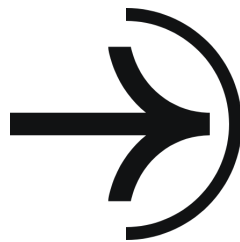


AWS & DEVSECOPS



Atividade referente aos Conhecimento em AWS e
Linux

Claudio Gabriel Kosooski

Frederico Westphalen, 03/09/2023

Tópicos Abordados

1.0 - Requisitos AWS - Pg 3 a 6

1.1 - Gerar uma chave pública para acesso ao ambiente - Pg 3

1.2 - Criar 1 instância EC2 com o sistema operacional Amazon Linux 2 (Família t3.small, 16GB SSD) - Pg 3 a 4

1.3 - Gerar 1 elastic IP e anexar à instância EC2 - Pg 4 A 5

1.4 - Liberar as portas de comunicação para acesso público: (22/TCP, 111/TCP e UDP, 2049/TCP/UDP, 80/TCP, 443/TCP) - Pg 6

2.0 - Atividade referente aos conhecimentos no Linux - Pg 7 a 22

2.1 - Instalação do Oracle Linux - Pg 7 a 16

2.2 - Configurando o NFS - Pg 16 a 19

2.3 - Criação do diretório - Pg 19

2.4- Instalação do Servidor Apache - Pg 20

2.5 - Criação dos Scripts de Validação do Status- Pg 21 a 22

1.0 - Requisitos AWS:

1.1 - Gerar uma chave pública para acesso ao ambiente;

A criação da chave pública ocorre no momento da criação da instância EC2, onde, o ideal é criar uma chave RSA para acesso a instância via SSH denominada `minha_chave_pb.pem`.

▼ 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.

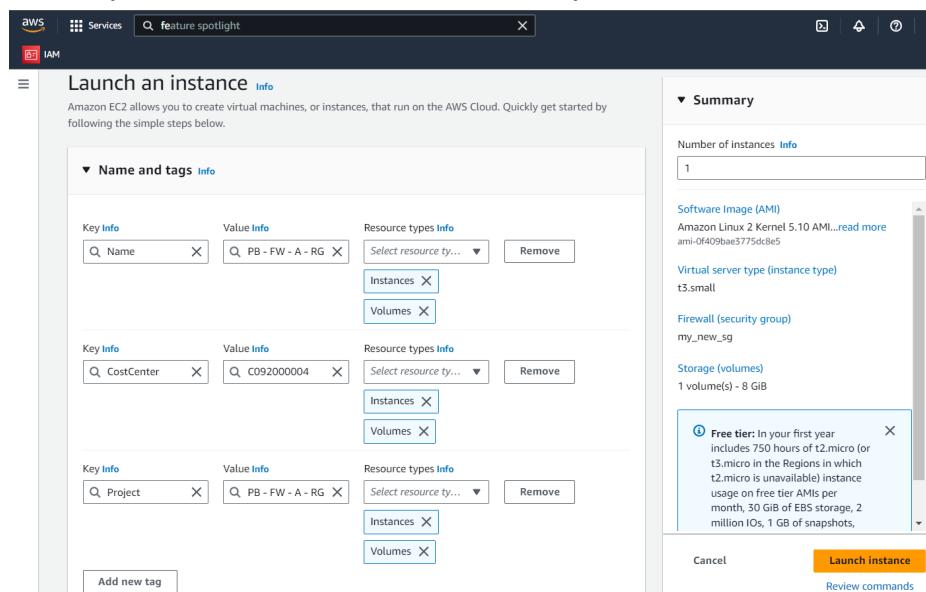
Key pair name - *required*

minha_chave_pb

↻ Create new key pair

1.2 - Criar 1 instância EC2 com o sistema operacional Amazon Linux 2 (Família t3.small, 16 GB SSD);

Neste segundo passo juntamente com o primeiro, foi criada a instância conforme as especificações passadas juntamente com as tags disponibilizadas no grupo do Teams para ocorrer o funcionamento esperado.



- Não se esqueça, que antes de finalizar, deve-se aumentar o armazenamento da instância para 16gb com SSD

▼
Configure storage
Info

1x
16
GiB
gp2
Root volume (Not encrypted)

Amazon Linux
macOS
Ubuntu
Windows
Red Hat
SUSE Li

Browse more AMIs
Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Amazon Linux 2 AMI (HVM) - Kernel 5.10, SSD Volume Type
ami-0f409bae3775dc8e5 (64-bit (x86)) / ami-0f0f7b386be96ec2d (64-bit (Arm))
Virtualization: hvm ENA enabled: true Root device type: ebs

Free tier eligible

Description

Amazon Linux 2 Kernel 5.10 AMI 2.0.20230822.0 x86_64 HVM gp2

Architecture

AMI ID

64-bit (x86)
ami-0f409bae3775dc8e5
Verified provider

▼
Instance type
Info

Instance type

t3.small
Family: t3 2 vCPU 2 GiB Memory Current generation: true
On-Demand SUSE base pricing: 0.0518 USD per Hour
On-Demand Linux base pricing: 0.0208 USD per Hour
On-Demand RHEL base pricing: 0.0808 USD per Hour
On-Demand Windows base pricing: 0.0392 USD per Hour

All generations

Compare instance types

- Pronto, agora é só iniciá-la e aguardar mudar o status para 'Running'

☐ PB - FW - A - RG - SB - HA
 [i-0689e3f4b537371fd](#)
Pending
t3.small

1.3 - Gerar 1 elastic IP e anexar à instância EC2;

No menu Elastic IP, em Allocate Elastic IP address, é possível alocar um determinado IP4 ou IP6 público e adicionar à instância criada no passo anterior.

Elastic IP addresses (1/2)						Actions
Filter Elastic IP addresses						Allocate Elastic IP address
<input type="checkbox"/>	Name	Allocated IPv4 address	Type	Allocation ID	Reverse	
<input checked="" type="checkbox"/>	-	44.218.189.37	Public IP	eipalloc-0a75955b18ef68e39	-	View details Release Elastic IP address Associate Elastic IP address Disassociate Elastic IP address
<input type="checkbox"/>	-	52.207.24.34	Public IP	eipalloc-004a045c4b317f3a7	-	

Elastic IP address: 44.218.189.37

Resource type

Choose the type of resource with which to associate the Elastic IP address.

☒ Instance
 ☐ Network interface

If you associate an Elastic IP address with an instance that already has an Elastic IP address associated, the previously associated Elastic IP address will be disassociated, but the address will still be allocated to your account. [Learn more](#)

If no private IP address is specified, the Elastic IP address will be associated with the primary private IP address.

Instance

Private IP address

The private IP address with which to associate the Elastic IP address.

Reassociation

Specify whether the Elastic IP address can be reassociated with a different resource if it already associated with a resource.

☐ Allow this Elastic IP address to be reassociated

Cancel

Associate

Após seguir o passo anterior, nesta tela acima eu faço a Associação à instância. O ideal é sempre criar uma subnet pública e associar na VPC (pode ser criada uma nova ou utilizada a padrão) uma tabela de rotas que aponte, além da rede local, para um Internet Gateway;

▼ Network settings

Info

VPC - required

Info

10.0.1.0/24

Subnet

Info

my_subnet_publica

VPC: vpc-0d1477ab9345744c2

Owner: 335505840552

Availability Zone: us-east-1a

IP addresses available: 250

CIDR: 10.0.1.0/24

[Create new subnet](#)

Auto-assign public IP

Info

Firewall (security groups)

Info

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

☐ Create security group
 ☒ Select existing security group

Common security groups

Info

VPC: vpc-0d1477ab9345744c2

[Compare security group rules](#)

Security groups that you add or remove here will be added to or removed from all your network interfaces.

1-4 - Liberar as portas de comunicação para acesso público: (22/TCP, 111/TCP e UDP, 2049/TCP/UDP, 80/TCP, 443/TCP).

Após criada a instância ec2, no menu security, encontramos o security group denominado 'my_new_sg', vinculado a nova VPC e Subnet Pública, criados durante a criação da instância. Em inbound rules (Regras de entrada, ou seja, acesso), clicamos em edit inbound rules e adicionamos todas as regras requeridas conforme imagem e tabela abaixo:

Editar regras de entrada [Informações](#)

As regras de entrada controlam o tráfego de entrada que tem permissão para acessar a instância.

ID da regra do grupo de segurança	Tipo Informações	Protocolo Informações	Intervalo de portas Informações	Origem Informações	Descrição - opcional Informações
sgr-0ac8da3cd799d4e8d	SSH	TCP	22	Personaliza... Q 0.0.0.0/0 X	Excluir
sgr-069d4c9a774f88372	TCP personalizado	TCP	111	Personaliza... Q 0.0.0.0/0 X	Excluir
sgr-0b83895500bc6c705	UDP personalizado	UDP	111	Personaliza... Q 0.0.0.0/0 X	Excluir
sgr-0d54ea394b662ac28	UDP personalizado	UDP	2049	Personaliza... Q 0.0.0.0/0 X	Excluir
sgr-0f137f5f1ee1ab5b5	HTTPS	TCP	443	Personaliza... Q 0.0.0.0/0 X	Excluir
sgr-0d17c46399dc39c62	NFS	TCP	2049	Personaliza... Q 0.0.0.0/0 X	Excluir

Tipo	Protocolo	Portas	Origem
UDP personalizado	UDP	111	0.0.0.0/0
TCP personalizado	TCP	111	0.0.0.0/0
UDP personalizado	UDP	2049	0.0.0.0/0
SSH	TCP	22	0.0.0.0/0
HTTP	TCP	80	0.0.0.0/0
HTTPS	TCP	443	0.0.0.0/0
NFS	TCP	2049	0.0.0.0/0

Sempre ao acabar, revise as configurações, e se tudo estiver ok, clique em 'Salvar Regras' para assim, iniciar a instância com as configurações já definidas de segurança.

2.0- Atividade referente aos conhecimentos no Linux

Requisitos da entrega:

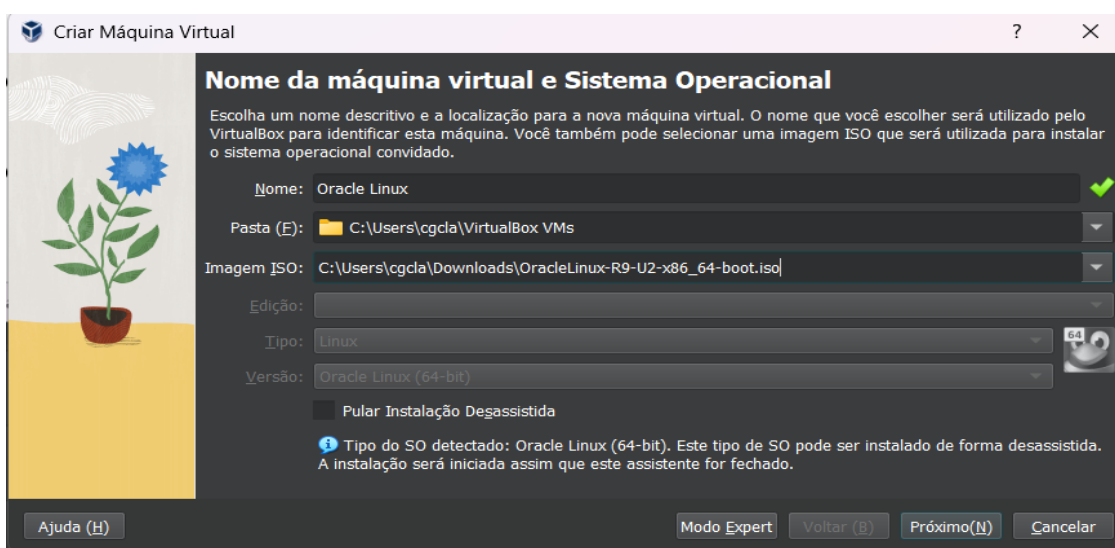
- Configurar o NFS entregue;
- Criar um diretório dentro do filesystem do NFS com seu nome;
- Subir um apache no servidor - o apache deve estar online e rodando;
- Criar um script que valide se o serviço está online e envie o resultado da validação para o seu diretório no nfs;
- O script deve conter - Data HORA + nome do serviço + Status + 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.
- Fazer o versionamento da atividade;
- Fazer a documentação explicando o processo de instalação do Linux.

2.1 - Instalação do Oracle Linux

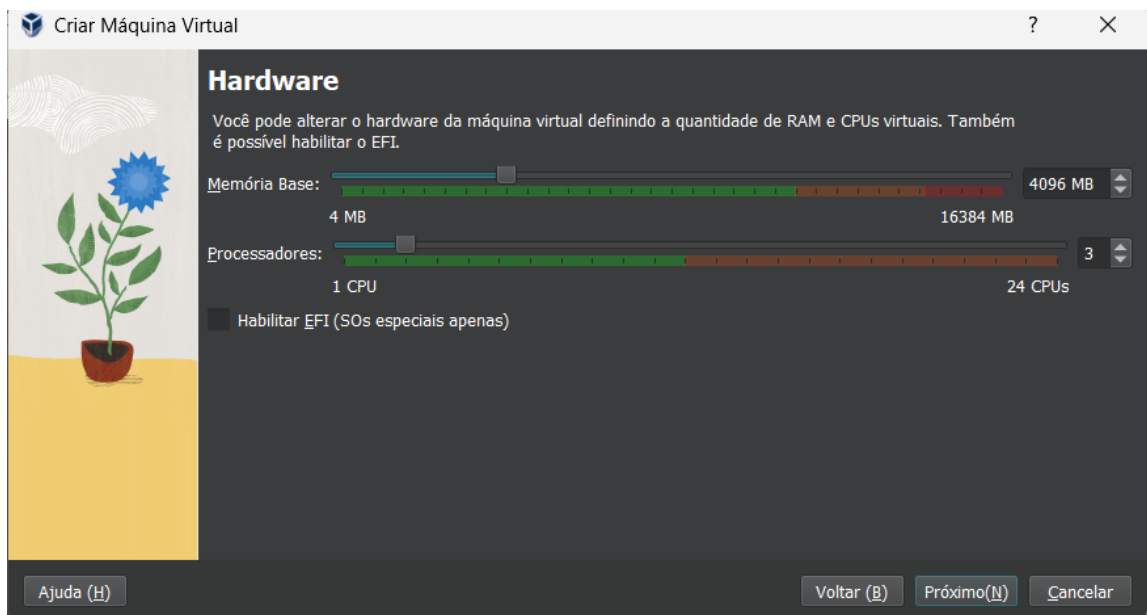
- Primeiramente, deve-se fazer acesso ao site do Oracle LINUX (no seguinte link <https://yum.oracle.com/oracle-linux-downloads.html>), onde poderá ser feito o Download da ISO conforme desejar
- Depois, deve ser feita a instalação do VirtualBox, para a continuidade da atividade, no link '<https://www.virtualbox.org/wiki/Downloads>'
- Após fazer o download da ISO e do VirtualBox, deve-se entrar na VB e adicionar a ISO. Para isso, tem que ir em ferramentas, Novo e assim irá abrir o menu para fazer as configurações iniciais da VM.



- Como dito antes, ao clicar em novo, será aberta a janela conforme imagem a seguir. Agora, você pode escolher um nome para a sua VM, escolher o diretório de instalação, a ISO que foi baixada e iniciar a instalação no VirtualBox



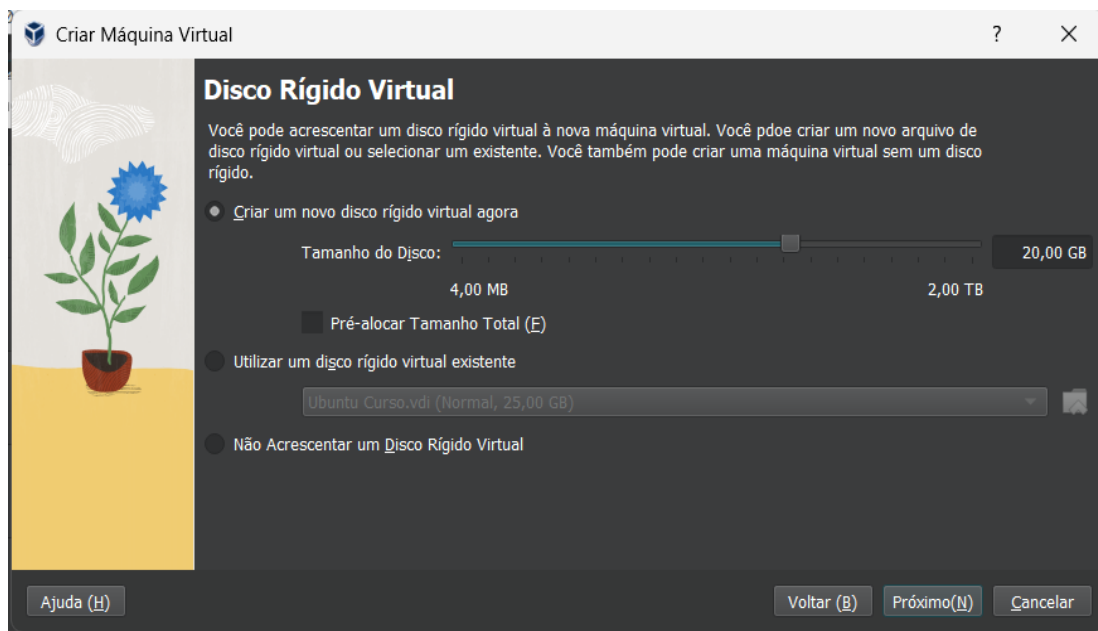
- Logo em seguida, você consegue determinar o quanto de memória RAM e de processamento você deseja alocar, nesse caso aloquei 4GB e 3 núcleos do CPU para processamento, o que é uma ótima configuração para uma máquina com Linux sem modo gráfico



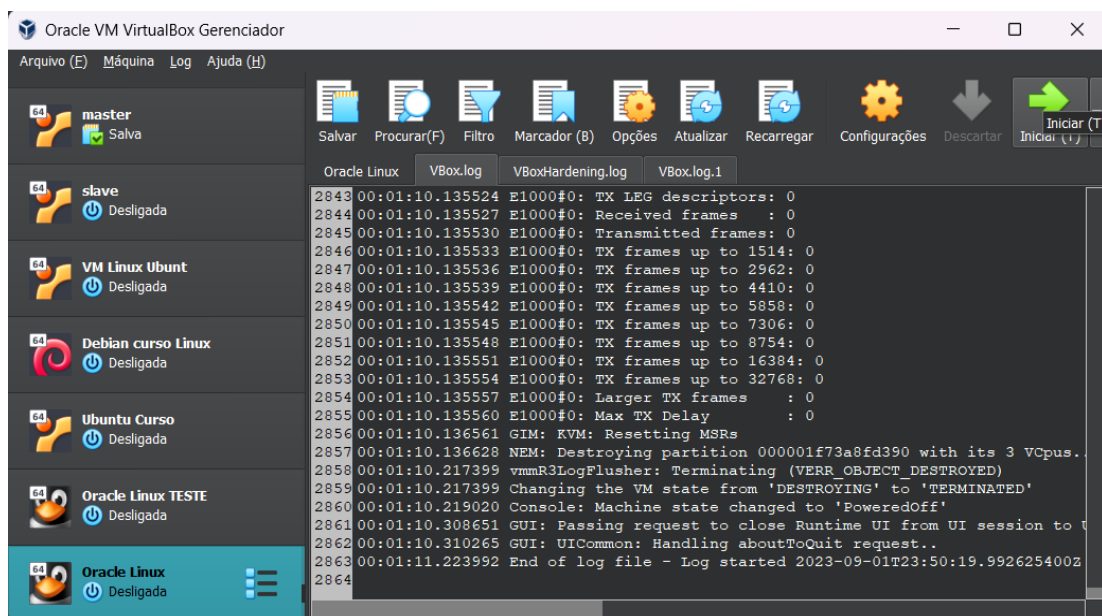
- Aqui você pode ver as configurações resultantes da máquina.



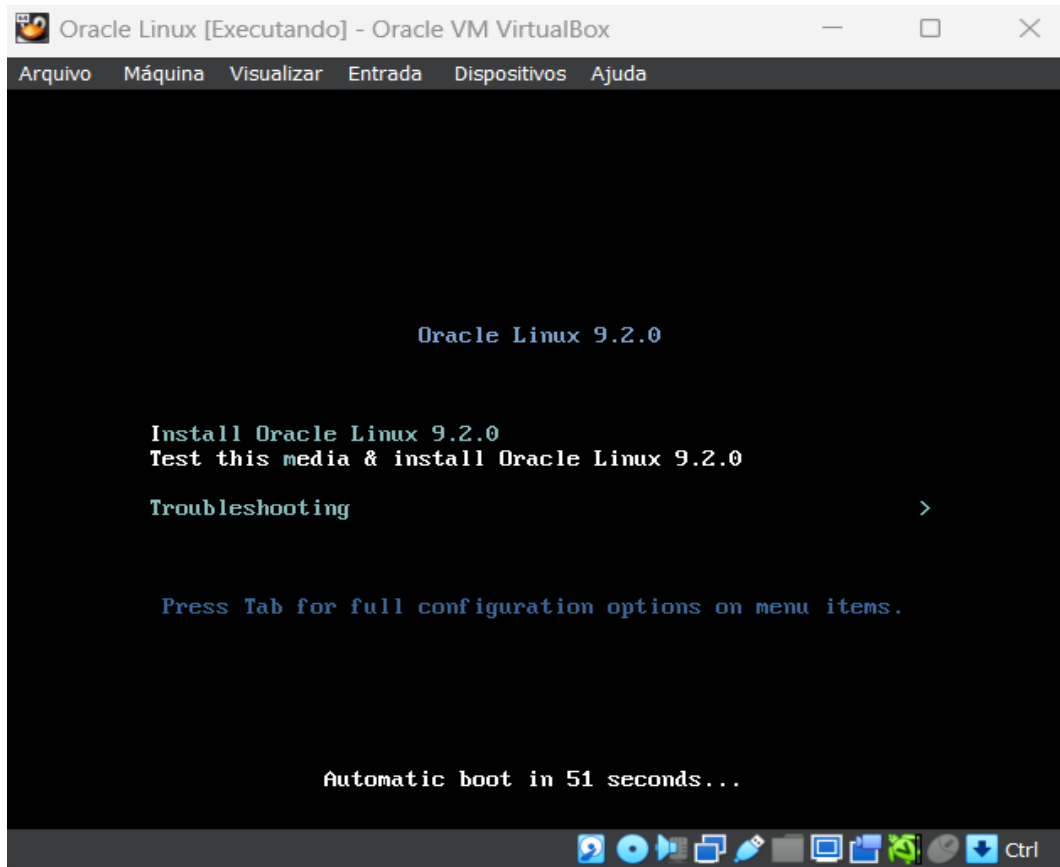
- Em seguida, será possível definir o armazenamento da máquina Virtual, nesse caso, deixei o espaço padrão, 20GB, após, é só clicar em próximo e ir para os passos seguintes.



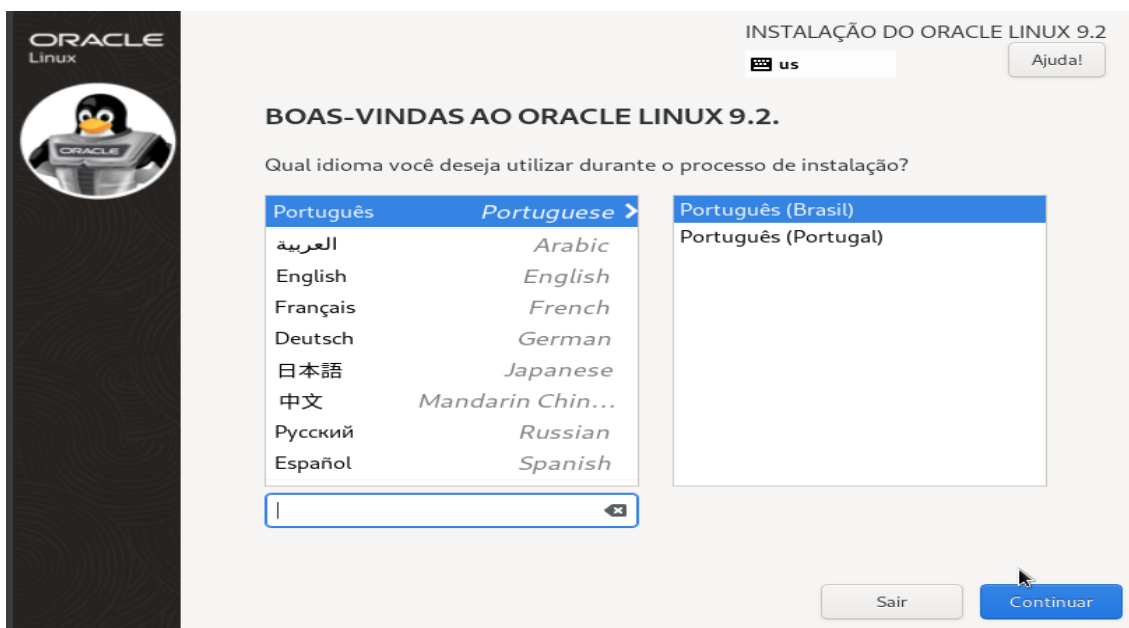
- Máquina pré-configurada, agora é só selecioná-la no canto esquerdo e clicar em iniciar para começar a instalação.



- Iniciada a Instalação, clique na primeira opção, 'Install Oracle Linux 9.2.0'.



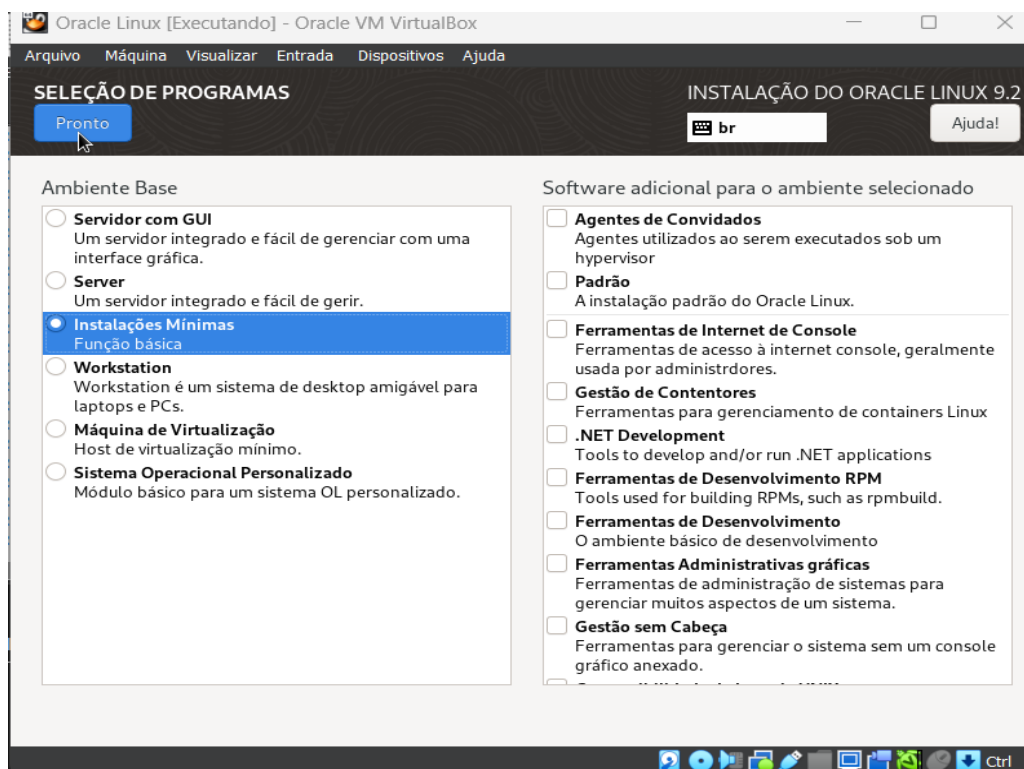
- Feito isso, iniciará o menu do Linux para definir as configurações básicas que você desejar. A primeira é o idioma, à sua preferência.



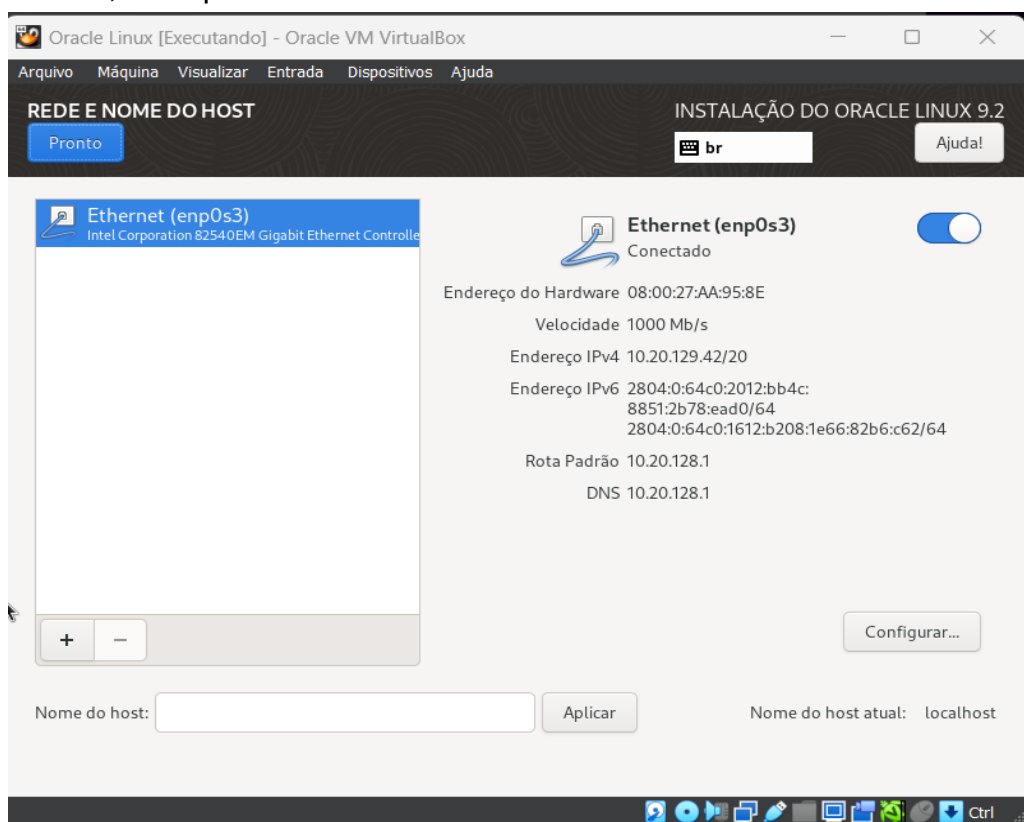
- Após, irá abrir o seguinte Menu:



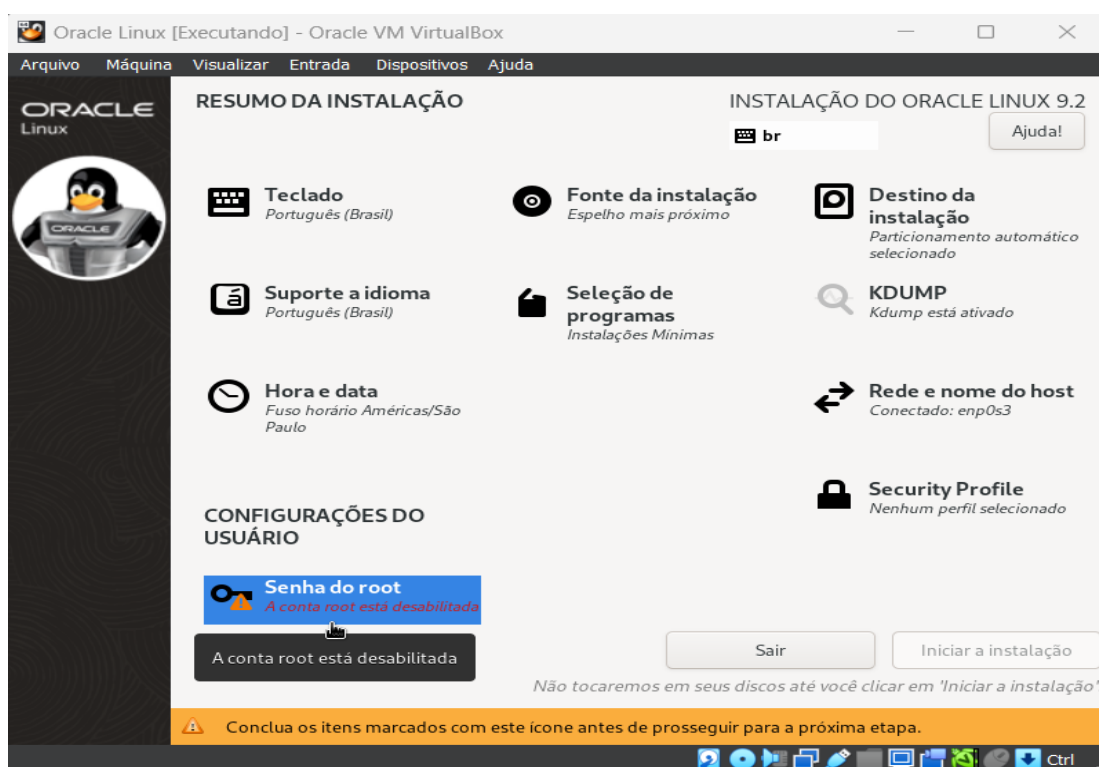
- Primeiramente, como o Linux será instalado sem o Modo gráfico, você precisa ir na opção 'Seleção de Programas' e selecionar a opção 'Instalações Mínimas', para assim, instalar apenas o console e demais configurações necessárias para funcionar corretamente o sistema.



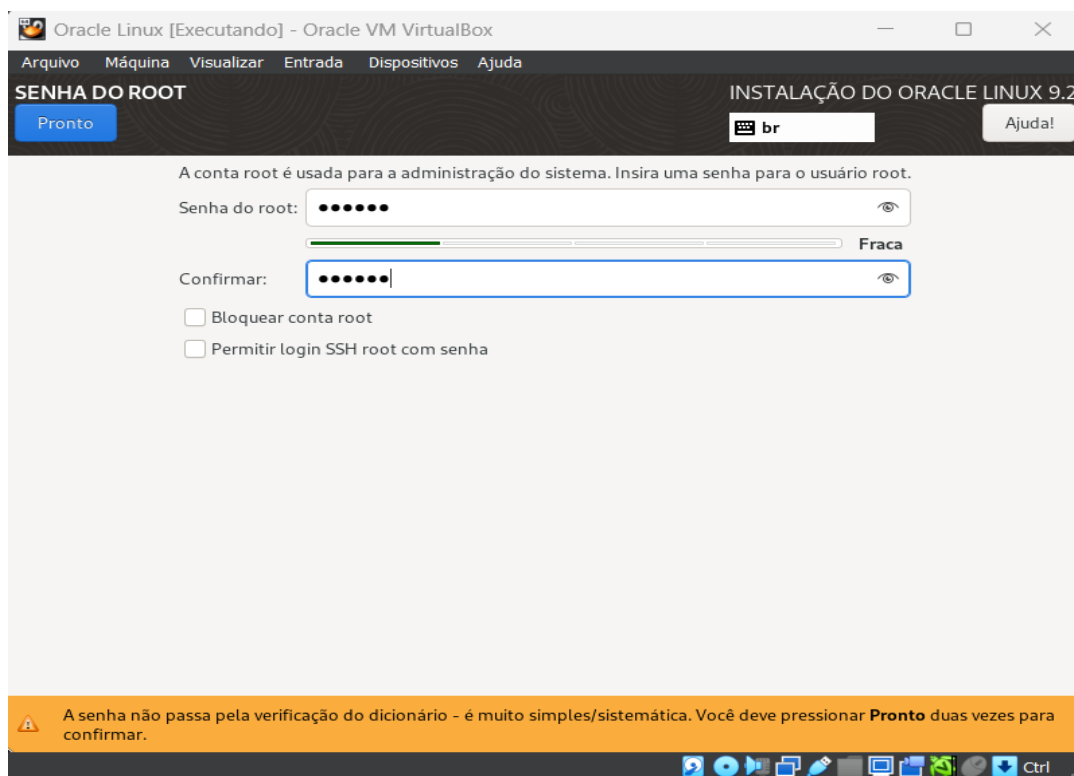
- Após, o bom é sempre conferir as configurações de rede na opção 'Rede e nome do host', se a interface de rede enp0s3 está habilitada, caso não estiver, você pode habilitar manualmente.



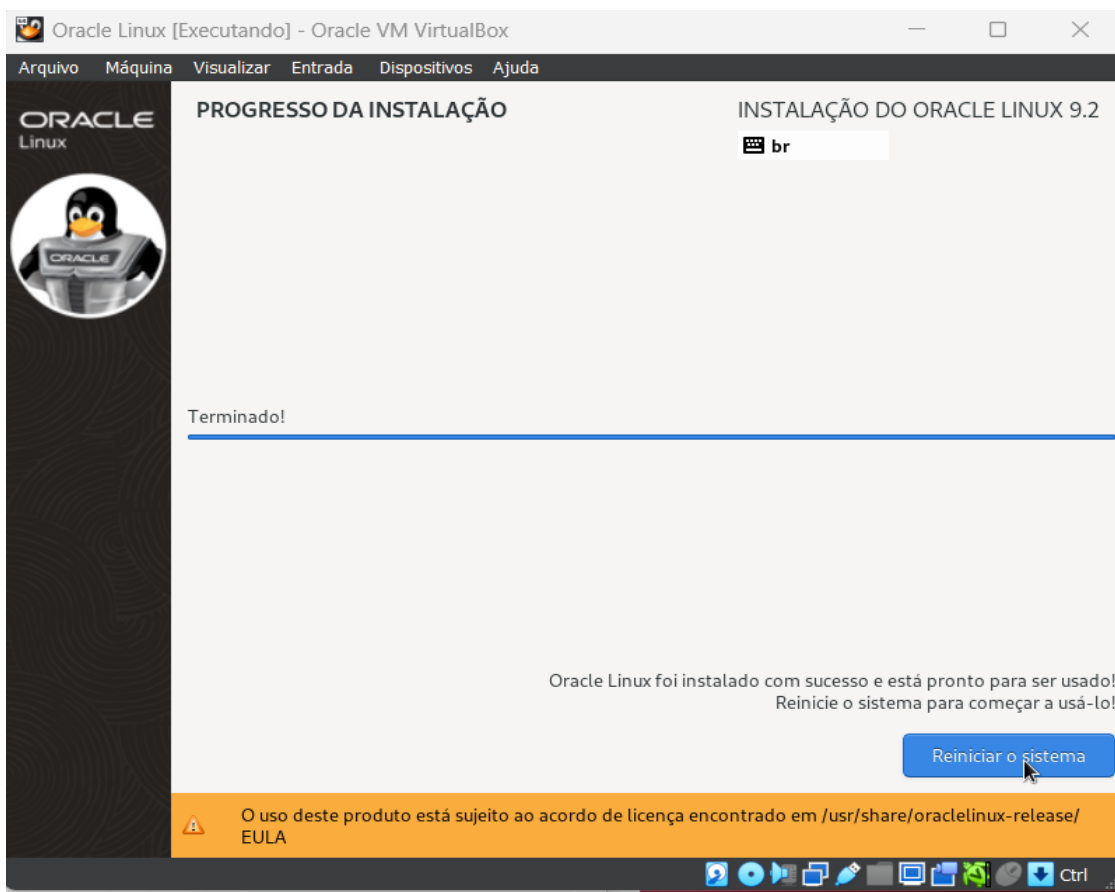
- E por último, você deve ativar a conta root, conforme imagem abaixo.



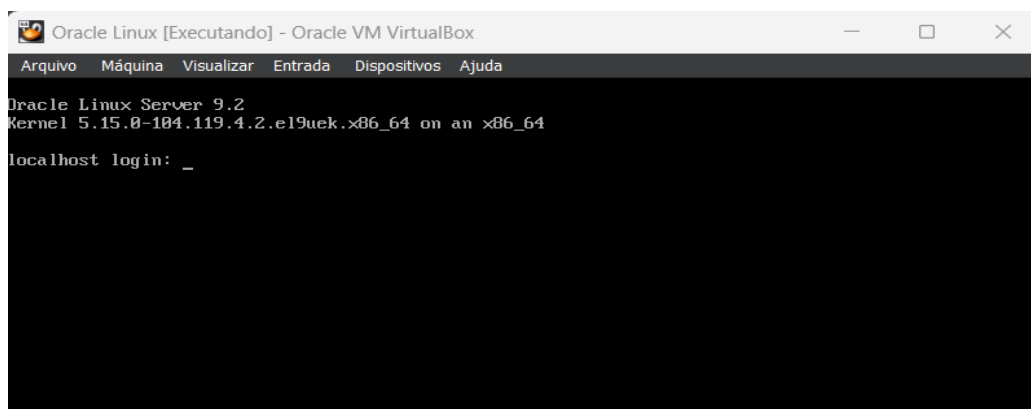
- Para ativá-la é bem simples, apenas colocar a senha de seu gosto e assim estará ok. Certifique-se de escolher uma senha forte, neste caso, escolhi uma fácil pois será utilizado apenas para testes.



- Pronto, é só clicar em iniciar instalação e após, reiniciar o sistema conforme imagens abaixo!



- Iniciado o sistema, pode usufruir dos seus recursos.



2.2 - Configurando o NFS

- Primeiramente, o que é NFS?
NFS (Network File System) é nada mais do que um protocolo de comunicação utilizado em sistemas de computadores para permitir que um computador acesse arquivos e recursos em outro dentro da mesma rede. É muito usado nos ambientes Linux e Unix para compartilhar arquivos e diretórios entre sistemas.
- Para realizar a configuração do seu NFS, o primeiro passo é executar o comando “sudo yum install nfs-utils”. Após a instalação, devemos criar uma pasta que irá conter os arquivos compartilhados no nfs. Nas instâncias amazon Linux 2, o serviço NFS já vem instalado nativamente.

```
[root@ip-10-0-1-210 ec2-user]# sudo yum install nfs-utils
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core | 3.7 kB 00:00
amzn2extra-docker | 3.0 kB 00:00
amzn2extra-kernel-5.10 | 3.0 kB 00:00
(1/7): amzn2-core/2/x86_64/group_gz | 2.5 kB 00:00
(2/7): amzn2-core/2/x86_64/updateinfo | 677 kB 00:00
(3/7): amzn2extra-docker/2/x86_64/primary_db | 108 kB 00:00
(4/7): amzn2extra-kernel-5.10/2/x86_64/updateinfo | 35 kB 00:00
(5/7): amzn2extra-docker/2/x86_64/updateinfo | 12 kB 00:00
(6/7): amzn2extra-kernel-5.10/2/x86_64/primary_db | 21 MB 00:00
(7/7): amzn2-core/2/x86_64/primary_db | 76 MB 00:01
Package 1:nfs-utils-1.3.0-0.54.amzn2.0.2.x86_64 already installed and latest version
Nothing to do
[root@ip-10-0-1-210 ec2-user]#
```

- Comando para startar o serviço de nfs: ‘systemctl start nfs-server’

```
[root@ip-10-0-1-210 ~]# systemctl start nfs-server
[root@ip-10-0-1-210 ~]#
```


- Comando para habilitar sempre o NFS ao iniciar o servidor: 'systemctl enable nfs-server'

```
[root@ip-10-0-1-210 ~]# 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.
[root@ip-10-0-1-210 ~]#
```

- Em seguida, utilizando o comando 'mkdir /claudio', criamos o diretório onde iremos compartilhar na raiz, em seguida, podemos definir permissão máxima de leitura, escrita e execução a todos usuários e grupos com o comando 'chmod -R 777 /claudio', como nas imagens a seguir:

```
[root@ip-10-0-1-210 /]# mkdir /claudio
[root@ip-10-0-1-210 /]# ls
bin    claudio  etc     lib      local   mnt
boot  dev      home   lib64    media   nfs
[root@ip-10-0-1-210 /]#
```

```
[root@ip-10-0-1-210 /]# chmod -R 777 /claudio
[root@ip-10-0-1-210 /]#
```

- Em seguida, devemos configurar a pasta, para que ela seja compartilhada no NFS. Para isso, devemos alterar o arquivo exports usando o editor vim, com o seguinte comando 'vim /etc/exports', adicionando o comando "/claudio *(rw,sync,no_root_squash,no_subtree_check)", contendo o diretório e as devidas permissões.

```
root@ip-10-0-1-210:/
/claudio *(rw,sync,no_root_squash,no_subtree_check)
```

- Agora você pode salvar, no 'vim', pode-se utilizar os seguintes comandos simultaneamente no teclado 'esc + shift + z + z', salvando e fechando o arquivo editado automaticamente. Caso desejar verificar se deu certo, no diretório raiz, é só digitar cat /etc/exports, aparecendo o conteúdo caso tenha êxito anteriormente.
- Em seguida, executa-se o comando 'exportfs -a', que é usado para atualizar as configurações de exportação do NFS e aplicar quaisquer alterações feitas no arquivo de configuração "/etc/exports".

```
[root@ip-10-0-1-210 ~]# exportfs -a
```

Após, restarte o serviço para aplicar as mudanças com o comando 'systemctl restart nfs-server':

```
[root@ip-10-0-1-210 ~]# systemctl restart nfs-server
```

Após isso, digite o comando 'showmount -e', para ver o diretório a qual está a ser compartilhado.

```
[root@ip-10-0-1-210 ~]# showmount -e
Export list for ip-10-0-1-210.ec2.internal:
/claudio *
```

- Na máquina cliente, que pode ser uma outra instância, inicialize os pacotes do NFS com o comando 'sudo yum install nfs-utils -y'
 - obs: o '-y' serve para dar um yes automaticamente para todas as requisições ao usuário durante a instalação dos pacotes, auxiliando na eficiência do processo como um todo)
- Veja a seguir o comando na CLI:

```
[ec2-user@ip-10-0-1-138 ~]$ sudo yum install nfs-utils -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core
amzn2extra-docker
amzn2extra-kernel-5.10
(1/7): amzn2-core/2/x86_64/group_gz
(2/7): amzn2-core/2/x86_64/updateinfo
(3/7): amzn2extra-docker/2/x86_64/primary_db
(4/7): amzn2extra-kernel-5.10/2/x86_64/updateinfo
(5/7): amzn2extra-docker/2/x86_64/updateinfo
(6/7): amzn2extra-kernel-5.10/2/x86_64/primary_db
(7/7): amzn2-core/2/x86_64/primary_db
Package 1:nfs-utils-1.3.0-0.54.amzn2.0.2.x86_64 already installed and latest version
Nothing to do
[ec2-user@ip-10-0-1-138 ~]$
```

- Comando 'sudo su' e 'mkdir claudio_kosooski' para ser hospedeiro do compartilhamento.
- Após isso, você pode montar o diretório pegando /claudio do ip 10.0.1.210 que é a máquina server 'mount -t nfs 10.0.1.210:/claudio /claudio_kosooski'
- Para conferir se deu certo, digite o comando 'df -h' para listar os diretórios presentes e ali estará a outra instância compartilhada

```

bin boot dev diretório etc home lib lib64 local media mnt opt proc root run s
n srv sys tmp usr var
[root@ip-10-0-1-138 /]# mkdir claudio_kosooski
[root@ip-10-0-1-138 /]# ls
bin boot claudio_kosooski dev diretório etc home lib lib64 local media mnt opt
oc root run sbin srv sys tmp usr var
[root@ip-10-0-1-138 /]# df -h
Filesystem      Size  Used Avail Use% Mounted on
devtmpfs        954M   0  954M   0% /dev
tmpfs           963M   0  963M   0% /dev/shm
tmpfs           963M 408K  962M   1% /run
tmpfs           963M   0  963M   0% /sys/fs/cgroup
/dev/nvme0n1p1  16G  1.7G   15G  11% /
tmpfs           193M   0  193M   0% /run/user/1000
[root@ip-10-0-1-138 /]# mount -t nfs 10.0.1.210:/claudio /claudio_kosooski
[root@ip-10-0-1-138 /]# ls
bin boot claudio_kosooski dev diretório etc home lib lib64 local media mnt opt
oc root run sbin srv sys tmp usr var
[root@ip-10-0-1-138 /]# df -h
Filesystem      Size  Used Avail Use% Mounted on
devtmpfs        954M   0  954M   0% /dev
tmpfs           963M   0  963M   0% /dev/shm
tmpfs           963M 412K  962M   1% /run
tmpfs           963M   0  963M   0% /sys/fs/cgroup
/dev/nvme0n1p1  16G  1.7G   15G  11% /
tmpfs           193M   0  193M   0% /run/user/1000
10.0.1.210:/claudio 16G  1.7G   15G  11% /claudio_kosooski
[root@ip-10-0-1-138 /]#

```

- Após verificado com o comando `df -h`, podemos fazer a seguinte validação: Na máquina server, entrar no diretório `'cd claudio/claudio_kosooski'` e criar um arquivo qualquer, no meu caso criei o `'teste.txt'` e escrever algo nele para exemplo. Logo após, você pode logar na instância cliente, entrar também no diretório `'cd claudio/claudio_kosooski'` e dar o comando `'ls'` para listar o conteúdo, caso possua o arquivo `'teste.txt'`, o servidor NFS estará em pleno funcionamento

```

[root@ip-10-0-1-210 claudio_kosooski]# nano teste
[root@ip-10-0-1-210 claudio_kosooski]# la
-bash: la: command not found
[root@ip-10-0-1-210 claudio_kosooski]# ls
teste.txt

```

```

[root@ip-10-0-1-138 claudio_kosooski]# ls
claudio_kosooski diretório
[root@ip-10-0-1-138 claudio_kosooski]# cd claudio_kosooski/
[root@ip-10-0-1-138 claudio_kosooski]# ls
teste.txt

```

2.3 - Criação do diretório

- Como solicitado, para a criação do diretório com meu nome dentro do NFS, feito nos passos anteriores já, é necessário primeiramente acessar a pasta do NFS, `'cd /claudio'`, após, digitar o comando `'mkdir claudio_kosooski'` para criar o diretório com meu nome, basicamente seria isso.

2.4- Instalação do Servidor Apache

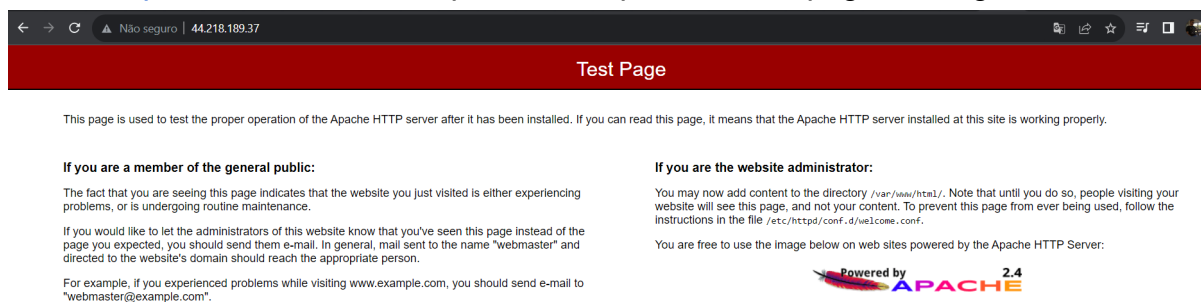
- Acessando a instância criada para a implementação do servidor NFS via SSH, podemos realizar a instalação do servidor Apache.
- Primeiramente, após o acesso, devemos digitar o comando 'sudo su' para acessar como root, no meu caso, fiz os acessos via Putty e deixei o Root como primário.
- Instância acessada, podemos executar o comando 'yum install httpd -y' para fazermos a instalação do servidor Apache.

```
https://aws.amazon.com/amazon-linux-2/
[root@ip-10-0-1-210 ~]# yum install httpd -y
```

- Após instalado com sucesso, podemos iniciar o serviço, com o comando 'service httpd start' e coletar o status com o comando 'service httpd status' se está ativo 'active (running)' ou inativo 'inactive (dead)' conforme imagem a seguir:

```
https://aws.amazon.com/amazon-linux-2/
[root@ip-10-0-1-210 ~]# service httpd start
Redirecting to /bin/systemctl start httpd.service
[root@ip-10-0-1-210 ~]# service httpd status
Redirecting to /bin/systemctl status httpd.service
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; vendor prese
t: disabled)
   Active: active (running) since Sun 2023-09-03 20:34:15 UTC; 20s ago
     Docs: man:httpd.service(8)
   Main PID: 5846 (-httpd)
```

- Agora é necessário configurar para que o apache inicie com o sistema operacional. Para isso usamos o comando "systemctl enable httpd". Isso irá garantir que cada vez que a máquina iniciar o serviço do apache irá iniciar também. Após todos os passos o serviço do apache deverá estar iniciado e rodando. Para ter certeza de que tudo funcionou devemos nos conectar ao endereço ip público do servidor apache usando o navegador, digitamos '<https://44.218.189.37>' e pronto, se aparecer esta página a seguir, estará ok.



2.5 - Criação dos Scripts de Validação de Status

- Para a criação do script usamos o editor de texto nano, passando os parâmetros de nome de arquivo e extensão, que no caso é “.sh”, extensão dos arquivos shell.
- No script, há uma análise se o serviço está online ou offline, caso estiver online, irá definir uma mensagem de log .txt .

```
[root@ip-10-0-1-210 claudio_kosooski]# nano script_status_httpd.sh
```

```
[root@ip-10-0-1-210 claudio_kosooski]# ls
script status httpd.sh teste.txt
```

- Este é o script:

```
GNU nano 2.9.8 script_status httpd.sh
#!/bin/bash
service_name="Apache"
timestamp=$(date +%Y-%m-%d %H:%M:%S)

if systemctl is-active httpd; then
    echo "Data/Hora: $timestamp - $service_name - Serviço HTTPD - Status: O serviço está online" >> "/claudio/claudio_kosooski/httpd-online.txt"
else
    echo "Data/Hora: $timestamp - $service_name - Serviço HTTPD - Status: O serviço está offline" >> "/claudio/claudio_kosooski/httpd-offline.txt"
fi
```

- Após criado, devemos dar permissões de execução ao arquivo .sh, para isso usamos o comando ‘chmod =x /claudio/claudio_kosooski/script_status_httpd.sh’, onde diretório é o diretório em que o arquivo .sh está localizado, conforme imagem a seguir:

```
[root@ip-10-0-1-210 claudio_kosooski]# chmod =x /claudio/claudio_kosooski/script_status_httpd.sh
[root@ip-10-0-1-210 claudio_kosooski]#
```

- Para conferir as permissões dos arquivos no diretório, podemos usar o comando ‘ls -l’ para verificar, conforme abaixo, onde mostra a permissão dos dois arquivos do repositório

```
GNU coreutils online help: <http://www.gnu.org/software/coreutils/>
For complete documentation, run: info coreutils 'chmod invocation'
[root@ip-10-0-1-210 claudio_kosooski]# ls -l
total 8
--x--x--x 1 root root 538 Sep  3 22:55 script_status_httpd.sh
-rw-r--r-- 1 root root  6 Sep  3 20:00 teste.txt
[root@ip-10-0-1-210 claudio_kosooski]#
```

- Agora, para fazer a execução do script de 5 em 5 minutos conforme solicitado, devemos usar um agendador de tarefas no Linux. Nesse caso, utilizei o contrab, para acessá-lo usamos o comando “crontab -e”, e na tela

que abriu adicionamos o seguinte comando ‘*/5 * * * *
/claudio/claudio_kosooski/script_status_httpd.sh’.

```
For complete documentation, run: info coreutils 'chmod invocation
[root@ip-10-0-1-210 claudio_kosooski]# ls -l
total 8
---x--x--x 1 root root 538 Sep  3 22:55 script_status_httpd.sh
-rw-r--r-- 1 root root  6 Sep  3 20:00 teste.txt
[root@ip-10-0-1-210 claudio_kosooski]# crontab -e
no crontab for root - using an empty one
crontab: installing new crontab
[root@ip-10-0-1-210 claudio_kosooski]#
```

- Com o comando anterior, abriu o editor vim sem dados, com isso, adicionamos o script conforme dito anteriormente e salvamos usando o ‘esc + shift + z +z’ novamente:

```
*/5 * * * * /claudio/claudio_kosooski/script_status_httpd.sh
```

Após salvar o script no ‘crontab -e’, deve-se validar se foi criado os arquivos para salvar o status do httpd, para isso, no diretório ‘claudio/claudio_kosooski’, digite ‘ls’ e valide se possui os arquivos ‘httpd-online.txt’ e ‘httpd-offline.txt’, caso não, crie para que seja possível salvar o log corretamente.

```
[root@ip-10-0-1-210 claudio]# cd claudio_kosooski/
[root@ip-10-0-1-210 claudio_kosooski]# ls
httpd-offline.txt httpd-online.txt script_status_httpd.sh teste.txt
[root@ip-10-0-1-210 claudio_kosooski]#
```

Feito isso, é só correr pro abraço, entre no arquivo com o comando ‘nano httpd-online.txt’ após os 5 minutos e se estiver com log, é porque tudo ocorreu conforme esperado:

```
root@ip-10-0-1-210/claudio/claudio_kosooski
GNU nano 2.9.8 httpd-online.txt
Data/Hora: 2023-09-04 00:23:40 - Apache - Serviço HTTPD - Status: O serviço está online
Data/Hora: 2023-09-04 00:25:01 - Apache - Serviço HTTPD - Status: O serviço está online
Data/Hora: 2023-09-04 00:30:01 - Apache - Serviço HTTPD - Status: O serviço está online
Data/Hora: 2023-09-04 00:35:01 - Apache - Serviço HTTPD - Status: O serviço está online
Data/Hora: 2023-09-04 00:40:01 - Apache - Serviço HTTPD - Status: O serviço está online
```

Também é possível visualizar o status sem abrir o arquivo, só estando no diretório e digitando ‘head nome_do arquivo’, por exemplo:

```
[root@ip-10-0-1-210 claudio_kosooski]# head httpd-online.txt
Data/Hora: 2023-09-04 00:23:40 - Apache - Serviço HTTPD - Status: O serviço está online
Data/Hora: 2023-09-04 00:25:01 - Apache - Serviço HTTPD - Status: O serviço está online
Data/Hora: 2023-09-04 00:30:01 - Apache - Serviço HTTPD - Status: O serviço está online
Data/Hora: 2023-09-04 00:35:01 - Apache - Serviço HTTPD - Status: O serviço está online
```

Pronto, o script de validação de status do Apache está ok.