

Atividade prática

TALITA VARGAS DE SOUZA

Parte 1: AWS

PASSO 1:

Gerar uma chave pública para acesso ao ambiente;

PASSO 2:

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

PASSO 3:

Gerar 1 elastic IP e anexar à instância EC2;

PASSO 4:

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

PASSO 1:

Gerar uma chave pública para acesso ao ambiente;

Ao iniciar a criação de uma instância você irá ver que aparecerá a opção “Par de chaves (login)”, e o seguinte texto: “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.” Como na imagem abaixo:

The screenshot shows the 'Par de chaves (login)' section of the AWS CloudFormation 'Create New Stack' wizard. It includes a note about using a key pair for secure connection, a dropdown menu for selecting an existing key pair ('Selecionar'), and a button to 'Criar novo par de chaves' (Create new key pair).

Caso você ainda não tenha um par de chaves, clique em “Criar novo par de chaves”. De um nome a sua chave e mantenha as configurações que aparecem na imagem abaixo:

The screenshot shows the 'Criar par de chaves' (Create new key pair) dialog box. It includes fields for 'Nome do par de chaves' (Key pair name), a note about key pairs for secure connection, a text input field for 'Insira o nome do par de chaves' (Insert key pair name), a note about character limits, a 'Tipo de par de chaves' (Key pair type) section with 'RSA' selected, a note about RSA keys being public and private, a 'Formato de arquivo de chave privada' (Private key file format) section with '.pem' selected, a note about PEM format being for OpenSSH, a note about .ppk format being for PuTTY, and a warning message about storing the private key securely.

PASSO 1:

Gerar uma chave pública para acesso ao ambiente;

Em seguida a chave será baixada em seu aparelho. Ao utilizar o WSL (Windows Subsystem for Linux), uma tecnologia desenvolvida pela Microsoft que permite a execução de distribuições Linux diretamente em sistemas Windows. Faça uma cópia da chave para a sua pasta de usuário dentro do sistema utilizado como no exemplo a seguir:

Linux > Ubuntu-22.04 > home > talita > compass				
	Nome	Data de modificação	Tipo	Tamanho
	talita.pem	28/08/2023 14:24	Arquivo PEM	2 KB

Com isso você tem sua chave criada e salva no local correto.

PASSO 2:

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

Em sua conta da AWS, entre no serviço de EC2, em seguida clique em "Instâncias" e posteriormente em "Executar Instâncias".

Na parte de nome e tags, você deve utilizar essas tags:

Name: PB - FW - A - RG - SB - HA

CostCenter: C092000004

Project: PB - FW - A - RG - SB - HA

Não se esqueça de marcar as opções "Instâncias" e "Volumes" em tipos de recursos.

▼ Nome e tags [Informações](#)

Chave Informações	Valor Informações	Tipos de recurso Informações	Remover
<input type="text" value="Name"/> X	<input type="text" value="PB - FW - A - RG"/> X	<input type="button" value="Selecionar tipos d..."/>	<input type="button" value="Remover"/>
		<input type="checkbox"/> Instâncias X	
		<input type="checkbox"/> Volumes X	
Chave Informações	Valor Informações	Tipos de recurso Informações	Remover
<input type="text" value="CostCenter"/> X	<input type="text" value="C092000004"/> X	<input type="button" value="Selecionar tipos d..."/>	<input type="button" value="Remover"/>
		<input type="checkbox"/> Instâncias X	
		<input type="checkbox"/> Volumes X	
Chave Informações	Valor Informações	Tipos de recurso Informações	Remover
<input type="text" value="Project"/> X	<input type="text" value="PB - FW - A - RG"/> X	<input type="button" value="Selecionar tipos d..."/>	<input type="button" value="Remover"/>
		<input type="checkbox"/> Instâncias X	
		<input type="checkbox"/> Volumes X	

Você pode adicionar até mais 47 etiquetas.

PASSO 2:

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

Na parte de imagens de aplicação e de sistema operacional selecione a opção Amazon Linux 2.

▼ **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) necessário para executar a instância. Pesquise ou navegue pelas AMIs se você não estiver vendo o que está buscando abaixo

Pesquise nosso catálogo completo, incluindo milhares de imagens de aplicações e sistemas operacionais

[Recentes](#) [Início rápido](#)

 [Procurar mais AMIs](#)
Incluindo AMIs da AWS, do Marketplace e da comunidade

Imagen de máquina da Amazon (AMI)

Amazon Linux 2 AMI (HVM) - Kernel 5.10, SSD Volume Type Qualificado para o nível gratuito ▾
ami-0f409bae3775dc8e5 (64 bits (x86)) / ami-0f0f7b386be96ec2d (64 bits (Arm))
Virtualização: hvm ENA habilitado: true Tipo de dispositivo raiz: ebs

Descrição
Amazon Linux 2 Kernel 5.10 AMI 2.0.20230822.0 x86_64 HVM gp2

Arquitetura 64 bits (x86) ▾ **ID da AMI** ami-0f409bae3775dc8e5 **Provedor verificado**

PASSO 2:

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

No tipo de instância selecione t3.small.

Tipo de instância [Informações](#)

Tipo de instância

t3.small

Família: t3 2 vCPU 2 GiB Memória Geração atual: true
Sob demanda SUSE base definição de preço: 0.0518 USD por hora
Sob demanda Linux base definição de preço: 0.0208 USD por hora
Sob demanda RHEL base definição de preço: 0.0808 USD por hora
Sob demanda Windows base definição de preço: 0.0392 USD por hora

Additional costs apply for AMIs with pre-installed software

Todas as gerações

[Comparar tipos de instância](#)

Em par de chaves selecione a sua chave ou crie como citado no passo 1.

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

[Criar novo par de chaves](#)

PASSO 2:

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

Em configurações de rede certifique-se de selecionar uma subnet pública, caso não tenha uma, faça as configurações necessárias.

The screenshot shows the 'Network configurations' section of a CloudFormation template. It includes a dropdown for the VPC (selected: 'vpc-06a851b08cc6e9ad4 (aws-controltower-VPC) 172.31.0.0/16'), a dropdown for the subnet ('subnet-0206a9ddfc556f836 publica1a'), and a checkbox for 'Enable automatic public IP assignment' (unchecked). A 'Create new subnet' button is also visible.

Você deve criar um novo grupo de segurança, posteriormente iremos configurá-lo. De início selecione apenas a opção de ssh.

The screenshot shows the 'Security Group (Security Groups)' section of a CloudFormation template. It includes a radio button for 'Create security group' (selected), a radio button for 'Select existing security group', and input fields for 'Name' ('de_um_nome') and 'Description' ('coloque_uma_descricao'). A note states that the security group will be added to all network interfaces and lists allowed characters (a-z, A-Z, 0-9, spaces, and specific symbols).

PASSO 2:

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

Regras do grupo de segurança de entrada

▼ Regra de grupo de segurança 1 (TCP, 22, 0.0.0.0/0) Remover

Tipo	Informações	Protocolo	Informações	Intervalo de portas	Informações
ssh		TCP		22	

Tipo de origem	Informações	Origem	Informações	Descrição - optional	Informações
Qualquer lugar			Adicionar CIDR, lista de prefixo	p. ex. SSH para a área de trabalho, 0.0.0.0/0	

Adicionar regra de grupo de segurança

► Configuração avançada de rede

Modifique o tamanho do volume para 16 GB.

▼ Armazenamento (volumes) Informações Simples

Volumes do EBS Ocultar detalhes

▼ Volume 1 (Raiz da AMI) (Personalizada)

Tipo de armazenamento	Informações	Nome do dispositivo - required	Informações	Snapshot	Informações
EBS		Informações	/dev/xvda	snap-0b41aa919c7bcb5a6	

Tamanho (GiB)	Informações	Tipo de volume	Informações	IOPS	Informações
16		gp2		100 / 3000	

Excluir no encerramento	Informações	Criptografado	Informações	Chave do KMS	Informações
Sim		Não criptografado		Selecionar	

Adicionar novo volume

Sistemas de arquivos Mostrar detalhes

PASSO 2:

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

Em seguida clique em "Executar instância"

▼ Resumo

Número de instâncias [Informações](#)

1

Imagen do software (AMI)

Amazon Linux 2 Kernel 5.10 AMI...[Ler mais](#)
ami-0f409bae3775dc8e5

Tipo de servidor virtual (tipo de instância)

t3.small

Firewall (grupo de segurança)

Novo grupo de segurança

Armazenamento (volumes)

1 volume(s) - 16 GiB

[Cancelar](#) [Executar instância](#) [Revisar comandos](#)

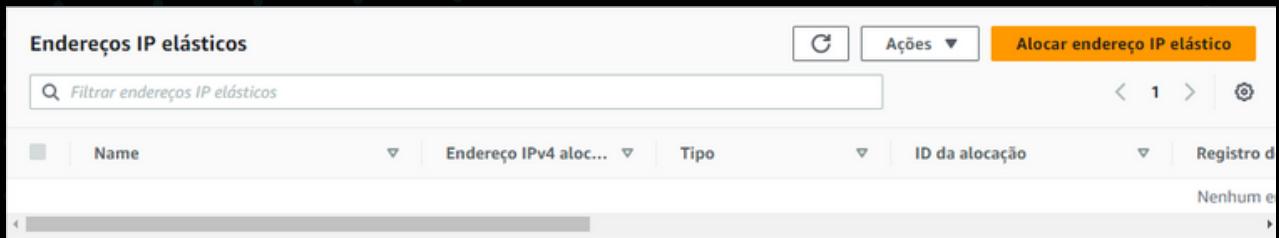
Em poucos minutos sua instância vai estar em execução e você poderá usá-la.

Instâncias (1) Informações		<input type="button" value="C"/>	Conectar	Estado da instância	Ações	Executar instâncias
<input type="text"/> Localizar instância por atributo ou tag (case-sensitive)						
Name	ID de instância	Estado da instância	Tipo de inst...	Verificação de status		
<input type="checkbox"/> PB - FW - A - R...	i-09cc51ebd73d879c6	<input checked="" type="checkbox"/> Executando	<input checked="" type="checkbox"/> t3.small	<input checked="" type="checkbox"/> 2/2 verificações aprovadas		

PASSO 3:

Gerar 1 elastic IP e anexar à instância EC2;

Na aba de IPs elásticos clique em "Alocar endereço IP elástico"



Name	Endereço IPv4 aloc...	Tipo	ID da alocação	Registro d...
Nenhum e...				

Mantenha as configurações padrões e adicione as suas tags

Alocar endereço IP elástico Informações

Configurações de endereço IP elástico Informações

Grupo de Borda de Rede Informações

Q us-east-1 X

Conjunto de endereços IPv4 públicos

- Conjunto de endereços IPv4 da Amazon
- Endereço IPv4 público que você leva para sua conta da AWS (opção desativada porque nenhum conjunto foi encontrado) [Saiba mais](#)
- Grupo com os endereços IPv4 pertencentes ao cliente (opção desabilitada porque nenhum grupo pertencente ao cliente foi encontrado) [Saiba mais](#)

Endereços IP estáticos globais

O AWS Global Accelerator pode fornecer endereços IP estáticos globais anunciados em todo o mundo usando anycast de pontos de presença da AWS. Isso pode ajudar a melhorar a disponibilidade e a latência do tráfego de usuários usando a rede global da Amazon. [Saiba mais](#)

[Criar acelerador](#)

Tags opcional

Uma tag é um rótulo que você atribui a um recurso da AWS. Cada tag consiste em uma chave e um valor opcional. É possível usar as tags para pesquisar e filtrar seus recursos ou para controlar seus custos da AWS.

Chave

Q Name X

Valor - opcional

Q meu_ip X

[Remover](#)

[Adicionar nova tag](#)

Você pode adicionar até mais 49 tags

[Cancelar](#)

[Alocar](#)

PASSO 3:

Gerar 1 elastic IP e anexar à instância EC2;

Clique em alocar, posteriormente selecione o IP elástico, vá em ações na opção associar endereço IP elástico

Endereços IP elásticos (1/1)			
Ações Alocar endereço IP elástico			
Name	Endereço IPv4 aloc...	Tipo	Ações
meu_ip	107.23.100.190	IP público	eipalloc-0b0dce591a1f25483

Selecione seus dados como mostrado no exemplo abaixo:

Associar endereço IP elástico Informações

Selecione a instância ou a interface de rede para associar a esse endereço IP elástico (3.222.191.166)

Endereço IP elástico: 3.222.191.166

Tipo de recurso
Selecione o tipo de recurso ao qual associar o endereço IP elástico.

Instância
 Interface de rede

⚠️ Se você associar um endereço IP elástico a uma instância que já tem um endereço IP elástico associado, o endereço IP elástico associado anteriormente será desassociado, mas o endereço ainda estará alocado à sua conta. [Saiba mais](#)

Se nenhum endereço IP privado for especificado, o endereço IP elástico será associado ao endereço IP privado primário.

Interface de rede
eni-0fb5afc20785dac25

Endereço IP privado
O endereço IP privado ao qual associar o endereço IP elástico.

172.31.13.158

Reassociação
Especifique se o endereço IP elástico pode ser reassociado a um recurso diferente se ele já estiver associado a um recurso.

Permitir que o endereço IP elástico seja reassociado

[Cancelar](#) [Associar](#)

Feito isso ele já estará associado a sua instância e poderá usá-lo para acessá-la

PASSO 4:

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

Acesse a aba Security groups, e clique no ID do grupo de segurança que está sendo usado na sua instância.

The screenshot shows the AWS Management Console interface for security groups. At the top, there's a header with 'Grupos de segurança (2)', 'Informações', 'Ações', 'Exportar grupos de segurança para CSV', and a yellow 'Criar grupo de segurança' button. Below the header is a search bar labeled 'Filtrar grupos de segurança'. A table lists two security groups: 'default' (ID: sg-03ffaee8a7a061703) and 'grupo' (ID: sg-05b4b0cf0dbaaa4b2). The 'grupo' row is selected, indicated by a blue border. The table has columns for Name, ID do grupo de segurança, Nome do grupo de..., ID da VPC, and Descrição.

Name	ID do grupo de segurança	Nome do grupo de...	ID da VPC	Descrição
-	sg-03ffaee8a7a061703	default	vpc-06a851b08cc6e9ad4	default VPC security gr...
-	sg-05b4b0cf0dbaaa4b2	grupo	vpc-06a851b08cc6e9ad4	grupo de segurança linux

Em regras de entrada selecione editar regras

The screenshot shows the 'sg-05b4b0cf0dbaaa4b2 - grupo' details page. At the top, there's a breadcrumb navigation: EC2 > Grupos de segurança > sg-05b4b0cf0dbaaa4b2 - grupo. Below the breadcrumb is a 'Detalhes' section with fields: Nome do grupo de segurança (sg-grupo), ID do grupo de segurança (sg-05b4b0cf0dbaaa4b2), Descrição (grupo de segurança linux), and ID da VPC (vpc-06a851b08cc6e9ad4). The 'Proprietário' field shows the user ID 861757766714. Under the 'Regras de entrada' tab, there's a table with one rule: 'Regras de entrada (1)'. The table includes a search bar 'Filtrar regras de grupo de segurança' and buttons for 'C', 'Gerenciar tags', and 'Editar regras de entrada'.

PASSO 4:

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

E adicione as seguintes regras

Regras de entrada						
ID da regra do grupo de segurança	Tipo	Informações	Protocolo	Informações	Intervalo de portas	Origem
				Informações		Informações
sgr-0f71987ebe29b37a2	SSH		TCP	22	Person... ▾	0.0.0.0/0 X
sgr-0135ea5806936742a	NFS		TCP	2049	Person... ▾	0.0.0.0/0 X
sgr-0453fd3cce6dfc62e	TCP personalizado		TCP	111	Person... ▾	0.0.0.0/0 X
sgr-09b87728854e35d98	HTTPS		TCP	443	Person... ▾	0.0.0.0/0 X
sgr-03695a4bd9a36efaa	HTTP		TCP	80	Person... ▾	0.0.0.0/0 X

Pronto seus serviços da AWS estão configurados e você pode seguir para a parte de linux.

Parte 2: LINUX

PASSO 1:

Configurar o NFS e criar um diretorio dentro do filesystem do NFS com seu nome;

PASSO 2:

Subir um apache no servidor - o apache deve estar online e rodando;

PASSO 3:

Criar um script que valide se o serviço esta online e envie o resultado da validação para o seu diretorio no nfs;

PASSO 4:

Preparar a execução automatizada do script a cada 5 minutos.

TESTES

SUMÁRIO

PASSO 1:

Configurar o NFS e criar um diretório dentro do filesystem do NFS com seu nome

LINUX

Antes de tudo você deve saber o que é o NFS, Network File System ou em português Sistema de Arquivos em Rede. O NFS nada mais é 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 em ambientes Unix e Linux para compartilhar arquivos e diretórios entre sistemas. Ou seja, é uma maneira eficaz de tornar os arquivos disponíveis de forma centralizada para vários dispositivos em uma rede.

O primeiro passo é conectar-se à instância via SSH. Você lembra do arquivo .pem que foi salvo anteriormente? Vai precisar dele.

Abra o seu terminal Linux do WSL e digite o comando seguindo o exemplo:

```
talita@DESKTOP-NVQ80RS: ~$ chmod 400 /home/talita/chave/talita.pem  
talita@DESKTOP-NVQ80RS: ~$
```

Em seguida acesse a instância via ssh

- `ssh -i caminho/da/sua/chave.pem ec2-user@ip_publico_da_instancia`
(ip elástico associado)

```
talita@DESKTOP-NVQ80RS: ~$ ssh -i /home/talita/chave/talita.pem ec2-user@3.222.191.166  
The authenticity of host '3.222.191.166 (3.222.191.166)' can't be established.  
ED25519 key fingerprint is SHA256:/xtwpeFIVynlKRnjcVLDANTL%Y+WYrlQNWgYaohga2k.  
This host key is known by the following other names/addresses:  
~/ssh/known_hosts:27: [hashed name]  
Are you sure you want to continue connecting (yes/no/[fingerprint])?
```

Irá aparecer uma mensagem semelhante a esta, digite yes para se conectar.

PASSO 1:

Configurar o NFS e criar um diretório dentro do filesystem do NFS com seu nome

LINUX

```
talita@DESKTOP-NVQ80RS: ~$ ssh -i /home/talita/chave/talita.pem ec2-user@3.222.191.166
The authenticity of host '3.222.191.166 (3.222.191.166)' can't be established.
ED25519 key fingerprint is SHA256:/xtwpeFlVynlKRnjcVLDANTL%Y+WYrlQNWgYaohga2k.
This host key is known by the following other names/addresses:
~/.ssh/known_hosts:27: [hashed name]
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '3.222.191.166' (ED25519) to the list of known hosts.
Last login: Mon Aug 28 20:42:48 2023 from 143-208-299-57.cznet.com.br
```

```
 _|_ _|_
_|_(_/_ Amazon Linux 2 AMI
__\|_|_
```

```
https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-172-31-13-158 ~]$
```

Conexão feita com sucesso

```
[ec2-user@ip-172-31-13-158 ~]$ sudo su -
[root@ip-172-31-13-158 ~]#
```

O comando “`sudo su -`” é usado para abrir uma nova sessão de terminal como usuário root, ele é o usuário com privilégios administrativos mais altos em um sistema Linux.

- `sudo`: usado para executar comandos administrativos;
- `su`: usado para trocar de usuário, quando não é colocado argumentos adicionais, assume-se que é desejado trocar para o usuário root;
- `-`: usando o hífen após o “`su`” indica-se que o desejo de iniciar uma nova sessão de shell como o usuário especificado. Ou seja, obtém-se um ambiente de shell completamente novo, como se tivesse feito login diretamente como o usuário especificado, nesse caso o root.

Ao executar este comando você inicia uma nova sessão de terminal como root, com todas as variáveis de ambiente configuradas como se o login fosse feito diretamente como root. Isso é útil quando é preciso realizar várias operações como superusuário, pois evita a necessidade de digitar `sudo` antes de cada comando.

PASSO 1:

Configurar o NFS e criar um diretório dentro do filesystem do NFS com seu nome

LINUX

```
[root@ip-172-31-13-158 ~]# yum install nfs-utils
```

O comando “`yum install nfs-utils`” é usado em sistemas baseados em Red Hat, para instalar o pacote “nfs-utils”. Este pacote é essencial para a configuração e uso do NFS no sistema.

- `yum`: é um gerenciador de pacotes que permite a instalação, atualização e remoção de software no sistema de maneira fácil e automatizada;
- `install`: é a ação a ser realizada com o “`yum`”. Neste caso, a instrução é de instalar um pacote;
- `nfs-utils`: é o nome do pacote a ser instalado. Este pacote, é um conjunto de utilitários e ferramentas que são necessários para configurar e gerenciar o serviço NFS em um sistema.

Após a execução do comando e a confirmação da instalação, o pacote será baixado e instalado no sistema. Depois da instalação, você pode configurar e usar o NFS.

```
[root@ip-172-31-13-158 ~]# yum install nfs-utils
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core                                         | 3.7 KB  00:00:00
Package 1:nfs-utils-1.3.0-0.54.amzn2.0.2.x86_64 already installed and latest version
Nothing to do
[root@ip-172-31-13-158 ~]#
```

No entanto pode aparecer esta mensagem mostrando que no sistema Amazon Linux 2 utilizado, mostrando que o pacote já estava instalado na versão mais recente, que é “`1:nfs-utils-1.3.0-0.54.amzn2.0.2.x86_64`”. Neste caso o pacote estava instalado por padrão na instância.

PASSO 1:

Configurar o NFS e criar um diretório dentro do filesystem do NFS com seu nome

LINUX

```
[root@ip-172-31-13-158 ~]# systemctl start nfs-server  
[root@ip-172-31-13-158 ~]#
```

O comando “`systemctl start nfs-server`” é usado para iniciar o serviço NFS no sistema Linux.

- `systemctl`: é responsável pela interação com o sistema de inicialização e os serviços no Linux. Permite o controle e gerenciamento de serviços, como iniciar, parar, reiniciar e verificar o status dos mesmos;
- `start`: é a ação a ser realizada com o “`systemctl`”. Neste caso, a instrução é de iniciar o serviço especificado;
- `nfs-server`: é o nome do serviço especificado que irá iniciar. O mesmo, é responsável por gerenciar os compartilhamentos NFS no sistema.

Portanto, ao executar este comando, é iniciado o serviço NFS no sistema. Ou seja, o sistema ficará pronto para compartilhar diretórios e arquivos por meio do protocolo NFS, outros computadores na rede poderão montar esses compartilhamentos NFS e acessar os dados compartilhados.

```
[root@ip-172-31-13-158 ~]# systemctl enable nfs-server
```

O comando “`systemctl enable nfs-server`” é usado para habilitar automaticamente o serviço NFS durante o processo de inicialização do sistema Linux.

- `systemctl`: também é usado para habilitar e desabilitar serviços;
- `enable`: é a ação que será realizada com o “`systemctl`”. Neste caso, a instrução é habilitar o serviço especificado;
- `nfs-server`: é o serviço especificado que será habilitado.

Ao executar este comando o serviço de NFS é configurado para iniciar automaticamente sempre que o sistema for iniciado ou reiniciado. Isto faz com que o processo não tenha que ser feito manualmente quando houver uma necessidade de reboot do sistema.

PASSO 1:

Configurar o NFS e criar um diretório dentro do filesystem do NFS com seu nome

LINUX

```
[root@ip-172-31-13-158 ~]# systemctl enable nfs-server  
Created symlink from /etc/systemd/multi-user.target.wants/nfs-server.service to /usr/lib/systemd/system/  
/nfs-server.service.  
[root@ip-172-31-13-158 ~]#
```

A mensagem indica que o processo foi feito com sucesso, mostrando os diretórios onde foi criado o link simbólico que é usado durante o processo de inicialização para determinar quais serviços devem ser iniciados quando o sistema é ligado.

```
[root@ip-172-31-13-158 ~]# cd /  
[root@ip-172-31-13-158 /]#
```

O comando “`cd /`” é usado para mudar o diretório de trabalho atual para a raiz do sistema de arquivos.

- `cd`: é usado para navegar pelos diretórios do sistema de arquivos e significa "change directory" (mudar diretório);
- `/`: é o caminho que representa a raiz do sistema de arquivos.

Quando utilizado o shell é instruído a mudar o diretório atual para a raiz do sistema de arquivos. Ou seja, irá para o diretório raiz do sistema, onde todos os outros diretórios e arquivos estão localizados.

Isso foi feito para que em seguida seja possível ver os diretórios existentes para isso deve usar o seguinte comando:

```
[root@ip-172-31-13-158 /]# ls  
bin boot dev etc home lib lib64 local media mnt opt proc root rum sbin srv sys tmp usr var
```

- `ls`: é usado para listar os arquivos e diretórios no diretório atual (ou em um diretório especificado, se fornecido como argumento).

Neste caso, irá listar o conteúdo presente no diretório atual pois não foi adicionado argumentos.

PASSO 1:

Configurar o NFS e criar um diretório dentro do filesystem do NFS com seu nome

LINUX

Chegou a hora de criar o diretório, siga o exemplo, mas utilize o seu nome.

```
[root@ip-172-31-13-158 /]# mkdir /talita
```

O comando “`mkdir /talita`” é usado para criar um diretório chamado “talita” na raiz do sistema de arquivos.

- `mkdir`: utilizado para criar diretórios, pastas, e significa "make directory" (criar diretório);
- `/talita`: é o argumento que especifica o caminho do diretório a ser criado. Neste caso, será criado um diretório na raiz do sistema (/) nomeado como “talita”.

Quando você executa “`mkdir /nome`”, o sistema cria um diretório chamado “nome” na raiz do sistema de arquivos.

Para conferir se ele foi criado corretamente você pode utilizar o comando “`ls`” e verificar se o mesmo será listado.

```
[root@ip-172-31-13-158 /]# ls  
bin boot dev etc home lib lib64 local media mnt opt proc root rum sbin srv sys talita tmp usr var
```

Podemos ver que o diretório denominado “`talita`” foi criado corretamente.

Para que seu diretório possa ser acessado deve configurá-lo segundo as próximas instruções.

PASSO 1:

Configurar o NFS e criar um diretório dentro do filesystem do NFS com seu nome

LINUX

```
[root@ip-172-31-13-158 ~]# chmod -R 777 /talita
```

O comando “`chmod -R 777 /talita`”, está instruindo o sistema a dar permissões de leitura, escrita e execução completas para todos os usuários e grupos em todos os arquivos e diretórios dentro do diretório “/talita” e em si mesmo.

- `chmod`: é utilizado para modificar as permissões de arquivos e diretórios;
 - `-R`: indica que o comando deve ser executado recursivamente, ou seja, aplicado a todos os arquivos e subdiretórios dentro do diretório especificado;
 - `777`: são os valores numéricos que representam as permissões. Cada número representa um conjunto de permissões para três grupos de usuários: proprietário, grupo e outros.
 - O primeiro dígito representa as permissões do proprietário, que são rwx (leitura, escrita e execução).
 - O segundo dígito representa as permissões do grupo, também rwx.
 - E o terceiro representa as permissões para outros usuários, seguindo a mesma lógica de leitura, escrita e execução.
- Neste caso, foi utilizado “777” que concede permissão completa para ambos;
- `/talita`: é o diretório especificado, o qual será aplicado as permissões.

Fique atento, pois em um ambiente de teste a liberação total de permissão para todos os grupos de usuários não irá gerar problemas, mas em um ambiente de produção as permissões devem ser restritas a alguns grupos dependendo da finalidade da aplicação.

PASSO 1:

Configurar o NFS e criar um diretório dentro do filesystem do NFS com seu nome

LINUX

```
[root@ip-172-31-13-158 ~]# nano /etc/exports
```

O comando “`nano /etc/exports`” é usado para editar o arquivo de configuração do NFS. O arquivo em questão, é onde se especifica quais diretórios ou sistemas de arquivos serão compartilhados via NFS e quais permissões são concedidas aos clientes NFS.

- `nano`: é um editor de texto no terminal, que será usado para abrir um arquivo especificado;
- `/etc/exports`: é o caminho do arquivo que será aberto com o nano.

Quando você executa este comando, o arquivo será aberto no editor, permitindo que você veja e edite seu conteúdo.

```
GNU nano 2.9.8          /etc/exports      Modified
/talita *(rw,sync,no_root_squash,no_subtree_check)
```

A linha “`/talita *(rw,sync,no_root_squash,no_subtree_check)`” é um exemplo de uma entrada no arquivo de configuração.

- `/talita`: é o caminho completo para o diretório que está sendo compartilhado via NFS;
- `*`: o asterisco indica que todos os clientes têm permissão para acessar esse compartilhamento, ou seja, permite que qualquer cliente NFS na rede acesse o diretório;
- `(rw)`: essa opção concede permissão de leitura e escrita aos clientes, podem ler e escrever arquivos no diretório compartilhado;
- `(sync)`: indica que as operações de gravação no compartilhamento devem ser sincronizadas, com isso o NFS aguardará até que os dados sejam gravados fisicamente antes de confirmar a gravação;

PASSO 1:

Configurar o NFS e criar um diretório dentro do filesystem do NFS com seu nome

LINUX

- **(no_root_squash)**: esta opção desativa o mecanismo de segurança chamado "root squashing". Ele mapeia o usuário root de um cliente para um usuário não privilegiado no servidor, reduzindo os privilégios do root. Desativá-lo permite que o root do cliente tenha privilégios completos no compartilhamento NFS;
- **(no_subtree_check)**: desativa a verificação do acesso dos clientes a subdiretórios dentro do diretório compartilhado. Desativar essa verificação pode melhorar o desempenho, mas requer cuidado ao definir permissões para evitar problemas de segurança.

Portanto, indica que o diretório está sendo compartilhado via NFS com permissões de leitura e escrita, sincronização ativada, root squashing desativado e verificação de subárvore desativada.

```
[root@ip-172-31-13-158 ~]# cat /etc/exports  
/talita *(rw,sync,no_root_squash,no_subtree_check)  
[root@ip-172-31-13-158 ~]#
```

O comando “`cat /etc/exports`” mostra o conteúdo do arquivo de configuração do NFS.

- `cat`: é um comando utilizado para concatenar e exibir o conteúdo de arquivos de texto;
- `/etc/exports`: este é o caminho completo do arquivo que será exibido.

Quando você executa este comando, o conteúdo do arquivo “`/etc/exports`” será exibido na tela do terminal. Permitindo que você veja as configurações atuais de compartilhamento NFS em seu sistema, incluindo os diretórios compartilhados, as permissões e as configurações específicas para cada compartilhamento.

Neste caso, este é o conteúdo exibido:

- `/talita *(rw,sync,no_root_squash,no_subtree_check)`

PASSO 1:

Configurar o NFS e criar um diretório dentro do filesystem do NFS com seu nome

LINUX

```
[root@ip-172-31-13-158 ~]# exportfs -a
```

O comando “`exportfs -a`” é usado para atualizar as configurações de exportação do NFS e aplicar quaisquer alterações feitas no arquivo de configuração “`/etc/exports`”.

- `exportfs`: é utilizado para gerenciar as exportações NFS;
- `-a`: é uma opção que significa “all” (todos), ou seja, todas as exportações definidas no arquivo de configuração serão aplicadas.

Quando você executa este comando o sistema verifica o arquivo de configuração do NFS em busca de qualquer nova configuração ou alteração nas configurações existentes. Além disso, atualiza o serviço NFS, permitindo que os compartilhamentos definidos estejam disponíveis para os clientes.

```
[root@ip-172-31-13-158 ~]# systemctl restart nfs-server
```

O comando “`systemctl restart nfs-server`” é usado para reiniciar o serviço do servidor NFS.

- `systemctl`: ferramenta de gerenciamento de serviços, usada para iniciar, parar, reiniciar e verificar o status de serviços do sistema;
- `restart`: é a opção usada junto ao `systemctl` para instruir o `systemd` a reiniciar o serviço especificado;
- `nfs-server`: é o nome do serviço NFS no `systemd`, o serviço que irá reiniciar.

De modo geral, ao executar este comando o `systemd` interrompe o serviço NFS em execução, em seguida inicia o mesmo aplicando as configurações e modificações feitas.

PASSO 1:

Configurar o NFS e criar um diretório dentro do filesystem do NFS com seu nome

LINUX

```
[root@ip-172-31-13-158 ~]# showmount -e  
Export list for ip-172-31-13-158.ec2.internal:  
/talita *
```

O comando “`showmount -e`” é usado para exibir a lista de diretórios compartilhados por um servidor NFS.

- `showmount`: é utilizado para consultar informações sobre os compartilhamentos NFS em um servidor;
- `-e`: é uma opção que significa "exports" (exportações). Ou seja, quando utilizada o sistema irá listar os diretórios exportados, compartilhados, pelo servidor NFS.

A saída do comando listará todos os diretórios ou recursos que estão sendo compartilhados pelo servidor e disponíveis para acesso por clientes NFS. Cada linha da saída mostrará o caminho completo do diretório compartilhado. Neste caso o diretório listado foi:

```
Export list for ip-172-31-13-158.ec2.internal:  
/talita *
```

- `ip-172-31-13-158.ec2.internal`: nome de host do servidor;
- `/talita`: o caminho do diretório que está sendo compartilhado;
- `*`: o asterisco indica que todos os clientes têm permissão para acessar esse compartilhamento.

De modo geral, ao executar o comando, será apresentado uma lista de diretórios que estão sendo compartilhados pelo servidor.

PASSO 2:

Subir um apache no servidor - o apache deve estar online e rodando

LINUX

```
[root@ip-172-31-13-158 ~]# yum install httpd -y
```

O comando “`yum install httpd -y`” é usado para instalar o servidor web Apache

- `yum`: é um utilitário de gerenciamento de pacotes;
- `install`: é a ação que instrui o YUM a fazer uma instalação;
- `httpd`: é o nome do pacote a ser instalado, ele contém o servidor web Apache;
- `-y`: essa opção diz ao YUM para assumir automaticamente "yes" (sim) como resposta a todas as perguntas de confirmação que possam surgir durante a instalação.

Ao executar este comando, o YUM verifica os repositórios de software configurados em seu sistema, procura o pacote “`httpd`”, resolve quaisquer dependências necessárias e procede com a instalação do servidor web Apache.

```
[root@ip-172-31-13-158 ~]# systemctl start httpd
```

O comando “`systemctl start httpd`” inicia o Apache.

```
[root@ip-172-31-13-158 ~]# systemctl enable httpd
Create symlink /etc/systemd/system/multi-user.target.wants/httpd.service to /usr/lib/systemd/system/httpd.service.
[root@ip-172-31-13-158 ~]#
```

O comando “`systemctl enable httpd`” é usado para habilitar inicialização automática do Apache durante a inicialização do sistema.

PASSO 2:

Subir um apache no servidor - o apache deve estar online e rodando

LINUX

```
[root@ip-172-31-13-158 ~]# systemctl status httpd
```

O comando “systemctl status httpd” é usado para verificar o status do serviço Apache no sistema. Ele fornece informações sobre se o serviço está em execução, parado ou com problemas.

- **systemctl**: ferramenta de gerenciamento usada para iniciar, parar, reiniciar e verificar o status de serviços do sistema;
- **status**: é a opção usada junto ao “systemctl” para obter informações detalhadas sobre o status de um serviço específico;
- **httpd**: é o nome do serviço especificado, o Apache.

Ao executar este comando o sistema exibirá informações detalhadas sobre o status do serviço Apache, como se está em execução ou não, quando foi iniciado pela última vez, mensagens de log recentes e outros.

```
[root@ip-172-31-13-158 ~]# systemctl status httpd
● httpd.service - The Apache HTTP Server
  Loaded: loaded (/usr/lib/systemd/system/httpd.service; enable; vendor preset: disabled)
  Active: active (running) since Tue 2023-08-29 16:23:55 UTC; 1min 1s ago
    Docs: man:httpd.service(8)
   Main PID: 28925 (httpd)
     Status: "Total requests: 0; Idle/Busy workers 100/0;Requests/sec: 0; Bytes served/sec: 0 B/sec"
   CGroup: /system.slice/httpd.service
           |-28925 /usr/sbin/httpd -DFOREGROUND
           |-28926 /usr/sbin/httpd -DFOREGROUND
           |-28928 /usr/sbin/httpd -DFOREGROUND
           |-28939 /usr/sbin/httpd -DFOREGROUND
           |-28950 /usr/sbin/httpd -DFOREGROUND
           |-28956 /usr/sbin/httpd -DFOREGROUND

Aug 29 16:23:55 ip-172-31-13-158.ec2.internal systemd[1]: Starting The Apache HTTP Server...
Aug 29 16:23:55 ip-172-31-13-158.ec2.internal systemd[1]: Started The Apache HTTP Server.
root@ip-172-31-13-158 ~#
```

PASSO 2:

Subir um apache no servidor - o apache deve estar online e rodando

Neste caso a mensagem apresenta que o serviço Apache está em execução, funcionando corretamente. Além disso, é possível notar que o sistema está em um fuso horário diferente do local no qual os testes foram aplicados, para corrigir isso e facilitar os testes dos próximos passos você pode seguir os exemplos abaixo:

```
[root@ip-172-31-13-158 ~]# date  
Tue Aug 29 16:26:07 UTC 2023  
[root@ip-172-31-13-158 ~]#
```

O comando “`date`” é usado para exibir a data e a hora atuais em sistemas. Executar este comando sem argumentos irá mostrar a data e a hora atuais no fuso horário UTC.

```
[root@ip-172-31-13-158 ~]# timedatectl set-timezone America/Sao_Paulo
```

O comando “`timedatectl set-timezone America/Sao_Paulo`” é usado para definir o fuso horário do sistema para o horário de São Paulo, Brasil.

- `timedatectl`: é usado para realizar várias configurações relacionadas ao tempo e à data em sistemas que utilizam o gerenciamento de tempo do `systemd`;
- `set-timezone`: é a opção que permite definir o fuso horário do sistema;
- `America/Sao_Paulo`: é o nome da zona de fuso horário escolhida, ou seja, o horário de São Paulo, Brasil.

Ao executar este comando você ajusta o fuso horário do sistema, garantindo que o relógio do mesmo exiba a hora correta para a região escolhida, neste caso a região de São Paulo, Brasil.

```
[root@ip-172-31-13-158 ~]# date  
Tue Aug 29 13:28:16 -03 2023  
[root@ip-172-31-13-158 ~]#
```

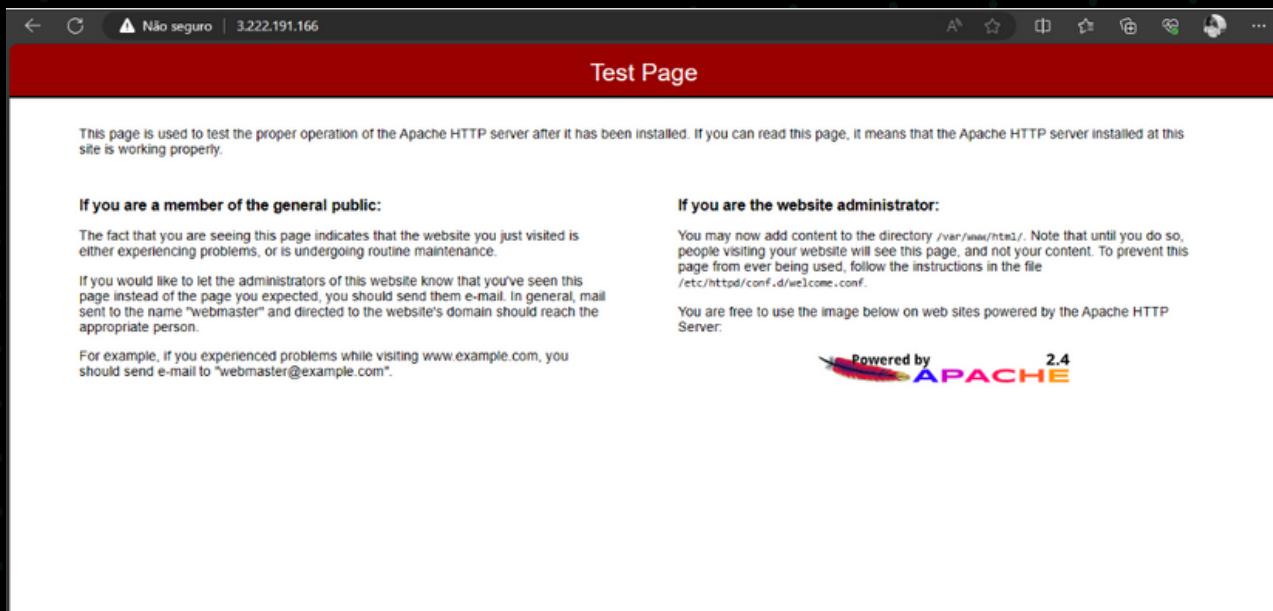
Ao executar o comando “`date`” novamente é possível notar que a data e hora estão corretas e condizentes com o fuso horário local.

PASSO 2:

Subir um apache no servidor - o apache deve estar online e rodando

LINUX

Outro teste que é possível ser feito para averiguar se o sistema Apache está de fato funcionando é digitando o endereço ip em um navegador web, se estiver funcionando e habilitado deve aparecer a seguinte página:



PASSO 3:

Criar um script que valide se o serviço esta online e envie o resultado da validação para o seu diretório no nfs

```
[root@ip-172-31-13-158 ~]# nano /opt/apache.sh
```

O comando “`nano /opt/apache.sh`” é usado para abrir o editor de texto nano e editar ou criar um arquivo chamado ‘apache.sh’ localizado no diretório ‘/opt’.

- `nano`: é uma ferramenta de edição de texto;
- `/opt/apache.sh`: é o caminho completo para o arquivo que será editado.

Neste caso, o comando foi utilizado para criar o arquivo dentro do diretório especificado e abrir o mesmo para edição.

```
GNU nano 2.9.8           /opt/apache.sh          Modified

#!/bin/bash
timestamp=$(date '+%Y-%m-%d %H:%M:%S')
service_name="Apache_Talita"
output_directory="/talita"

if systemctl is-active --quiet httpd; then
    status="ONLINE"
    message="O serviço está funcionando corretamente (online)."
    output_file="$output_directory/online_status.txt"
else
    status="OFFLINE"
    message="O serviço não está em execução (offline)."
    output_file="$output_directory/offline_status.txt"
fi

echo "$timestamp - $service_name -$status -$message" > "$output_file"
```

PASSO 3:

Criar um script que valide se o serviço esta online e envie o resultado da validação para o seu diretório no nfs

LINUX

Este script tem o objetivo de verificar o status do serviço Apache (httpd) e registrar um relatório com informações sobre seu status.

- `#!/bin/bash`

Esta linha é chamada de shebang e indica que o script deve ser executado usando o interpretador Bash.

- `timestamp=$(date '+%Y-%m-%d %H:%M:%S')`

Esta linha cria uma variável chamada "timestamp" que armazena a data e hora atuais no formato "AAAA-MM-DD HH:MM:SS". Ela usa o comando "date" para obter a data e hora atuais.

- `service_name="Apache_Talita"`

Aqui, está sendo definido uma variável chamada "service_name" que armazena o nome do serviço, que é "Apache_Talita" neste caso.

- `output_directory="/talita"`

Essa linha define a variável "output_directory", que especifica o diretório onde os relatórios de status serão armazenados. No exemplo, o diretório é "/talita".

- `if systemctl is-active --quiet httpd; then`

Esta linha verifica se o serviço Apache está ativo usando o comando "systemctl". Se o serviço estiver ativo, o script executa o bloco de código após o "then".

- `status="ONLINE"`

- `message="O serviço está funcionando corretamente (online)."`

- `output_file="$output_directory/online_status.txt"`

Se o serviço Apache estiver ativo, as variáveis "status", "message" e "output_file" são definidas para refletir que o serviço está online e um arquivo chamado "online_status.txt" será criado no diretório especificado.

PASSO 3:

Criar um script que valide se o serviço esta online e envie o resultado da validação para o seu diretório no nfs

LINUX

- `else`

Se o serviço Apache não estiver ativo, o script executa o bloco de código após o "else".

- `status="OFFLINE"`
- `message="O serviço não está em execução (offline)."`
- `output_file="$output_directory/offline_status.txt"`

Neste caso, as variáveis "status", "message" e "output_file" são definidas para indicar que o serviço está offline, e um arquivo chamado "offline_status.txt" será criado no diretório especificado.

- `fi`

Essa linha fecha a estrutura condicional "if-else".

- `echo "$timestamp - $service_name - $status - $message" > "$output_file"`

Por fim, esta linha cria uma linha de texto que contém o timestamp, o nome do serviço, o status e a mensagem. Essa linha de texto é redirecionada para o arquivo especificado em `output_file`, que pode ser "online_status.txt" ou "offline_status.txt", dependendo do status do serviço.

Em resumo, este script verifica o status do serviço Apache e registra informações sobre o status em um arquivo de saída, dependendo se o serviço está online ou offline. Os relatórios são armazenados em um diretório específico compartilhado pelo NFS.

PASSO 3:

Criar um script que valide se o serviço esta online e envie o resultado da validação para o seu diretório no nfs

LINUX

```
[root@ip-172-31-13-158 ~]# cat /opt/apache.sh
#!/bin/bash
timestamp=$(date '+%Y-%m-%d %H:%M:%S')
service_name="Apache_Talita"
output_directory="/talita"

if systemctl is-active --quiet httpd; then
    status="ONLINE"
    message="O servio est'a funcionando corretamente (online)."
    output_file="$output_directory/online_status.txt"
else
    status="OFFLINE"
    message="O servio n~ao est'a em execu~ao (offline)."
    output_file="$output_directory/offline_status.txt"
fi

echo "$timestamp - $service_name -$status -$message" > "$output_file"

[root@ip-172-31-13-158 ~]#
```

O comando “`cat /opt/apache.sh`” é usado para exibir o conteúdo do arquivo chamado “`apache.sh`” que está localizado no diretório “`/opt`”, criado e editado anteriormente.

- `cat`: é usado para concatenar e exibir o conteúdo de arquivos de texto;
- `/opt/apache.sh`: é o caminho completo para o arquivo que será exibido.

Ao executar este comando, o sistema exibe o conteúdo completo do arquivo no seu terminal, permitindo que você veja todos os comandos e textos contidos no arquivo.

PASSO 4:

Preparar a execução automatizada do script a cada 5 minutos

LINUX

```
[root@ip-172-31-13-158 ~]# chmod +x /opt/apache.sh
```

O comando “`chmod +x /opt/apache.sh`” é usado para conceder permissões de execução ao arquivo que contém o script. O mesmo se tornará executável, permitindo que o execute como um programa ou script.

- `chmod`: é utilizado para alterar as permissões de um arquivo ou diretório;
- `+x`: é a opção usada para adicionar permissão de execução ao arquivo. O símbolo “+” indica adição de permissão, e “x” indica permissão de execução;
- `opt/apache.sh`: é o caminho para o arquivo ao qual será adicionado permissão de execução.

```
[root@ip-172-31-13-158 ~]# crontab -e
```

O comando “`crontab -e`” é usado para editar as configurações do cron de um usuário no sistema Linux. O cron é um serviço que permite agendar tarefas para serem executadas automaticamente em horários específicos, datas ou intervalos regulares.

- `crontab`: é o comando usado para interagir com o cron;
- `-e`: é uma opção que indica ao “`crontab`” que você deseja editar ou criar o arquivo de cron do usuário atual.

Ao executar este comando o sistema abre o arquivo de cron do usuário no editor de texto padrão, neste caso o “`vi`”. Este arquivo é onde você pode especificar as tarefas que deseja agendar.

PASSO 4:

Preparar a execução
automatizada do script a cada
5 minutos

LINUX

```
*/5 * * * * /opt/apache.sh
~
~
~
~
~
~
~
:wq!
```

A linha `*/5 * * * * /opt/apache.sh` é uma entrada de cron que especifica uma tarefa a ser executada automaticamente pelo cron a cada 5 minutos.

- Os cinco campos separados por espaços representam:
 1. Minuto (0-59)
 2. Hora (0-23)
 3. Dia do mês (1-31)
 4. Mês (1-12)
 5. Dia da semana (0-7, onde 0 e 7 representam domingo, 1 segunda-feira, 2 terça-feira e assim por diante).
- O asterisco (*) em cada um dos cinco campos significa "todos os valores possíveis". Portanto, `*/5` no campo dos minutos indica "a cada 5 minutos".
- `/opt/apache.sh`: é o caminho completo para o comando ou script que será executado quando a tarefa de cron for acionada. Neste caso, o script "apache.sh" está localizado no diretório "/opt".

Portanto, esta linha diz ao cron para executar o script `"/opt/apache.sh"` a cada 5 minutos, independentemente da hora, do dia do mês, do mês ou do dia da semana.

PASSO 4:

Preparar a execução
automatizada do script a cada
5 minutos

LINUX

```
[root@ip-172-31-13-158 ~]# crontab -e
no crontab for root - using an empty one
crontab: installing new crontab
[root@ip-172-31-13-158 ~]#
```

Essa saída informa que está começando a configurar o cron para o usuário root, e pode adicionar as entradas de cron desejadas para agendar tarefas específicas para esse usuário. Qualquer entrada adicionada será executada com os privilégios do root, o que permite realizar tarefas administrativas no sistema.

- **no crontab for root - using an empty one**: esta parte da saída informa que o usuário root não possui um arquivo de cron existente. "Using an empty one" indica que um novo arquivo de cron vazio será criado para o usuário root;
- **crontab: installing new crontab**: esta linha indica que o novo arquivo de cron foi criado e está sendo instalado.

TESTES:

Testes dos passos 3 e 4

LINUX

Vá até o diretório compartilhado, onde o script direcionaria os arquivos.

```
[root@ip-172-31-13-158 ~]# cd /talita/  
[root@ip-172-31-13-158 talita]#
```

Liste os arquivos, é possível notar que o arquivo que se refere ao status online está criado

```
[root@ip-172-31-13-158 talita]# ls  
online_status.txt
```

De um cat neste arquivo para ver seu conteúdo de texto, neste caso podemos notar que o texto foi armazenado corretamente às 13 e 45.

```
[root@ip-172-31-13-158 talita]# cat online_status.txt  
2023-08-29 13:45:01 - Apache_Talita - Status: ONLINE - O serviço está funcionando corretamente  
(online).
```

Você pode inserir algumas linhas no script para apresentar quando foram as execuções do mesmo. Neste caso foi utilizado:

- `log_file="/var/log/apache.log"`: cria uma variável é passado o endereço do arquivo a ser armazenado os dados.
- `echo "$(date '+%Y-%m-%d %H:%M:%S') - Início do Script" >> "$log_file"`: armazena a data e o texto "Início do Script" no local indicado.
- `echo "$(date '+%Y-%m-%d %H:%M:%S') - Fim do Script" >> "$log_file"`: armazena a data e o texto "Fim do Script" no local indicado.

Veja o script atualizado:

TESTES:

Testes dos passos 3 e 4

LINUX

GNU nano 2.9.8

/opt/apache.sh

Modified

```
#!/bin/bash
timestamp=$(date '+%Y-%m-%d %H:%M:%S')
service_name="Apache_Talita"
output_directory="/talita"

log_file="/var/log/apache.log"

echo "$(date '+%Y-%m-%d %H:%M:%S') - Início do Script" >> "$log_file"

if systemctl is-active --quiet httpd; then
    status="ONLINE"
    message="O serviço está funcionando corretamente (online)."
    output_file="$output_directory/online_status.txt"
else
    status="OFFLINE"
    message="O serviço não está em execução (offline)."
    output_file="$output_directory/offline_status.txt"
fi

echo "$timestamp - $service_name -$status -$message" > "$output_file"

echo "$(date '+%Y-%m-%d %H:%M:%S') - Fim do Script" >> "$log_file"
```

Para verificar se ele está armazenando corretamente os dados você pode dar um cat no arquivo, assim como mostrado no exemplo:

```
[root@ip-172-31-13-158 talita]# cat /var/log/apache.log
2023-08-29 14:15:01 - Início do Script
2023-08-29 14:15:01 - Fim do Script
[root@ip-172-31-13-158 talita]#
```

Para testar a parte do script que indica que o serviço está offline você pode desligá-lo utilizando o seguinte comando:

```
[root@ip-172-31-13-158 talita]# systemctl stop httpd
[root@ip-172-31-13-158 talita]#
```

TESTES:

Testes dos passos 3 e 4

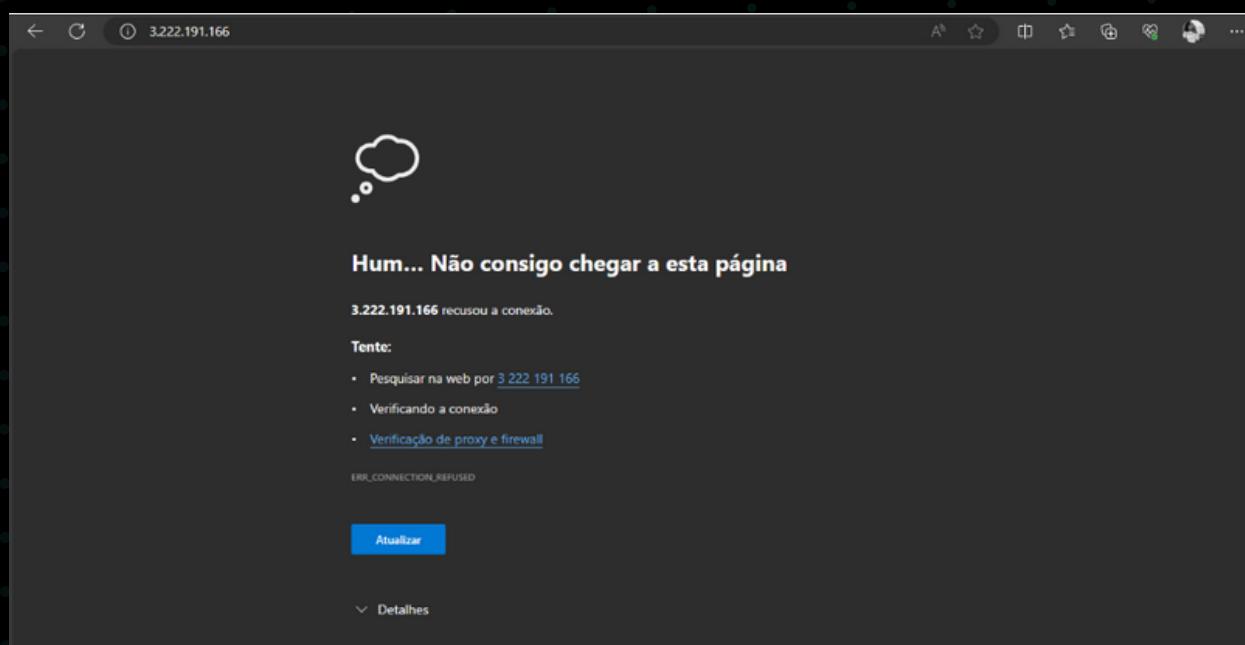
LINUX

Você pode verificar conferindo o status do serviço Apache (httpd), neste caso ele está inativo.

```
[root@ip-172-31-13-158 talita]# systemctl status httpd
● httpd.service - The Apache HTTP Server
  Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; vendor preset: disabled)
  Active: inactive (dead) since Tue 2023-08-29 14:21:47 -03; 14s ago
    Docs: man:httpd.service(8)
   Process: 28925 ExecStart=/usr/sbin/httpd $OPTIONS -DFOREGROUND (code=exited, status=0/SUCCESS)
   Main PID: 28925 (code=exited, status=0/SUCCESS)
     Status: "Total requests:13; Idle/Busy workers 100/0; Rewuests/sec: 0.00376; Bytes served/sec: 14 B/sec"

Aug 29 13:23:55 ip-172-31-13-158.ec2.internal systemd[1]: Starting The Apache HTTP Server...
Aug 29 13:23:55 ip-172-31-13-158.ec2.internal systemd[1]: Started The Apache HTTP Server.
Aug 29 14:21:46 ip-172-31-13-158.ec2.internal systemd[1]: Stopping The Apache HTTP Server...
Aug 29 14:21:47 ip-172-31-13-158.ec2.internal systemd[1]: Stopped The Apache HTTP Server.
[root@ip-172-31-13-158 talita]#
```

Também é possível verificar em um navegador digitando o endereço ip, o mesmo deve apresentar um erro ao tentar se conectar.



TESTES:

Testes dos passos 3 e 4

LINUX

Ao listar os arquivos do diretório é possível notar que arquivo que armazena a mensagem de offline foi criado.

```
[root@ip-172-31-13-158 talita]# ls  
offline_status.txt online_status.txt
```

Ao exibir o conteúdo do arquivo “[/var/log/apache.log](#)” podemos ver os horários em que o script foi executado e que o mesmo está sendo executado de 5 em 5 minutos.

```
[root@ip-172-31-13-158 talita]# cat /var/log/apache.log  
2023-08-29 14:15:01 - Início do Script  
2023-08-29 14:15:01 - Fim do Script  
2023-08-29 14:20:01 - Início do Script  
2023-08-29 14:20:01 - Fim do Script  
2023-08-29 14:25:01 - Início do Script  
2023-08-29 14:25:01 - Fim do Script  
[root@ip-172-31-13-158 talita]#
```

Ao exibir o conteúdo do arquivo “[offline_status.txt](#)” e comparar com a exibição anterior, vemos que a última execução do arquivo apresentou o status de offline.

```
[root@ip-172-31-13-158 talita]# cat offline_status.txt  
2023-08-29 14:25:01 - Apache_Talita - Status: OFFLINE - o serviço não está em execução (offline).
```

Já ao exibir o conteúdo do arquivo “[online_status.txt](#)” e fazer as comparações, vemos que na execução das 14:20 foi o último registro que o sistema estava online.

```
[root@ip-172-31-13-158 talita]# cat online_status.txt  
2023-08-29 14:20:01 - Apache_Talita - Status: ONLINE - o serviço está funcionando corretamente  
(online).
```

TESTES:

Testes dos passos 3 e 4

LINUX

Você pode religar o Apache e repetir os testes seguindo o exemplo abaixo:

```
[root@ip-172-31-13-158 talita]# systemctl start httpd  
[root@ip-172-31-13-158 talita]#
```

Verificação de status:

```
[root@ip-172-31-13-158/]# systemctl status httpd  
● httpd.service - The Apache HTTP Server  
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; enable; vendor preset: disabled)  
   Active: active (running) since Tue 2023-08-29 14:26:46 -03; 10s ago  
     Docs: man:httpd.service(8)  
   Main PID: 29418 (httpd)  
     Status: "Total requests: 0; Idle/Busy workers 100/0; Requests/sec: 0; Bytes served/sec: 0 B/sec"  
   CGroup: /system.slice/httpd.service  
           |-29418 /usr/sbin/httpd -DFOREGROUND  
           |-29419 /usr/sbin/httpd -DFOREGROUND  
           |-29421 /usr/sbin/httpd -DFOREGROUND  
           |-29430 /usr/sbin/httpd -DFOREGROUND  
           |-29443 /usr/sbin/httpd -DFOREGROUND  
           |-29449 /usr/sbin/httpd -DFOREGROUND  
  
Aug 29 14:26:46 ip-172-31-13-158.ec2.internal systemd[1]: Starting The Apache HTTP Server...  
Aug 29 14:26:46 ip-172-31-13-158.ec2.internal systemd[1]: Started The Apache HTTP Server.  
root@ip-172-31-13-158/]#
```

Exibição o conteúdo nos arquivos de texto “[/var/log/apache.log](#)”, “[offline_status.txt](#)” e “[online_status.txt](#)”:

```
[root@ip-172-31-13-158 talita]# cat /var/log/apache.log  
2023-08-29 14:15:01 - Início do Script  
2023-08-29 14:15:01 - Fim do Script  
2023-08-29 14:20:01 - Início do Script  
2023-08-29 14:20:01 - Fim do Script  
2023-08-29 14:25:01 - Início do Script  
2023-08-29 14:25:01 - Fim do Script  
2023-08-29 14:30:01 - Início do Script  
2023-08-29 14:30:01 - Fim do Script  
[root@ip-172-31-13-158 talita]#
```

TESTES:

Testes dos passos 3 e 4

LINUX

```
[root@ip-172-31-13-158 talita]# cat offline_status.txt  
2023-08-29 14:25:01 - Apache_Talita - Status: OFFLINE - o serviço não está em execução (offline).
```

```
[root@ip-172-31-13-158 talita]# cat online_status.txt  
2023-08-29 14:30:01 - Apache_Talita - Status: ONLINE - o serviço está funcionando corretamente  
(online).
```

Com isso é possível perceber que a execução dos passos 3 e 4 está funcionando corretamente como deveria.

TESTES:

Testes como Cliente NFS

LINUX

Para executar testes de cliente NFS, é possível criar uma nova instância semelhante à criada para o servidor, nesse caso a única diferença é que **não** foi atribuído um ip elástico e possui apenas 8GB. Após a criação e inicialização da instância que será usada como cliente nos testes, você deve acessá-la via ssh e seguir os próximos passos.

```
[ec2-user@ip-172-31-9-71-]$ sudo su -  
[root@ip-172-31-9-71 ~]#
```

Como visto anteriormente este comando permite abrir uma nova sessão de terminal como usuário root, com permissões administrativas.

É necessário que o NFS esteja instalado na máquina “cliente”, você pode usar o comando a seguir para conferir se ele de fato está instalado por padrão na instância utilizada.

- **rpm -q nfs-utils**
- **rpm**: é o comando usado para trabalhar com pacotes RPM no sistema Linux;
- **-q**: a opção "q" é usada para consultar informações sobre um pacote RPM. Neste contexto, estamos consultando informações sobre o pacote "nfs-utils";
- **nfs-utils**: é o nome do pacote que está sendo verificado.

A saída deste comando pode ser uma das seguintes:

- Se o pacote "nfs-utils" estiver instalado, o comando retornará o nome do pacote e sua versão.
- Se o pacote "nfs-utils" não estiver instalado, o comando não retornará nada.

```
[root@ip-172-31-9-71 ~]# rpm -q nfs-utils  
nfs-utils-1.3.0-0.54.amzn2.0.2.x86_64  
[root@ip-172-31-9-71 ~]#
```

Como mostrado acima, podemos ver que o pacote estava instalado por padrão na instância “cliente”.

TESTES:

Testes como Cliente NFS

LINUX

É criado o diretório “/cliente”, onde será feito a montagem do sistema de arquivos compartilhados com o uso do NFS

```
[root@ip-172-31-9-71 ~]# cd /
[root@ip-172-31-9-71 /]# ls
bin boot dev etc home lib lib64 local media mnt opt proc root rum sbin srv sys tmp usr var
[root@ip-172-31-9-71 /]#
```

```
[root@ip-172-31-9-71 /]# mkdir cliente
```

```
[root@ip-172-31-9-71 /]# ls
bin boot cliente dev etc home lib lib64 local media mnt opt proc root rum sbin srv sys tmp usr var
[root@ip-172-31-9-71 /]#
```

O comando “`mount -t nfs ip-172-31-13-158.ec2.internal:/talita /cliente`” é usado para montar um sistema de arquivos NFS.

- `mount`: é usado para montar sistemas de arquivos em diretórios específicos no sistema operacional Linux;
- `-t nfs`: esta opção especifica o tipo de sistema de arquivos a ser montado, que é NFS neste caso. O sistema de arquivos NFS é usado para montar compartilhamentos de rede exportados por um servidor NFS;
- `ip-172-31-13-158.ec2.internal:/talita`: este é o endereço do servidor NFS e o caminho do compartilhamento a ser montado;
- `/cliente`: este é o diretório no sistema de arquivos local onde o compartilhamento NFS será montado.

Portanto, ao executar este comando você está instruindo o sistema operacional a montar o compartilhamento NFS do servidor "ip-172-31-13-158.ec2.internal" com o caminho "/talita" no diretório local "/cliente". Isso permitirá que você acesse os arquivos e diretórios do compartilhamento NFS.

```
[root@ip-172-31-9-71 /]# mount -t nfs ip-172-31-13-158.ec2.internal:/talita /cliente
[root@ip-172-31-9-71 /]#
```

TESTES:

Testes como Cliente NFS

LINUX

Para verificar se a montagem ocorreu corretamente você pode executar o comando “mount”, sem argumentos e observar se irá aparecer a montagem criada, como no exemplo a seguir:

```
[root@ip-172-31-9-71 ]# mount  
[...]  
op-172-31-158.ec2.internal:/talita on /cliente type nfs4 (rw,realatime,vers=4.1,rsize=262144,wszie=262  
144,namlen=255,hard,proto=tcp,timeo=600,retrans=2,sec=sys,clientaddr=172.31.9.71,local_lock=none  
,addr=172.31.13.158)  
[root@ip-172-31-9-71 ]#
```

Vá até o diretório e verifique se os arquivos compartilhados estão lá, se estiverem o processo foi feito corretamente e tudo estará funcionando.

```
[root@ip-172-31-9-71 ]# ls  
bin boot cliente dev etc home lib lib64 local media mnt opt proc root rum sbin srv sys tmp usr var  
[root@ip-172-31-9-71 ]#
```

```
[root@ip-172-31-9-71 ]# cd /cliente/  
[root@ip-172-31-9-71 cliente]#
```

```
[root@ip-172-31-9-71 cliente]# ls  
offline_status.txt online_status.txt
```

TESTES:

Testes como Cliente NFS

LINUX

Exiba o conteúdo dos arquivos e compare com os mesmos do servidor.

Cliente:

```
[root@ip-172-31-9-71 cliente]# cat offline_status.txt  
2023-08-29 14:25:01 - Apache_Talita - Status: OFFLINE - o serviço não está em execução (offline).
```

```
[root@ip-172-31-9-71 cliente]# cat online_status.txt  
2023-08-29 15:15:01 - Apache_Talita - Status: ONLINE - o serviço está funcionando corretamente  
(online).
```

Servidor:

```
[root@ip-172-31-13-158 talita]# cat offline_status.txt  
2023-08-29 14:25:01 - Apache_Talita - Status: OFFLINE - o serviço não está em execução (offline).
```

```
[root@ip-172-31-13-158 talita]# cat online_status.txt  
2023-08-29 15:15:01 - Apache_Talita - Status: ONLINE - o serviço está funcionando corretamente  
(online).
```

Note que o conteúdo coincide, ou seja, é igual.

Para que esta montagem não precise ser feita quando reiniciar a instância "cliente", você pode seguir os passos do próximo exemplo.

```
[root@ip-172-31-9-71 cliente]# nano /etc/fstab
```

O comando "[nano /etc/fstab](#)" é usado para editar o arquivo de configuração que lista os sistemas de arquivos e as opções de montagem que o sistema operacional deve usar durante a inicialização.

- O arquivo "[/etc/fstab](#)" é usado para definir configurações de montagem de sistemas de arquivos, como partições locais, compartilhamentos NFS, sistemas de arquivos swap e muito mais.

TESTES:

Testes como Cliente NFS

LINUX

```
GNU nano 2.9.8          /etc/fstab           Modified  
#  
UUID=9518dcb4=5eb4-4e05-9e86-786668d21b80   /   xfs defaults,noatime 1 1  
ip-172-32-13-158.ec2.internal:/talita /cliente nfs defaults 0 0
```

A linha “`ip-172-32-13-158.ec2.internal:/talita /cliente nfs defaults 0 0`” define como um compartilhamento NFS será montado automaticamente durante a inicialização do sistema.

- `ip-172-32-13-158.ec2.internal:/talita`: é o servidor NFS e o caminho do compartilhamento NFS que será montado;
- `/cliente`: é o diretório local no sistema de arquivos onde o compartilhamento NFS será montado;
- `nfs`: este campo especifica o tipo de sistema de arquivos, que é NFS neste caso;
- `defaults`: é uma lista de opções de montagem para o compartilhamento NFS. esta opção geralmente inclui configurações padrão, como permissões de leitura e gravação (rw), montagem automática durante a inicialização (auto), permissões para o usuário que monta o compartilhamento (user), entre outras;
- `0` e `0`: esses valores são usados para determinar a ordem de verificação do sistema de arquivos durante a inicialização e a realização de verificações periódicas de integridade do sistema de arquivos. No contexto de compartilhamentos NFS, eles geralmente são definidos como 0, o que significa que não ocorrerão verificações específicas durante a inicialização.

Para testar isto, você pode reiniciar a instância, acessá-la via ssh e realizar os próximos testes exemplificados abaixo.

TESTES:

Testes como Cliente NFS

LINUX

Vá até o diretório clientes

```
[root@ip-172-31-9-71 ~]# cd /cliente/  
[root@ip-172-31-9-71 cliente]#
```

Liste os arquivos

```
[root@ip-172-31-9-71 cliente]# ls  
offline_status.txt  online_status.txt
```

Note que a montagem foi feita automaticamente ao iniciar a instância.

Crie um diretório chamado **teste** com a instância “cliente” e verifique se ele aparece para ambas as máquinas.

```
[root@ip-172-31-9-71 cliente]# mkdir teste
```

Cliente:

```
[root@ip-172-31-9-71 cliente]# ls  
offline_status.txt  online_status.txt  teste
```

Servidor:

```
[root@ip-172-31-13-158 talita]# ls  
offline_status.txt  online_status.txt  teste
```

Com estes testes é possível notar que o sistema NFS configurado no passo 1, está funcionando corretamente como deveria.

Documentação

TALITA VARGAS DE SOUZA