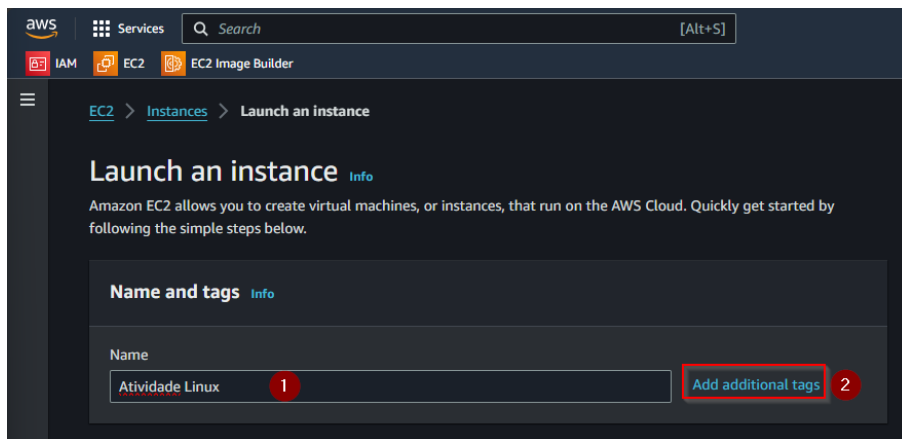


3. Na próxima página, você precisará fornecer um nome para a instância e escolher algumas configurações de TAGs

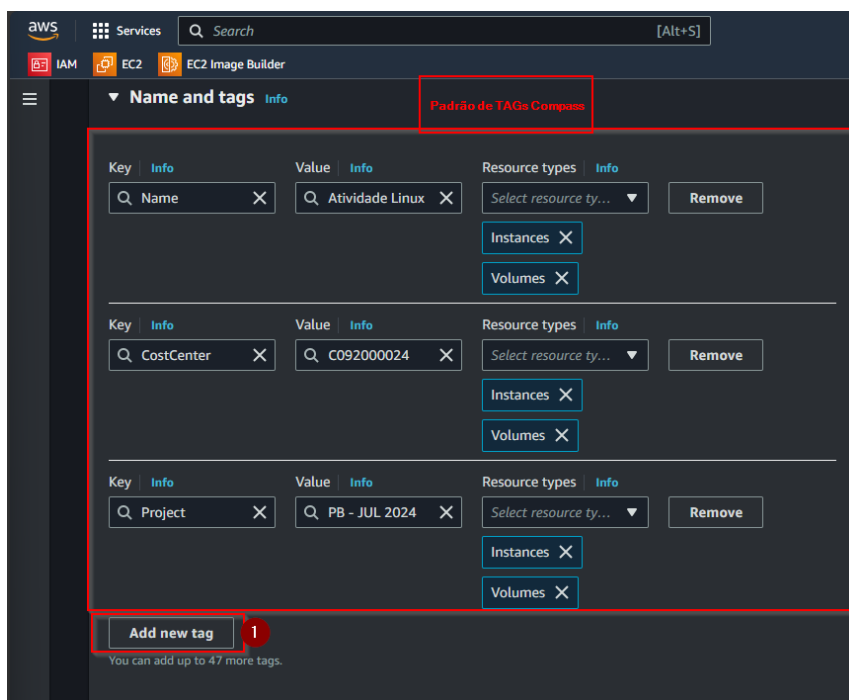
Por que usar TAGs?

- Organização: TAGs ajudam a organizar recursos, permitindo categorizá-los por projeto, departamento, ambiente (produção, teste etc.) ou qualquer outro critério relevante.
- Gerenciamento de custos: Ao atribuir TAGs a recursos, você pode rastrear e associar custos a diferentes departamentos, projetos ou equipes, facilitando a alocação de custos.
- Busca e filtragem: TAGs facilitam a busca e filtragem de recursos dentro do console AWS, especialmente quando você tem muitos recursos.
- Automação: Tags podem ser usadas em scripts e automações para identificar e operar em um conjunto específico de recursos.

4. Escolha um nome para sua atividade, no caso usarei *Atividade Linux* e em seguida clique em vá em *Add additional tags*



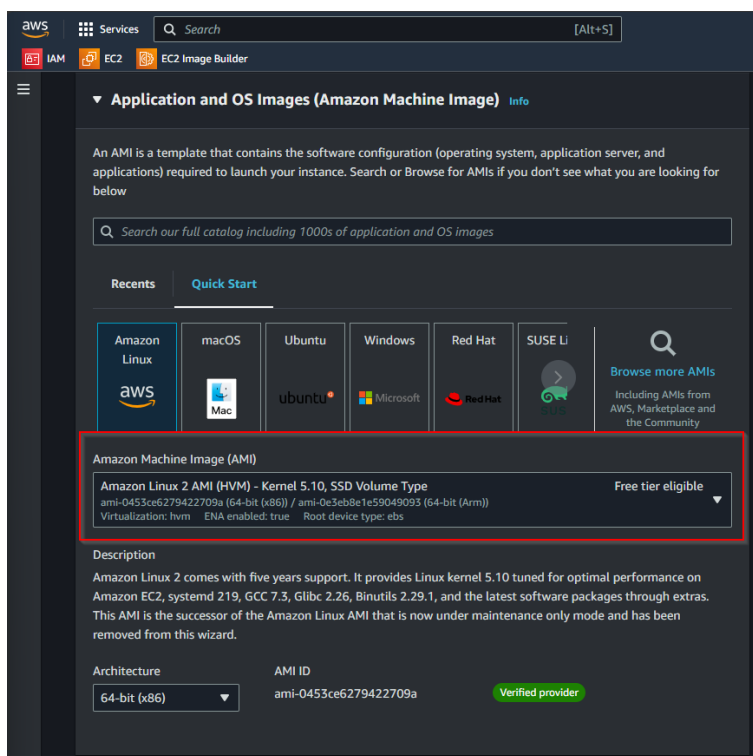
5. Na tela aberta crie as TAGs de acordo com sua necessidade na opção Add new tag ( 1 ), para os projetos da Compass usamos o padrão da imagem abaixo.



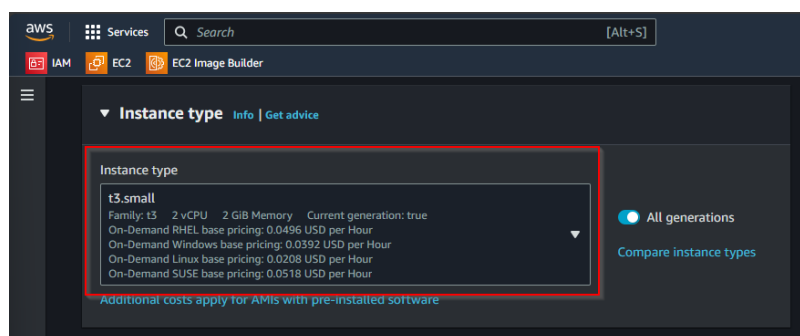
A AMI é o sistema operacional e o software pré-configurado que a sua instância EC2 usará.

6. Em *Application and OS Images (Amazon Machine Image - AMI)*, você pode escolher uma imagem baseada em Linux (Ubuntu, Amazon Linux, Red Hat etc.) ou Windows. Neste caso usaremos a imagem solicitada no projeto

“Criar 1 instância EC2 com o sistema operacional Amazon Linux 2”



7. Em *Instance type* nosso projeto solicita uma configuração “*Família t3.small*”



Para gerar uma chave pública para acesso ao ambiente AWS, você precisa criar um par de chaves (chave pública e chave privada) que será usado para autenticação.

8. Ao criar uma instância EC2, você será solicitado a escolher ou criar um par de chaves na seção *Key Pair (login)*. Vamos criar uma nova.

▼ Key pair (login) Info

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - required

Select

Q |

Proceed without a key pair (Not recommended) Default value

PB-JUL-2024

Type: rsa

Create new key pair

Edit

- a. Nomeie o par de chaves e selecione o formato de arquivo:
- PEM (usado para Linux/Unix).
  - PPK (usado para Windows com PuTTY).

Criaremos uma *PEM* pois utilizaremos para acesso no Linux

- b. Devemos escolher o tipo de criptografia das chaves.
- RSA (Maior compatibilidade, menor velocidade e menos seguro com relação ao ED25519)
  - ED25519 (Menor compatibilidade, maior velocidade, mais moderno e seguro se comparado ao RSA)

Como queremos ter a certeza de compatibilidade usaremos o *RSA* que não é inseguro e sim apenas tem uma segurança e velocidade inferior ao ED25519

Escolha as opções e em seguida *Create key pair* conforme imagem abaixo

Create key pair

Key pair name

Key pairs allow you to connect to your instance securely.

AtvLinux

The name can include up to 255 ASCII characters. It can't include leading or trailing spaces.

Key pair type

☒ RSA  
RSA encrypted private and public key pair

☐ ED25519  
ED25519 encrypted private and public key pair

Private key file format

☒ .pem  
For use with OpenSSH

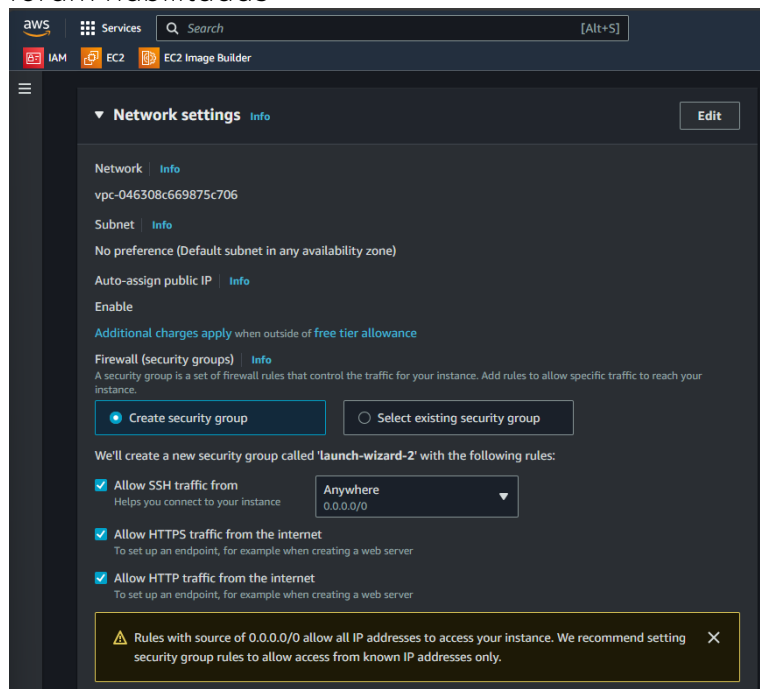
☐ .ppk  
For use with PuTTY

⚠ When prompted, store the private key in a secure and accessible location on your computer. You will need it later to connect to your instance. [Learn more](#)

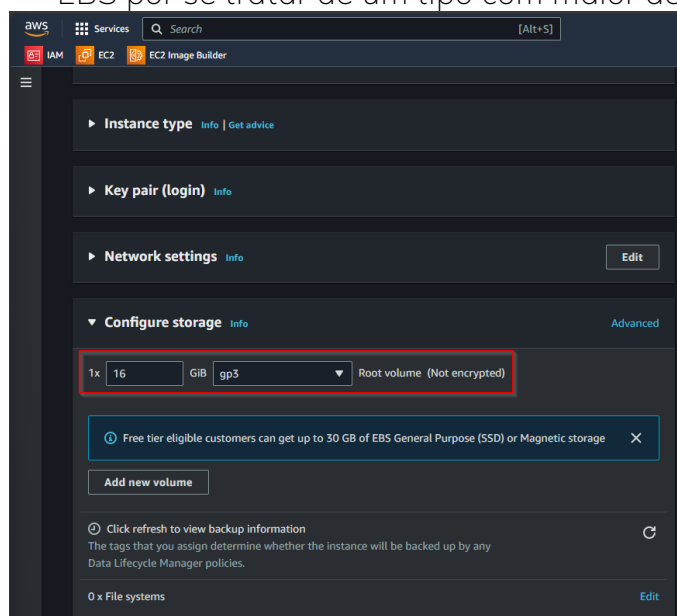
Cancel Create key pair

**ATENÇÃO:** Clique em *Download Key Pair* para salvar a chave privada em seu computador. Essa é a única chance de baixar a chave privada.

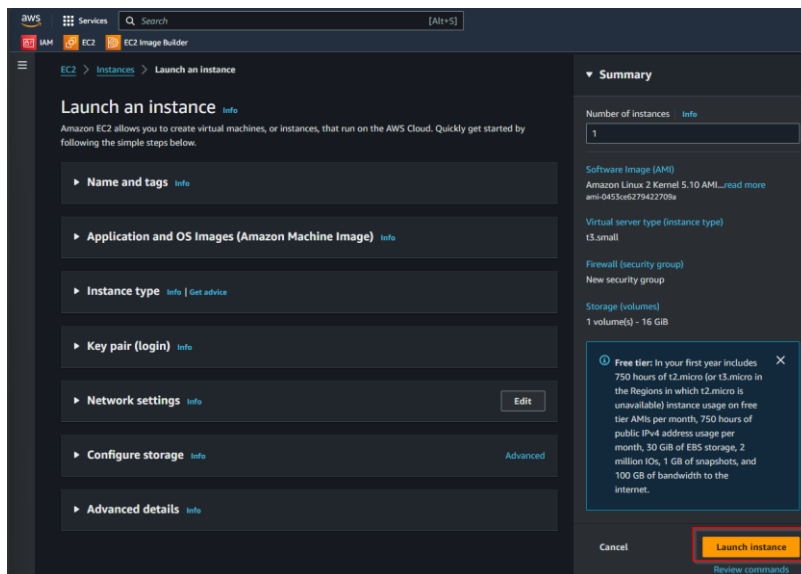
9. Devido ao projeto envolver acesso a um servidor web as opções *Allow HTTPS traffic from de internet* e *Allow HTTP traffic from de internet* foram habilitadas



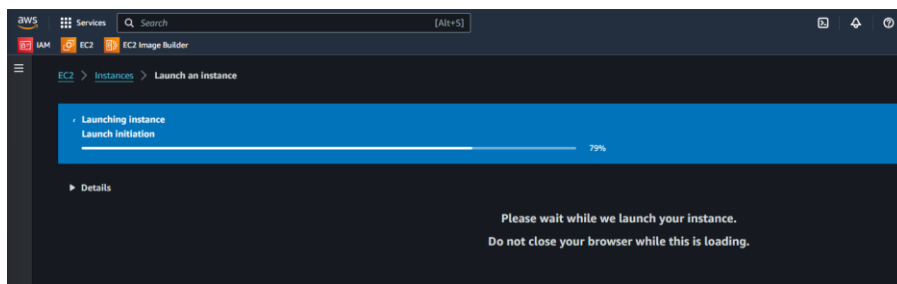
10. O próximo passo é realizar a configuração de armazenamento e para este projeto usaremos *16GBs*, também optei pelo *GP3* no tipo de volume do EBS por se tratar de um tipo com maior desempenho e menor custo



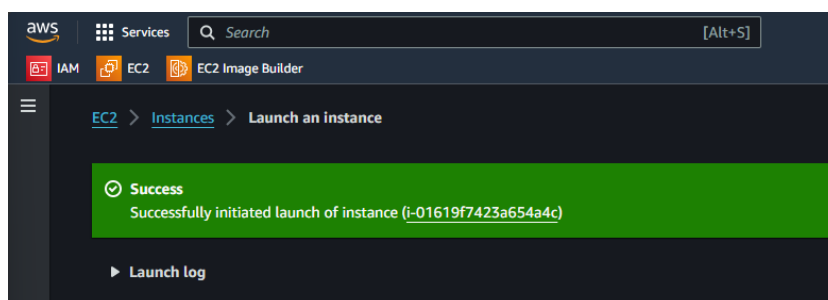
11. Após realizar a configuração, clicar em *Launch Instance* para criar a nossa instância EC2 com as configurações necessárias para prosseguir com projeto



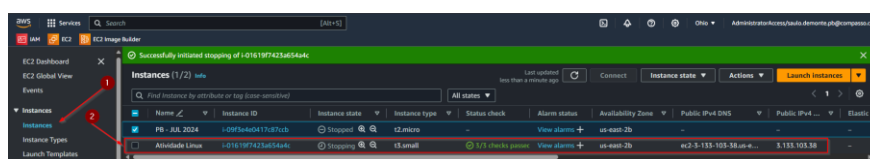
12. Aguarde a criação da instância, é para ser rápido.



13. A mensagem abaixo mostra sucesso na criação da instância



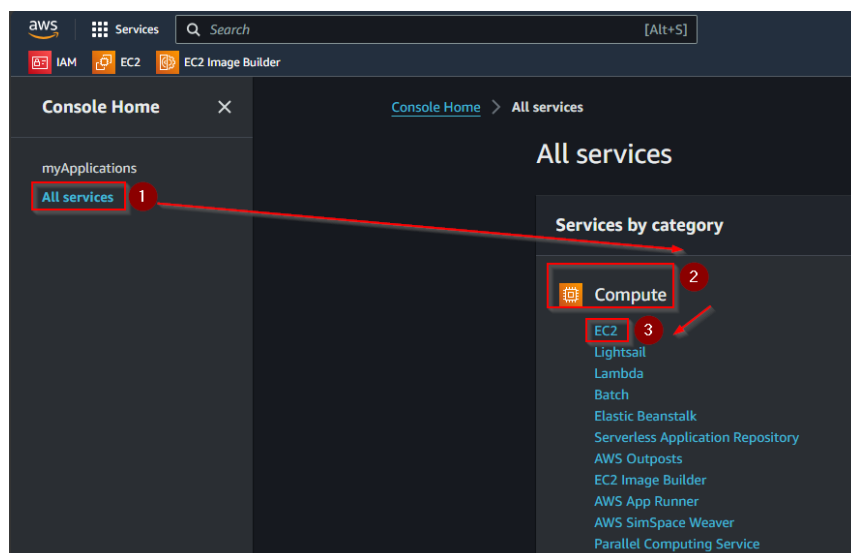
14. No menu esquerdo, acesse instância e verá que agora sua instância está criada e pronta para ser utilizada



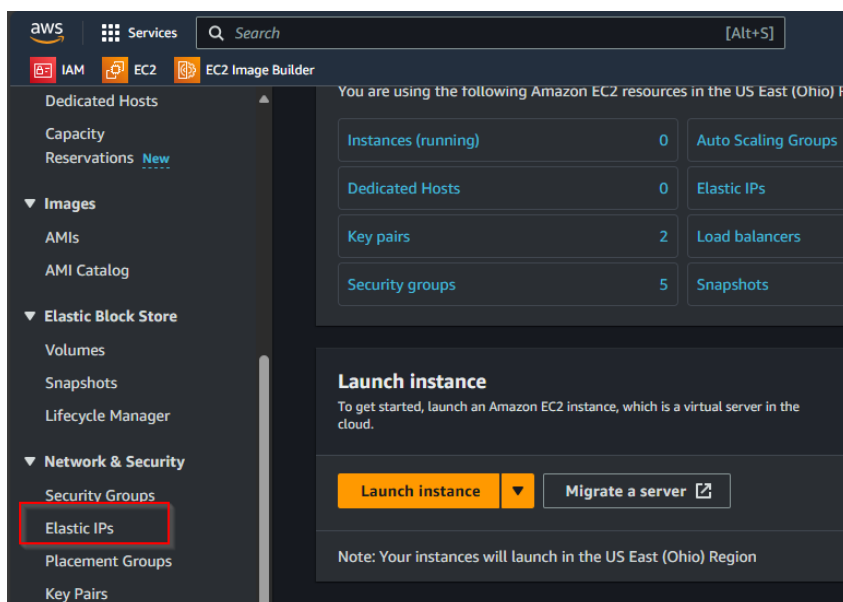
#Configurando um Elastic IP (EIP)

Um Elastic IP (EIP) é um endereço IPv4 público fornecido pela Amazon Web Services (AWS) que pode ser alocado para uso com instâncias do EC2. Ele é um recurso de IP fixo que permanece constante, diferentemente dos endereços IP públicos padrão, que podem mudar se a instância for interrompida ou reiniciada.

1. Entre no console AWS (Passo 1 do tutorial anterior);
2. No painel de navegação à esquerda, clique em *All services* no grupo *Compute* acesse *EC2*

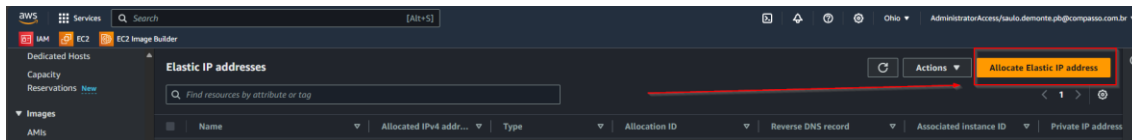


3. No painel esquerdo vá em *Elastic IPs*.

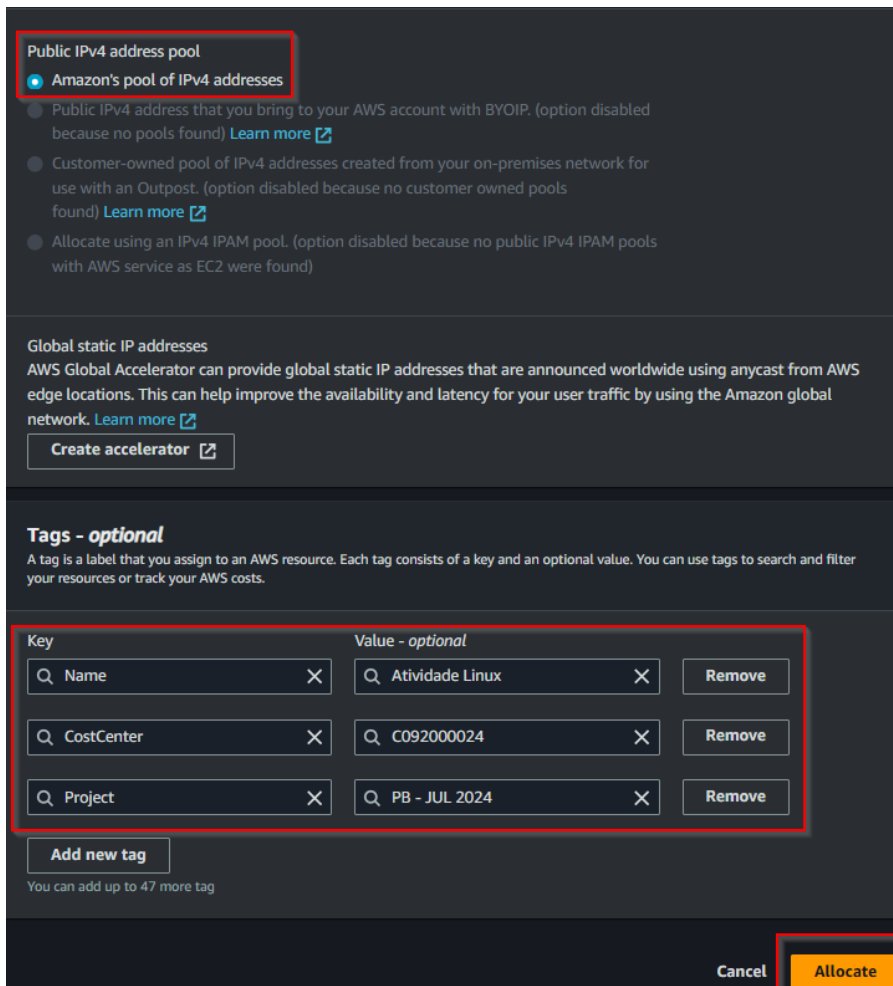


4. Clique em *Allocate Elastic IP address*.





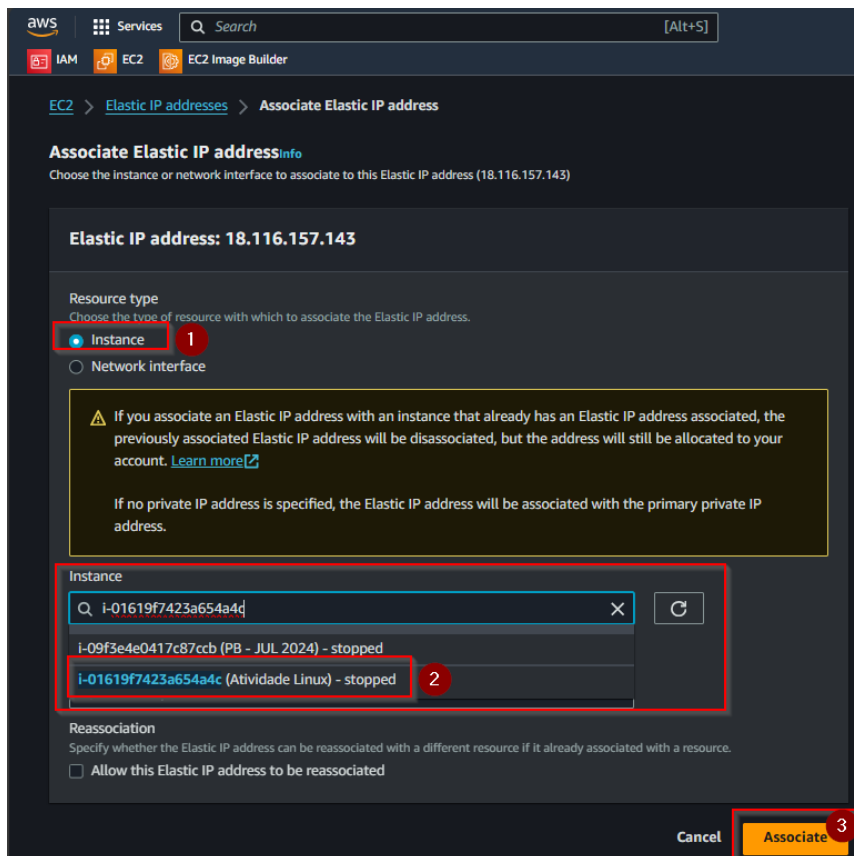
- Na tela seguinte, selecione a opção *Amazon's pool of IPv4 addresses*, adicione a tag do projeto para manter a organização e clique em *Allocate*.



- Anote o *Elastic IP address* que foi gerado e clique em *Associate this Elastic IP address*



- Na janela de associação, escolha a instância EC2 que você deseja associar e clique em *Associate*.



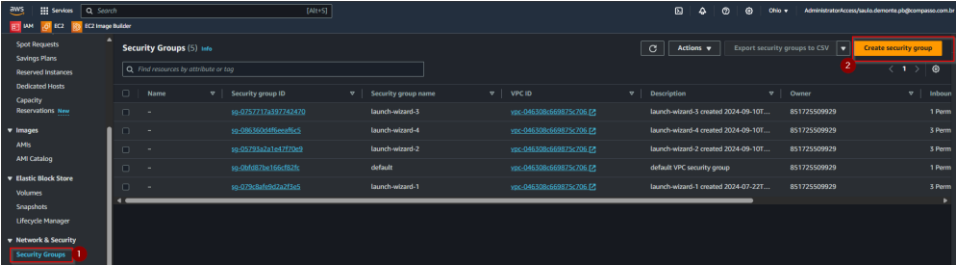
Agora o Elastic IP está vinculado à sua instância EC2.



## #Liberando as portas de comunicação

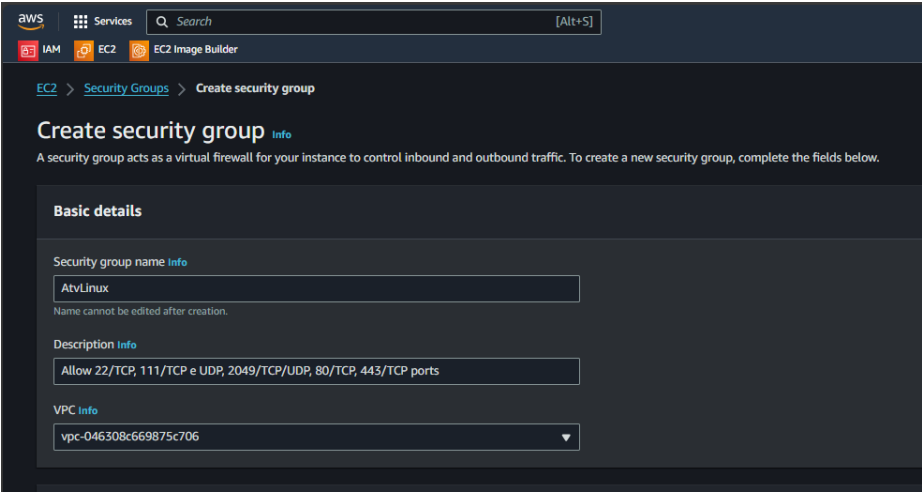
Um Security Group na Amazon Web Services (AWS) é uma configuração de firewall virtual que controla o tráfego de entrada e saída de recursos da AWS, como instâncias EC2. Ele define as regras que determinam quais tipos de tráfego podem alcançar ou sair de uma instância EC2, funcionando como uma camada adicional de segurança.

1. Acessar o Console AWS
2. No console do EC2, clique em *Security Groups* no painel de navegação à esquerda e em *Create Security Group*

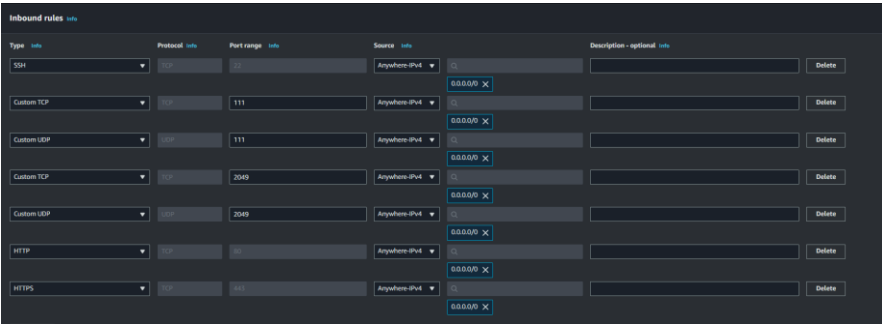


Preencha as seguintes informações:

- Name: Insira um nome descritivo para o Security Group (ex.: AtvLinux).
- Description: Adicione uma descrição que ajude a identificar a finalidade desse grupo (ex.: Grupo de segurança para servidor web com HTTP e SSH liberados).
- VPC: Escolha a VPC em que você deseja criar o Security Group (a maioria dos usuários terá apenas uma VPC padrão).



Adicionar as regras de entrada da nossa Instancia

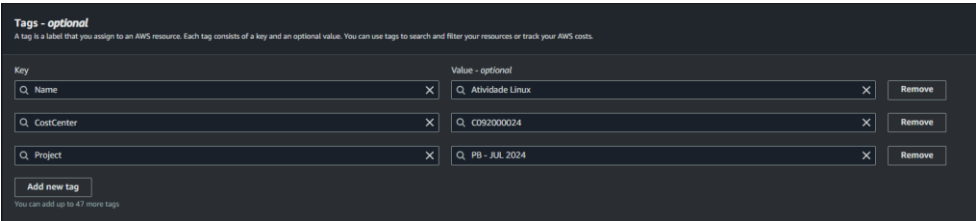


Resumo das Regras Adicionadas

Tipo	Protocolo	Faixa de portas	Origem
SSH	TCP	22	0.0.0.0/0
Custom TCP	TCP	111	0.0.0.0/0

Custom UDP	UDP	111	0.0.0.0/0
Custom TCP	TCP	2049	0.0.0.0/0
Custom UDP	UDP	2049	0.0.0.0/0
HTTP	TCP	80	0.0.0.0/0
HTTPS	TCP	443	0.0.0.0/0

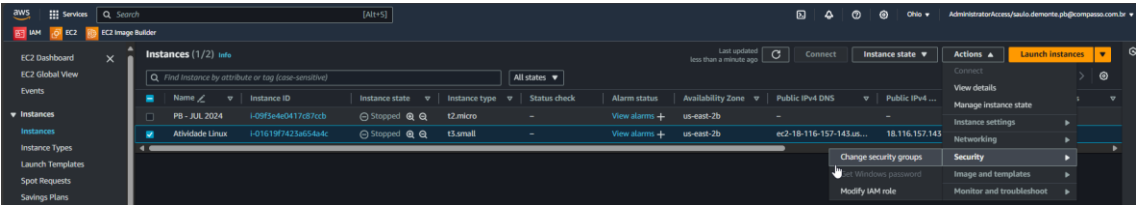
Uma boa prática é adicionar as TAGs para manter a organização.

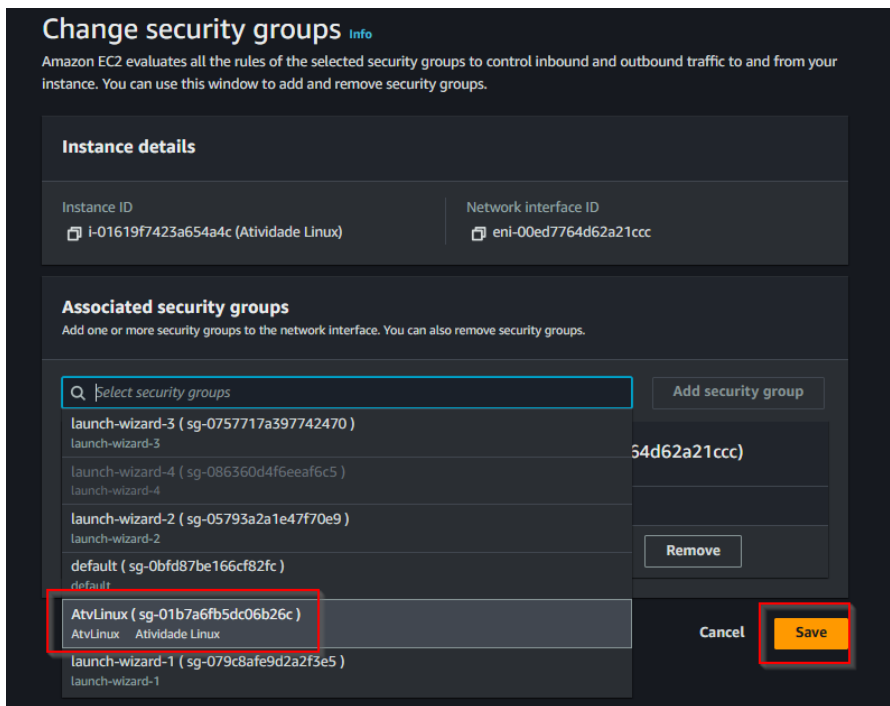


Mensagem de que nosso grupo de segurança foi criado com êxito



Não esqueça de associar o security group na instância que foi criada





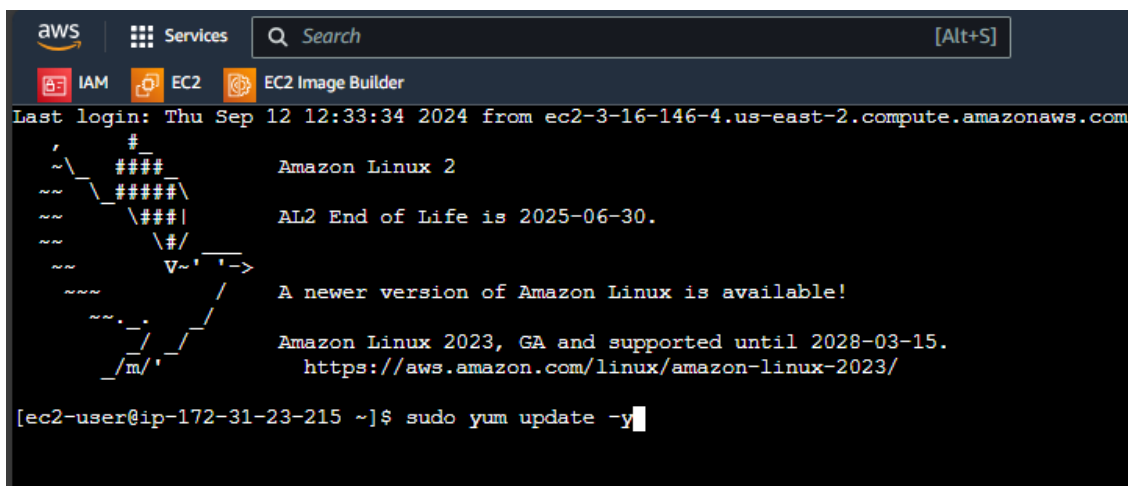
A instância já está pronta e com as configurações solicitadas para nosso projeto.

## #Instalado e Configurando o NFS

Atualizando os pacotes do servidor

Antes de instalar o NFS, é sempre bom garantir que seu sistema esteja com os pacotes atualizados

1. use o comando `sudo yum update -y`



2. Agora que os pacotes foram atualizados, execute o comando abaixo para instalar o serviço do NFS:

`sudo yum install nfs-utils -y`

Criado por: Saulo Demonte

Data de criação: 15/09/2024

Atualização: 15/09/2024

Caso já esteja instalado receberá esta mensagem

```
[ec2-user@ip-172-31-23-215 ~]$ sudo yum install nfs-utils -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Package 1:nfs-utils-1.3.0-0.54.amzn2.0.2.x86_64 already installed and latest version
Nothing to do
```

Explicação: O pacote `nfs-utils` contém os utilitários necessários para configurar e rodar o NFS no sistema. A opção `-y` confirma automaticamente a instalação.

3. Após a instalação, você precisa iniciar o serviço NFS e garantir que ele seja iniciado automaticamente quando o sistema for reinicializado:

```
[ec2-user@ip-172-31-23-215 ~]$ sudo systemctl start nfs-server
[ec2-user@ip-172-31-23-215 ~]$ sudo systemctl enable nfs-server
Created symlink from /etc/systemd/system/multi-user.target.wants/nfs-server.service to /usr/lib/systemd/system/nfs-server.service.
```

Explicação:

- `systemctl start nfs-server`: Inicia o serviço NFS.
- `systemctl enable nfs-server`: Configura o serviço NFS para iniciar automaticamente sempre que o sistema for ligado.

Você pode verificar se o NFS está rodando corretamente com o comando `sudo systemctl status nfs-server`. Isso deve te mostrar uma saída indicando que o serviço está ativo (rodando).

```
Created symlink from /etc/systemd/system/multi-user.target.wants/nfs-server.service to /usr/lib/systemd/system/nfs-server.service.
[ec2-user@ip-172-31-23-215 ~]$ sudo systemctl status nfs-server
● nfs-server.service - NFS server and services
   Loaded: loaded (/usr/lib/systemd/system/nfs-server.service; enabled; vendor preset: disabled)
   Active: active (exited) since Thu 2024-09-12 12:49:21 UTC; 1min 30s ago
     Main PID: 2852 (code=exited, status=0/SUCCESS)
    CGroup: /system.slice/nfs-server.service

Sep 12 12:49:20 ip-172-31-23-215.us-east-2.compute.internal systemd[1]: Starting NFS server and services...
Sep 12 12:49:21 ip-172-31-23-215.us-east-2.compute.internal systemd[1]: Started NFS server and services.
```

*Criando o diretório que será compartilhado*

Agora, precisamos criar o diretório no servidor que será compartilhado via NFS. Este diretório será acessível pelos clientes na rede.

1. Execute o seguinte comando para criar o diretório:

```
sudo mkdir -p /mnt/Saulo
```

```
sudo mkdir -p /mnt/saulo
```

Explicação:

- `mkdir`: Cria um diretório.
- `-p`: Garante que o diretório será criado, mesmo que os diretórios pais (como `/mnt`) ainda não existam.

Configurando permissões no diretório compartilhado

Defina permissões apropriadas para o diretório, permitindo que clientes NFS possam acessá-lo. Execute os seguintes comandos:

Criado por: Saulo Demonte

Data de criação: 15/09/2024

Atualização: 15/09/2024

2. Use os comandos abaixo

```
sudo chown nobody:nogroup /mnt/nfs_share
```

```
sudo chown nobody:nogroup /mnt/saulo
```

```
sudo chmod 777 /mnt/nfs_share
```

```
sudo chmod 777 /mnt/saulo
```

Explicação:

- chown: Altera o proprietário do diretório para o usuário nobody e o grupo nogroup, o que garante que o diretório seja acessível de forma genérica.
- chmod 777: garante que qualquer usuário (local ou remoto) tenha permissão para ler, escrever e executar dentro desse diretório.

Configurando permissões no arquivo /etc/exports

Certifique-se de que o arquivo /etc/exports no servidor NFS está configurado para permitir o acesso público.

Você precisa abrir o arquivo de configuração do NFS para adicionar essa linha.

Faça isso com um editor de texto, como o nano.

3. Abra o arquivo /etc/exports

```
sudo nano /etc/exports
```

```
sudo nano /etc/exports
```

Adicione a linha de configuração abaixo ao arquivo

```
/mnt/saulo *(rw, sync, no_subtree_check, no_root_squash)
```

```
/mnt/saulo *(rw, sync, no_subtree_check, no_root_squash)
```

Isso garante que qualquer máquina na internet poderá acessar o diretório compartilhado.

Explicação:

- /mnt/saulo: O diretório a ser compartilhado.
- \*: Permite acesso de qualquer IP.
- rw: Os clientes podem ler e escrever no diretório.
- sync: Gravações são feitas imediatamente no disco do servidor.
- no\_subtree\_check: Desativa verificações de subárvore para melhorar o desempenho.
- no\_root\_squash: Permite que clientes root tenham permissões de root no servidor, o que deve ser usado com cuidado.

**ATENÇÃO!** Isso cria um compartilhamento público e acessível, mas esse tipo de configuração pode ser vulnerável a ataques se a segurança da rede não for adequada.

4. Depois de adicionar a linha, salve o arquivo

Pressione Ctrl + X para sair.

Pressione Y para confirmar que deseja salvar as alterações.  
Pressione Enter para confirmar o nome do arquivo (ele já estará definido como /etc/exports)

Depois de salvar o arquivo /etc/exports, você precisa recarregar as configurações do NFS para que as novas regras entrem em vigor.

5. Execute o comando:

```
sudo exportfs -ra
```

```
sudo exportfs -ra
```

Esse comando faz com que o NFS reavalie todas as exportações definidas no arquivo /etc/exports e aplique as mudanças.

Explicação:

- sudo: Executa o comando com privilégios de superusuário, necessários para modificar as exportações NFS.
- exportfs: Utilitário que gerencia a exportação de sistemas de arquivos NFS no servidor.
- -r: Reexporta todos os sistemas de arquivos listados no arquivo /etc/exports. É utilizado para aplicar imediatamente quaisquer alterações feitas nesse arquivo sem precisar reiniciar o serviço NFS.
- -a: Exporta todos os sistemas de arquivos especificados no arquivo /etc/exports. Garante que todas as configurações de exportação sejam processadas.

Quando usar:

- Após editar /etc/exports: Sempre que fizer alterações no arquivo /etc/exports para adicionar, remover ou modificar diretórios compartilhados, use sudo exportfs -ra para aplicar essas mudanças sem a necessidade de reiniciar o serviço NFS.

Depois disso, o NFS estará configurado para compartilhar o diretório /mnt/saulo publicamente com as permissões definidas.

## Acessando o NFS

Use uma máquina Linux para acessar com a chave .pem e neste caso foi utilizado uma máquina virtual com o Debian instalado.

1. Instalar o Cliente NFS (Se Ainda Não Estiver Instalado)

Instale o pacote nfs-utils, que inclui o cliente NFS

```
sudo apt-get update
```

```
sudo apt-get update
```

```
sudo apt-get install -y nfs-common
```

```
sudo apt-get install -y nfs-common
```

Criado por: Saulo Demonte

Data de criação: 15/09/2024

Atualização: 15/09/2024



nfs-common: Este pacote inclui os utilitários necessários para se conectar a servidores NFS.

2. Crie um diretório onde o compartilhamento NFS será montado

```
sudo mkdir -p /mnt/nfs_shared
```

```
sudo mkdir -p /mnt/nfs_shared
```

/mnt/nfs\_shared: Este é o diretório local onde o compartilhamento NFS será montado na máquina cliente.

3. Configurar a Montagem Automática no /etc/fstab editando o Arquivo /etc/fstab:

Abra o arquivo /etc/fstab com um editor de texto, neste caso o nano foi utilizado

```
sudo nano /etc/fstab
```

```
sudo nano /etc/fstab
```

4. Adicione a seguinte linha ao final do arquivo /etc/fstab:

```
18.116.157.143:/mnt/saulo /mnt/nfs_shared nfs defaults 0 0
```

```
18.116.157.143:/mnt/saulo /mnt/nfs_shared nfs defaults 0 0
```

pressione CTRL + O para salvar e CTRL + X para sair

Detalhes da Configuração:

- 18.116.157.143:/mnt/saulo: O endereço IP do servidor NFS seguido do caminho do diretório compartilhado no servidor.
- /mnt/nfs\_shared: O caminho do diretório onde o compartilhamento NFS será montado na máquina cliente.
- nfs: Tipo de sistema de arquivos a ser montado.
- defaults: Opções de montagem padrão (rw, suid, dev, exec, auto, nouser, async).
- 0 0: Opções para backup e verificação de sistema de arquivos (geralmente deixadas como 0 para montagens NFS).

5. Para montar imediatamente o compartilhamento NFS sem reiniciar a máquina use o comando abaixo.

```
sudo mount -a
```

```
sudo mount -a
```

Se você quiser montar o compartilhamento NFS manualmente sem editar o arquivo /etc/fstab, use o seguinte comando:

```
sudo mount -t nfs 18.116.157.143:/mnt/saulo /mnt/nfs_shared
```

```
sudo mount -t nfs 18.116.157.143:/mnt/saulo /mnt/nfs_shared
```

Detalhes do comando

- -t nfs: Especifica que o tipo de sistema de arquivos é NFS.
- 18.116.157.143:/mnt/saulo: O endereço IP do servidor NFS e o caminho do diretório compartilhado.
- /mnt/nfs\_shared: O diretório local onde o compartilhamento NFS será montado.

**Nota:** Essa montagem é temporária e será removida na próxima reinicialização

6. Verifique se o compartilhamento NFS foi montado corretamente:

```
df -h | grep nfs
```

```
df -h | grep nfs
```

7. Testar a Montagem Automática (Opcional)

Reiniciar a Máquina Cliente usando o *sudo reboot*

```
sudo reboot
```

8. Verificar se a Montagem Persiste:

```
df -h | grep nfs
```

9. Para acessar o servidor NFS (neste caso, uma instância EC2) via SSH, use o seguinte comando:

```
ssh -i ~/AtvLinux.pem ec2-user@18.116.157.143
```

```
ssh -i ~/AtvLinux.pem ec2-user@18.116.157.143
```

Detalhes do comando

- -i ~/AtvLinux.pem: Especifica o caminho para a chave privada (AtvLinux.pem), necessária para a autenticação com a instância EC2.
- ec2-user: Nome do usuário padrão para instâncias Amazon Linux/Ubuntu na AWS.
- 18.116.157.143: O endereço IP público da instância EC2 (servidor NFS).

10. Verificar o Status do Serviço NFS

```
sudo systemctl status nfs-server
```

```
sudo systemctl status nfs-server
```

11. Reiniciar o Serviço NFS (Se Necessário)

```
sudo systemctl restart nfs-server
```

```
sudo systemctl restart nfs-server
```

## 12. Criar arquivo de teste

```
echo "Teste de compartilhamento NFS" > teste_nfs.txt
```

```
echo "Teste de compartilhamento NFS" > /mnt/nfs_shared/teste_nfs.txt
```

## Detalhes do comando

- **echo**: Este comando é usado para exibir uma linha de texto ou uma string no terminal.
- **"Teste de compartilhamento NFS"**: Esta é a string que será exibida ou, neste caso, escrita em um arquivo. Você pode alterar este texto para qualquer outra mensagem que desejar.
- **>**: Este é um operador de redirecionamento. Ele direciona a saída do comando **echo** para um arquivo. Se o arquivo especificado já existir, ele será sobrescrito. Se não existir, ele será criado.
- **teste\_nfs.txt**: Este é o nome do arquivo de destino onde a saída do comando **echo** será escrita. Neste exemplo, o comando cria (ou sobrescreve) um arquivo chamado **teste\_nfs.txt** e escreve a string **"Teste de compartilhamento NFS"** nele.

13. Verifica se o arquivo foi criado no diretório

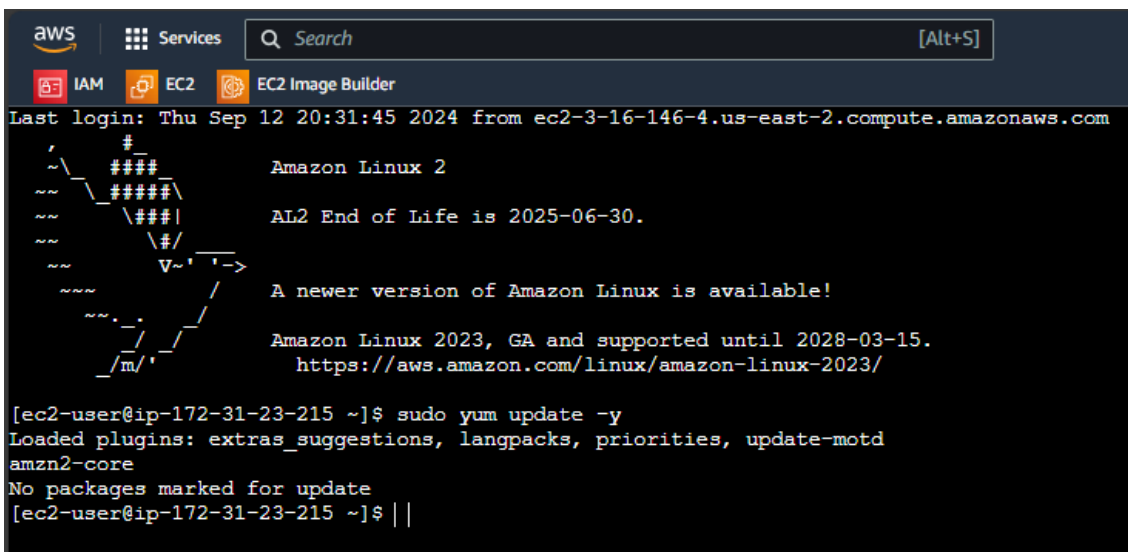
```
ls -l /mnt/nfs_shared/teste_nfs.txt
```

```
ls -l /mnt/nfs_shared/teste_nfs.txt
```

## #Instalação do Apache

1. Atualizar o servidor. Antes de instalar o Apache, é recomendável atualizar os pacotes do sistema:

```
sudo yum update -y
```



Criado por: Saulo Demonte

Data de criação: 15/09/2024

Atualização: 15/09/2024

2. Agora, instale o Apache no seu servidor.  
*sudo yum install httpd -y*

```
aws Services Search [Alt+S]
IAM EC2 EC2 Image Builder
No packages marked for update
[ec2-user@ip-172-31-23-215 ~]$ sudo yum install httpd -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Resolving Dependencies
--> Running transaction check
--> Package httpd.x86_64 0:2.4.62-1.amzn2.0.2 will be installed
--> Processing Dependency: httpdfilesystem = 2.4.62-1.amzn2.0.2 for package: httpd-2.4.62-1.amzn2.0.2.x86_64
--> Processing Dependency: httpd-tools = 2.4.62-1.amzn2.0.2 for package: httpd-2.4.62-1.amzn2.0.2.x86_64
--> Processing Dependency: /etc/mime.types for package: httpd-2.4.62-1.amzn2.0.2.x86_64
--> Processing Dependency: httpdfilesystem for package: httpd-2.4.62-1.amzn2.0.2.x86_64
--> Processing Dependency: mod_http2 for package: httpd-2.4.62-1.amzn2.0.2.x86_64
--> Processing Dependency: system-logos-httpd for package: httpd-2.4.62-1.amzn2.0.2.x86_64
--> Processing Dependency: libapr-1.so.0()(64bit) for package: httpd-2.4.62-1.amzn2.0.2.x86_64
--> Processing Dependency: libaprutil-1.so.0()(64bit) for package: httpd-2.4.62-1.amzn2.0.2.x86_64
--> Running transaction check
--> Package apr.x86_64 0:1.7.2-1.amzn2 will be installed
--> Package apr-util.x86_64 0:1.6.3-1.amzn2.0.1 will be installed
--> Processing Dependency: apr-util-bdb(x86-64) = 1.6.3-1.amzn2.0.1 for package: apr-util-1.6.3-1.amzn2.0.1.x86_64
```

3. Depois da instalação, inicie o serviço do Apache e configure-o para iniciar automaticamente no boot.  
*sudo systemctl start httpd*  
*sudo systemctl enable httpd*

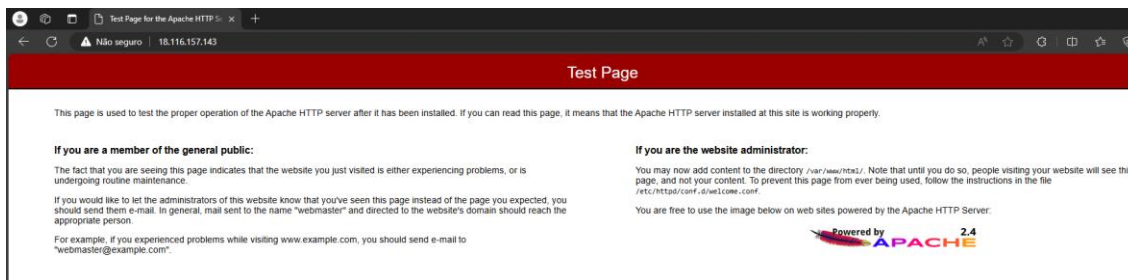
```
aws Services Search [Alt+S]
IAM EC2 EC2 Image Builder
[ec2-user@ip-172-31-23-215 ~]$ sudo systemctl start httpd
[ec2-user@ip-172-31-23-215 ~]$ sudo systemctl enable httpd
Created symlink from /etc/systemd/system/multi-user.target.wants/httpd.service to /usr/lib/systemd/system/httpd.service.
[ec2-user@ip-172-31-23-215 ~]$
```

4. Certifique-se de que o Apache está rodando corretamente:  
*sudo systemctl status httpd*

```
aws Services Search [Alt+S]
IAM EC2 EC2 Image Builder
[ec2-user@ip-172-31-23-215 ~]$ sudo systemctl start httpd
[ec2-user@ip-172-31-23-215 ~]$ sudo systemctl enable httpd
Created symlink from /etc/systemd/system/multi-user.target.wants/httpd.service to /usr/lib/systemd/system/httpd.service.
[ec2-user@ip-172-31-23-215 ~]$ sudo systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; vendor preset: disabled)
   Active: active (running) since Fri 2024-09-13 12:36:43 UTC; 48s ago
     Docs: man:httpd.service(8)
  Main PID: 2947 (httpd)
    Status: "Total requests: 0; Idle/Busy workers 100/0; Requests/sec: 0; Bytes served/sec: 0 B/sec"
    CGroup: /system.slice/httpd.service
            └─2947 /usr/sbin/httpd -DFOREGROUND
              └─2950 /usr/sbin/httpd -DFOREGROUND
                └─2952 /usr/sbin/httpd -DFOREGROUND
                  └─2953 /usr/sbin/httpd -DFOREGROUND
                    └─2954 /usr/sbin/httpd -DFOREGROUND
                      └─2955 /usr/sbin/httpd -DFOREGROUND

Sep 13 12:36:43 ip-172-31-23-215.us-east-2.compute.internal systemd[1]: Starting The Apache HTTP Server...
Sep 13 12:36:43 ip-172-31-23-215.us-east-2.compute.internal systemd[1]: Started The Apache HTTP Server.
[ec2-user@ip-172-31-23-215 ~]$
```

5. Para verificar se o Apache está funcionando corretamente, abra um navegador e digite o endereço IP do servidor. Se o Apache estiver configurado corretamente, você verá a página padrão do Apache.  
<http://18.116.157.143>



## #Script de monitoramento

**Objetivo:** Criar um script de monitoramento do servidor Apache onde no script deve conter:

- Data / Hora;
- Nome do Serviço;
- Status do serviço;
- Mensagem personalizada de online ou offline;

O script deve gerar 2 arquivos de saída: 1 para o serviço online e 1 para o serviço offline;

Preparar a execução automatizada do script a cada 5 minutos.

1. Acesse a máquina onde você deseja executar o script (no caso, [ec2-user@18.116.157.143](mailto:ec2-user@18.116.157.143)).
2. Navegue até o diretório onde deseja criar o script, por exemplo, /home/ec2-user
3. Use um editor de texto para criar o script CheckApache.sh:

```
sudo nano /home/ec2-user/CheckApache.sh
```

4. Adicionar o Conteúdo do Script

```
aws Services Search [Alt+S]
IAM EC2 EC2 Image Builder
GNU nano 2.9.8 /home

#!/bin/bash
# Nome do serviço que será verificado
SERVICO="httpd"

# Captura a data e hora atual
DATA_HORA=$(date "+%Y-%m-%d %H:%M:%S")

# Verifica o status do serviço
if systemctl is-active --quiet $SERVICO; then
    STATUS="ONLINE"
    MENSAGEM="O serviço $SERVICO está funcionando corretamente."
    echo "$DATA_HORA - $SERVICO - $STATUS - $MENSAGEM" >> /mnt/saulo/servico_online.log
else
    STATUS="OFFLINE"
    MENSAGEM="O serviço $SERVICO está parado ou enfrentando problemas."
    echo "$DATA_HORA - $SERVICO - $STATUS - $MENSAGEM" >> /mnt/saulo/servico_offline.log
fi

[ec2-user@ip-172-31-23-215 ~]$ nano /home/ec2-user/CheckApache.sh
```

Script Detalhado:

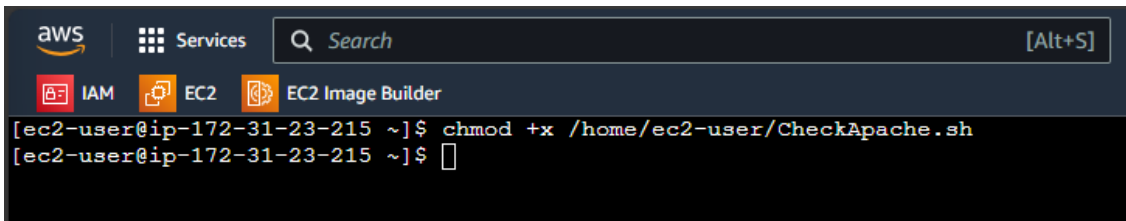
Criado por: Saulo Demonte  
Data de criação: 15/09/2024  
Atualização: 15/09/2024

- `#!/bin/bash`: Esta linha define o interpretador do script, neste caso, `/bin/bash`. É importante para que o sistema saiba como executar o script.
  - `SERVICO="httpd"`: Esta linha define o nome do serviço que será verificado. Aqui, ele está configurado para verificar o serviço `httpd`, que é o nome do Apache em distribuições como CentOS/Red Hat e pode ser ajustada para qualquer serviço que deseja monitorar
  - `DATA_HORA=$(date "+%Y-%m-%d %H:%M:%S")`: Captura a data e a hora atual no formato "AAAA-MM-DD HH:MM usando o comando `date`. A data e a hora são armazenadas na variável `DATA_HORA` para serem usadas no log.
  - `if systemctl is-active --quiet $SERVICO; then`: Usa o comando `systemctl is-active` para verificar se o serviço está ativo.
  - `--quiet` suprime a saída normal, retornando apenas o status. Se o serviço estiver ativo, a condição é verdadeira e o bloco de código seguinte (then) será executado.
  - `STATUS="ONLINE"`  
`MENSAGEM="O serviço $SERVICO está funcionando corretamente."`
  - `echo "$DATA_HORA - $SERVICO - $STATUS - $MENSAGEM" >> /mnt/saulo/servico_online.log`: Quando o serviço está ativo define `STATUS` como "ONLINE".  
Define uma mensagem indicando que o serviço está funcionando corretamente.  
Usa o comando `echo` para gravar a data, hora, nome do serviço, status e mensagem no arquivo `/mnt/saulo/servico_online.log`.  
`>>` é usado para anexar a mensagem ao arquivo, para que os registros não sejam sobrescritos.
  - `else`  
`STATUS="OFFLINE"`  
`MENSAGEM="O serviço $SERVICO está parado ou enfrentando problemas."`  
`echo "$DATA_HORA - $SERVICO - $STATUS - $MENSAGEM" >> /mnt/saulo/servico_offline.log`  
`fi`  
Se o serviço **não** estiver ativo: define `STATUS` como "OFFLINE".  
Define uma mensagem indicando que o serviço está parado ou enfrentando problemas.  
Usa o comando `echo` para gravar a data, hora, nome do serviço, status e mensagem no arquivo `/mnt/saulo/servico_offline.log`.  
Assim como antes, `>>` é usado para anexar a mensagem ao arquivo.
5. A imagem mostra o comando sendo executado com sucesso no terminal de uma instância AWS EC2.

```
chmod +x /home/ec2-user/CheckApache.sh
```

Explicação do Comando na Imagem

- `chmod +x`: Este comando altera as permissões do arquivo para torná-lo executável.
- `/home/ec2-user/CheckApache.sh`: Este é o caminho para o script `CheckApache.sh` criado.



6. Execute o script manualmente para verificar se ele funciona:  
`/home/ec2-user/CheckApache.sh`

```
/home/ec2-user/CheckApache.sh
```

7. Verifique se os logs foram gerados corretamente:  
`cat /mnt/saulo/servico_online.log`

```
cat /mnt/saulo/servico_online.log
```

```
cat /mnt/saulo/servico_offline.log
```

```
cat /mnt/saulo/servico_offline.log
```

## #Automatizando o Script

O `crontab` é uma ferramenta do sistema Linux usada para agendar a execução automática de tarefas em horários específicos. Ele é parte do serviço `cron`, que é um "agendador de tarefas" que executa comandos ou scripts automaticamente em segundo plano, de acordo com um cronograma definido.

1. Para editar as tarefas do `cron` para o usuário atual (neste caso, `ec2-user`), use o seguinte comando:

```
crontab -e
```

```
crontab -e
```

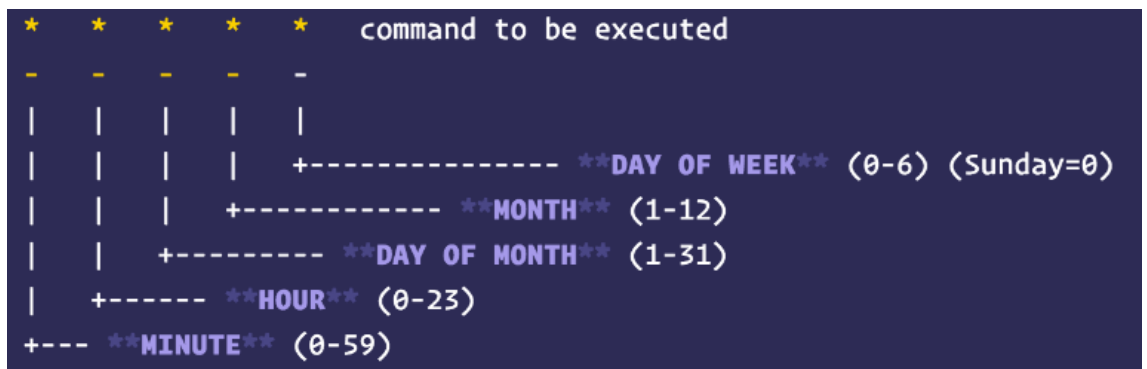
2. Adicione a seguinte linha ao `crontab` para executar o script a cada 5 minutos e redirecionar a saída para um log em `/tmp`:

```
*/5 * * * * /bin/bash /home/ec2-user/CheckApache.sh >>
/tmp/CheckApache.log 2>&1
```

```
*/5 * * * * /bin/bash /home/ec2-user/CheckApache.sh >> /tmp/CheckApache.log 2>&1
```

Explicação da Sintaxe de Agendamento (\*/\* \* \* \* \*):

- \*/5: Este campo representa os minutos. Neste caso, \*/5 significa "a cada 5 minutos". O asterisco (\*) indica todos os valores possíveis e o /5 é um intervalo, significando que a tarefa será executada a cada 5 minutos (0, 5, 10, 15, etc.).
- (Segundo Campo): Este campo representa as horas. Um único asterisco (\*) significa "todas as horas" (0-23). Assim, a tarefa será executada a cada 5 minutos de cada hora do dia.
- (Terceiro Campo): Este campo representa os dias do mês. Um único asterisco (\*) indica "todos os dias do mês" (1-31).
- (Quarto Campo): Este campo representa os meses. Um asterisco (\*) significa "todos os meses" (1-12).
- (Quinto Campo): Este campo representa os dias da semana. Um asterisco (\*) significa "todos os dias da semana" (0-6, onde 0 é domingo).



Explicação do restante do comando

- /bin/bash: Especifica o interpretador Bash para executar o script.
- /home/ec2-user/CheckApache.sh: Caminho completo para o script a ser executado.
- >> /tmp/CheckApache.log 2>&1: Redireciona a saída e erros do script para /tmp/CheckApache.log.

Caso queira checar o log de erro do script, use o comando:  
`cat /tmp/CheckApache.log`

### 3. Comandos úteis do crontab



- `crontab -l`: lista as tarefas
  - `crontab -r`: apaga todas as entradas do crontab sem confirmação
4. Comandos para verificar os logs
- `Ls -l /mnt/saulo`: verifica se os logs estão na pasta do nfs
- `cat /mnt/saulo/servico_online.log`: verifica o conteúdo dos logs de serviço online
- `cat /mnt/saulo/servico_offline.log`: verifica o conteúdo dos logs de serviço offline

## #Exibição de logs na página do Apache

### Tutorial: Exibindo Logs do Apache com Indicadores Visuais em PHP

#### Passo 1: Criar o Arquivo PHP para Exibir os Logs

Acessar o Servidor via SSH:

Conecte-se ao servidor onde o Apache e os logs estão configurados:

```
ssh -i ~/AtvLinux.pem ec2-user@18.116.157.143
```

Navegar para o Diretório da Web:

Vá para o diretório raiz da web do Apache (/var/www/html):

```
cd /var/www/html
```

Criar o Arquivo `exibir_logs.php`

Crie um arquivo PHP chamado `exibir_logs.php`

```
sudo nano exibir_logs.php
```

Adicionar o Código PHP

```

<?php
// Caminhos para os arquivos de log
// Defina os caminhos para os arquivos de log que você deseja exibir.
$logFileOnline = '/mnt/saulo/servico_online.log';
$logFileOffline = '/mnt/saulo/servico_offline.log';

// Função para exibir as 20 entradas mais recentes de um log
function displayLog($logFile, $logTitle) {
    // Verifica se o arquivo de log existe
    if (file_exists($logFile)) {
        // Lê o arquivo de log em um array, linha por linha
        $logContent = file($logFile);

        // Inverte a ordem das linhas (entradas mais recentes no início)
        $logContent = array_reverse($logContent);

        // Pega as 20 primeiras linhas (mais recentes)
        $recentEntries = array_slice($logContent, 0, 20);

        // Exibe o título do log
        echo "<h2>$logTitle</h2>";
        echo "<pre>";

        // Percorre cada entrada recente para exibir
        foreach ($recentEntries as $entry) {
            // Se a linha contém "ONLINE", adicione um ponto verde piscando
            if (strpos($entry, 'ONLINE') !== false) {
                echo "<span class='status online'></span> " . htmlspecialchars($entry);
            }
            // Se a linha contém "OFFLINE", adicione um ponto vermelho piscando
            elseif (strpos($entry, 'OFFLINE') !== false) {
                echo "<span class='status offline'></span> " . htmlspecialchars($entry);
            } else {
                // Exibe a entrada normalmente se não for "ONLINE" ou "OFFLINE"
                echo htmlspecialchars($entry);
            }
        }
    }
}

```

```

    }
}

echo "</pre>";
} else {
    // Exibe uma mensagem se o arquivo de log não for encontrado
    echo "<h2>$logTitle</h2>";
    echo "O arquivo de log não foi encontrado.";
}

```

Exibe os logs do serviço online e offline

```

playLog($logFileOnline, 'Logs do Serviço Online');
playLog($logFileOffline, 'Logs do Serviço Offline');

```

<!-- CSS para os indicadores -->

[Copiar código](#)

```

<style>
/* Define estilos para os pontos indicadores */
.status {
    display: inline-block; /* Faz com que o elemento seja exibido como um bloco inline */
    width: 10px; /* Define a largura do ponto */
    height: 10px; /* Define a altura do ponto */
    border-radius: 50%; /* Torna o elemento redondo */
    margin-right: 5px; /* Adiciona um espaçamento à direita */
}

/* Estilo para o ponto verde piscante */
.online {
    background-color: green; /* Define a cor verde */
    animation: blink 1s infinite; /* Aplica a animação de piscar */
}

/* Estilo para o ponto vermelho piscante */
.offline {
    background-color: red; /* Define a cor vermelha */
    animation: blink 1s infinite; /* Aplica a animação de piscar */
}

/* Define a animação de piscar */
@keyframes blink {
    0% { opacity: 1; }
    50% { opacity: 0; }
    100% { opacity: 1; }
}
</style>

```

## Passo 2: Testar a Página PHP




Reinicie o Apache para aplicar as alterações

```
sudo systemctl restart httpd
```

Acesse a página criada no navegador para visualizar os logs

[http://18.116.157.143/exibir\\_logs.php](http://18.116.157.143/exibir_logs.php)

```
http://18.116.157.143/exibir_logs.php
```

   Não seguro | 18.116.157.143/exibir\_logs.php

### Logs do Serviço Online

- 2024-09-15 12:20:01 - httpd - ONLINE - O serviço httpd está funcionando corretamente.
- 2024-09-15 12:15:01 - httpd - ONLINE - O serviço httpd está funcionando corretamente.
- 2024-09-15 12:10:01 - httpd - ONLINE - O serviço httpd está funcionando corretamente.
- 2024-09-15 12:05:01 - httpd - ONLINE - O serviço httpd está funcionando corretamente.
- 2024-09-15 12:00:01 - httpd - ONLINE - O serviço httpd está funcionando corretamente.
- 2024-09-15 11:55:01 - httpd - ONLINE - O serviço httpd está funcionando corretamente.
- 2024-09-15 11:50:01 - httpd - ONLINE - O serviço httpd está funcionando corretamente.
- 2024-09-15 11:45:01 - httpd - ONLINE - O serviço httpd está funcionando corretamente.
- 2024-09-15 11:40:01 - httpd - ONLINE - O serviço httpd está funcionando corretamente.
- 2024-09-15 11:35:01 - httpd - ONLINE - O serviço httpd está funcionando corretamente.
- 2024-09-15 11:30:01 - httpd - ONLINE - O serviço httpd está funcionando corretamente.
- 2024-09-15 11:25:01 - httpd - ONLINE - O serviço httpd está funcionando corretamente.
- 2024-09-15 11:20:01 - httpd - ONLINE - O serviço httpd está funcionando corretamente.
- 2024-09-15 11:15:01 - httpd - ONLINE - O serviço httpd está funcionando corretamente.
- 2024-09-15 11:10:01 - httpd - ONLINE - O serviço httpd está funcionando corretamente.
- 2024-09-15 11:05:01 - httpd - ONLINE - O serviço httpd está funcionando corretamente.
- 2024-09-15 11:00:01 - httpd - ONLINE - O serviço httpd está funcionando corretamente.
- 2024-09-15 10:55:01 - httpd - ONLINE - O serviço httpd está funcionando corretamente.
- 2024-09-15 10:50:01 - httpd - ONLINE - O serviço httpd está funcionando corretamente.
- 2024-09-15 10:45:01 - httpd - ONLINE - O serviço httpd está funcionando corretamente.

### Logs do Serviço Offline

- 2024-09-14 13:15:01 - httpd - OFFLINE - O serviço httpd está parado ou enfrentando problemas.
- 2024-09-14 13:10:02 - httpd - OFFLINE - O serviço httpd está parado ou enfrentando problemas.
- 2024-09-14 13:05:01 - httpd - OFFLINE - O serviço httpd está parado ou enfrentando problemas.

