

**ATIVIDADE REFERENTE A CONHECIMENTOS GERAIS
ADQUIRIDOS NOS CURSOS DE AWS E LINUX NA TRILHA DE
APRENDIZADO DO ESTÁGIO**

AWS & DEVSECOPS COMPASS UOL

Bolsista: Igor Henrique Buzatto

FREDERICO WESTPHALEN – RS

1 Atividade referente a parte prática sobre AWS

- **Gerar uma chave pública para acesso ao ambiente.**

A criação da chave pública se dá no momento da criação da instancia EC2, no meu caso criei uma chave RSA para acesso ao SSH denominada AcessoSSH.pem.

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

Neste passo efetuei a criação da instancia EC2 com os requisitos e tags citados. Também é necessária a criação de uma subnet publica, um Internet Gateway e uma tabela de rotas para que a instancia consiga se comunicar com a internet.

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

No menu “Elastic IP’s”, na opção “Allocate Elastic IP address”, um endereço IPV4 público é alocado, após a alocação é necessário assimilar ele a uma instância, que no caso foi a instancia criada anteriormente.

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

Nas configurações da instancia ec2, no menu security, encontramos o security group criado automaticamente durante a criação da instancia. Em “inbound rules”, clicamos em “edit inbound rules” e adicionamos todas as regras requeridas. As regras solicitadas deverão ficar como na figura, então clicamos em “Save rules”.

ID da regra do grupo de segurança	Tipo informações	Protocolo	Intervalo de portas	Origem informações	Descrição - opcional informações
sg-014c3a81e9618c77	SSH	TCP	22	Personalizado	0.0.0.0/0
sg-0e364c0f970d0e206	UDP personalizado	UDP	111	Personalizado	0.0.0.0/0
sg-0d011fed8ebd702d	HTTPS	TCP	443	Personalizado	0.0.0.0/0
sg-0ea3d9f1b110f2a61	UDP personalizado	UDP	2049	Personalizado	0.0.0.0/0
sg-0f0ea58391ca5d7c5	HTTP	TCP	80	Personalizado	0.0.0.0/0
sg-0972d50f47e21807b	TCP personalizado	TCP	111	Personalizado	0.0.0.0/0
sg-0b470d844e850d157	NFS	TCP	2049	Personalizado	0.0.0.0/0

Figura 01: Regras adicionadas ao grupo de segurança. Fonte: Captura de tela do próprio autor

2 - ATIVIDADE REFERENTE A PARTE PRÁTICA SOBRE LINUX

Requisitos:

- 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

Para iniciar fiz o download da ISO do Oracle Linux no site oficial da marca, na sessão de downloads. Outro software utilizado foi o Oracle VM virtual Box, da mesma empresa do sistema operacional. Esse software foi apenas para a instalação da ISO em uma máquina virtual.

Após a instalação do software de virtualização, deve se adicionar uma nova máquina virtual inserindo as configurações desejadas, além de inserir também a ISO baixada anteriormente. Ao iniciar a máquina virtual, selecionei a opção Instalar Oracle Linux. O sistema vai verificar os arquivos e hardware, depois vai iniciar a interface de configuração.

Nas telas de configuração existem configurações simples e complexas, são elas:

- **Simples:** Idioma, tipo de teclado, data e hora, senha do root, criação de usuário e senha.
- **Complexas:** Seleção de disco para a instalação, conexão de rede e configuração de repositórios que contém os arquivos de instalação.
Na configuração de repositórios devem ser adicionados os seguintes links de repositórios da própria Oracle: (https://yum.oracle.com/repo/OracleLinux/OL8/baseos/latest/x86_64/;
https://yum.oracle.com/repo/OracleLinux/OL8/appstream/x86_64/;
https://yum.oracle.com/repo/OracleLinux/OL8/UEKR6/x86_64/ ;)

A senha de root utilizada foi Adm@123, o usuário criado foi Igor, também com a senha Adm@123. Vale lembrar que estou utilizando este ambiente apenas para testes, por isso utilizei senhas simples. Em um ambiente de produção as senhas devem ser complexas, cumprindo diversos requisitos, além de a senha de root nunca ser igual a de nenhum usuário. Seguindo estas recomendações a segurança do ambiente será muito maior.

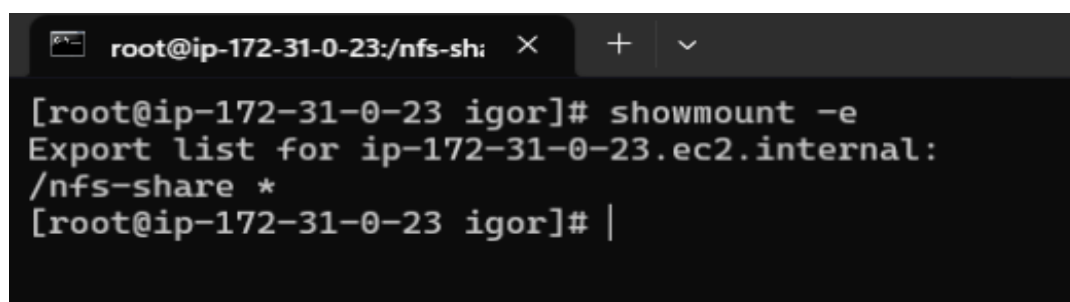
2.2 – CONFIGURAÇÃO DO NFS

Para a instalação do serviço nfs no Linux devemos primeiramente executar o comando **“sudo yum install nfs-utils”**. Após a instalação devemos criar uma pasta que irá conter os arquivos compartilhados no nfs. Nas instancias amazona Linux 2 o serviço nfs já vem instalado nativamente.

Usando o comando **“mkdir /nfs-share”**, a pasta nfs-share é criada na raiz /. As permissões dessa pasta variam de ambiente para ambiente, como neste caso o ambiente é de testes, usando o comando **“chmod -R 777 /nfs-share”**, são dadas todas as permissões de escrita, leitura e execução a todos os usuários, programas e grupos.

Agora para configurar a pasta para que ela seja compartilhada e funcione no nfs, é necessário alterar o seguinte arquivo. Usando o editor vim acessamos o arquivo **“vim /etc/exports”** e adicionamos a seguinte linha nele: **“/nfs-share *(rw,sync,no_root_squash,no_subtree_check)”**. Essa linha contém o diretório a ser adicionado e as permissões necessárias.

Agora devemos configurar a inicialização do serviço quando o sistema iniciar, para isso usamos o comando **“systemctl enable --now nfs-utils”**. Então devemos reiniciar o serviço nfs com o comando **“service restart nfs-utils”**, e então verificar se o compartilhamento está ativo. Para verificar se o compartilhamento está funcionando devemos executar o comando **“showmount -e”**, que irá retornar as pastas que estão compartilhadas no seu sistema. O comando irá retornar conforme a figura.



```
root@ip-172-31-0-23:/nfs-shi  ×  +  ∨  
[root@ip-172-31-0-23 igor]# showmount -e  
Export list for ip-172-31-0-23.ec2.internal:  
/nfs-share *  
[root@ip-172-31-0-23 igor]# |
```

Figura 02: Exemplo da execução do comando showmount -e. Fonte: Captura de tela do próprio autor

Para que a pasta fique utilizável em outras máquinas da rede deve ser criada uma pasta na máquina cliente, com o nome desejado. Após isso, com o comando **"sudo mount -t nfs 172.31.0.23:/nfs-share /diretório"** a pasta criada é montada com a pasta da rede. O IP inserido deve ser o do servidor de arquivos, o primeiro caminho de diretório é o da pasta da rede no servidor, e o segundo caminho é o caminho da pasta criada na máquina cliente.

2.3 CRIAÇÃO DO DIRETÓRIO

Como solicitado, para a criação do diretório com meu nome dentro do nfs é necessário primeiro acessar a pasta do nfs, com o comando **"cd /nfs-share"**, e então usar o comando **"mkdir Igor"**. Com o comando a pasta Igor é criada com sucesso dentro do diretório do nfs.

2.4 - INSTALAÇÃO DO APACHE:

A instalação do servidor apache no Linux foi realizada na instancia ec2 da AWS, pois esta possui um IP público fixo, o que facilita o acesso e testes.

Primeiro devemos acessar a instancia por SSH, após o acesso, viramos usuário root usando o comando **"sudo su -"**. Após isso efetuamos a atualização dos diretórios de programas do nosso Linux usando os comandos **"yum update"** e **"yum upgrade"**.

Com os diretórios atualizados podemos iniciar a instalação do servidor apache. Para isso usamos o comando **"yum install httpd -y"**. Com este comando a instalação do apache será iniciada, o parâmetro **"-y"** serve para que a instalação aceite todas as solicitações de Yes automaticamente, isso é mais utilizado para ter mais praticidade e diminuir um pouco o tempo de instalação.

Com o apache instalado, é necessário iniciar o serviço, para isso utilizamos o comando **"service httpd start"**. Para ter certeza de que o comando iniciou o serviço corretamente podemos utilizar o comando **"service httpd status"**, que irá retornar informações sobre o serviço, se está ativo (**active (running)**) ou inativo (**inactive (dead)**).

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 do servidor apache usando um navegador. Ao acessar o endereço devemos obter a tela padrão de boas vindas do apache conforme a figura.

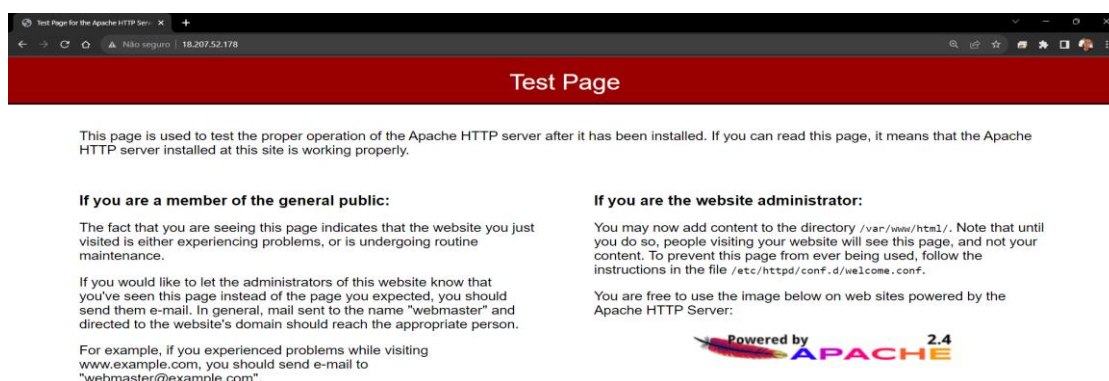


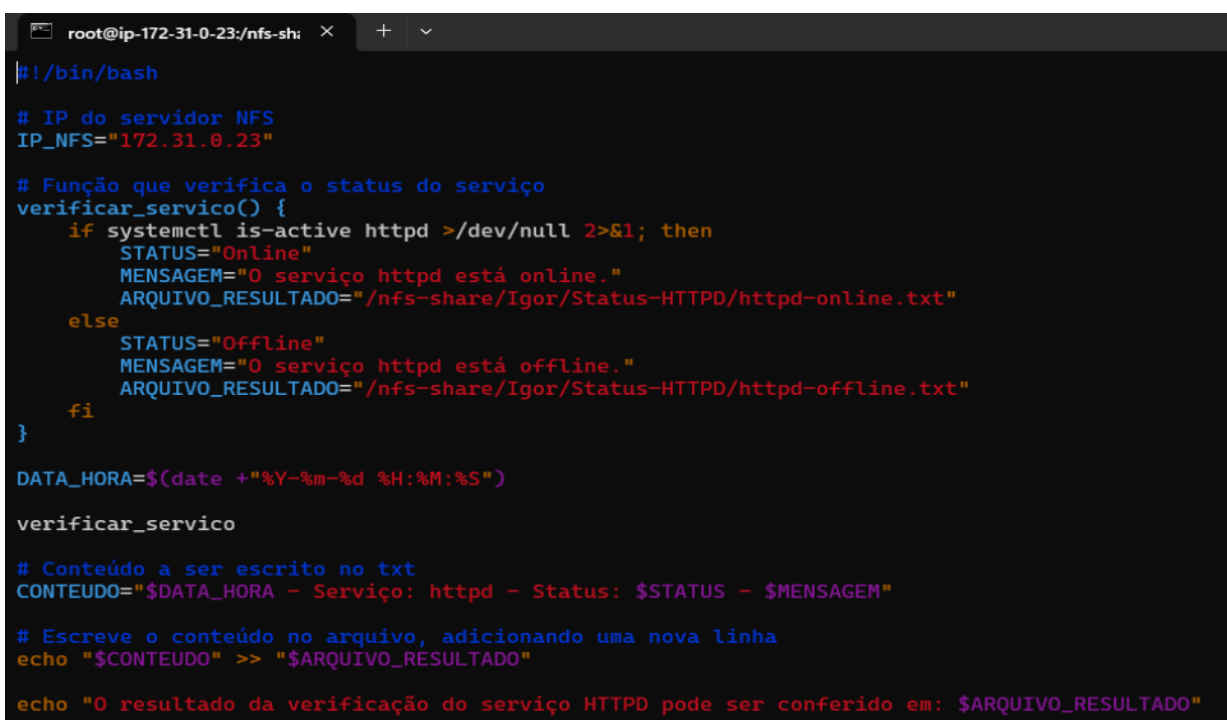
Figura 03: Pagina inicial do apache Fonte: Captura de tela do próprio autor

2.5 – SCRIPT DE VALIDAÇÃO SE O SERVIÇO ESTÁ ONLINE

No SO Linux é possível a configuração de scripts para as mais diversas funções. Uma delas é a verificação do status de um serviço, neste caso do apache (httpd).

Para a criação do script usamos o editor de texto vi, passando os parâmetros de nome de arquivo e extensão, que no caso é **“.sh”**. O script contém uma função para verificar se o serviço está online ou offline, em ambos os casos ele irá retornar uma mensagem em um documento de texto situado na pasta criada no nfs. São duas possibilidades de arquivos .txt, a primeira retorna quando o serviço estiver online, já a outra quando o serviço estiver offline. O arquivo contém a data e hora, o nome do serviço e seu status, seja ele online ou offline.

Abaixo podemos observar na figura o script **“HTTPD-ON-OFF.sh”** criado.



```

root@ip-172-31-0-23:/nfs-sh: x + v
#!/bin/bash

# IP do servidor NFS
IP_NFS="172.31.0.23"

# Função que verifica o status do serviço
verificar_servico() {
    if systemctl is-active httpd >/dev/null 2>&1; then
        STATUS="Online"
        MENSAGEM="O serviço httpd está online."
        ARQUIVO_RESULTADO="/nfs-share/Igor/Status-HTTPD/httpd-online.txt"
    else
        STATUS="Offline"
        MENSAGEM="O serviço httpd está offline."
        ARQUIVO_RESULTADO="/nfs-share/Igor/Status-HTTPD/httpd-offline.txt"
    fi
}

DATA_HORA=$(date +"%Y-%m-%d %H:%M:%S")

verificar_servico

# Conteúdo a ser escrito no txt
CONTEUDO="$DATA_HORA - Serviço: httpd - Status: $STATUS - $MENSAGEM"

# Escreve o conteúdo no arquivo, adicionando uma nova linha
echo "$CONTEUDO" >> "$ARQUIVO_RESULTADO"

echo "O resultado da verificação do serviço HTTPD pode ser conferido em: $ARQUIVO_RESULTADO"

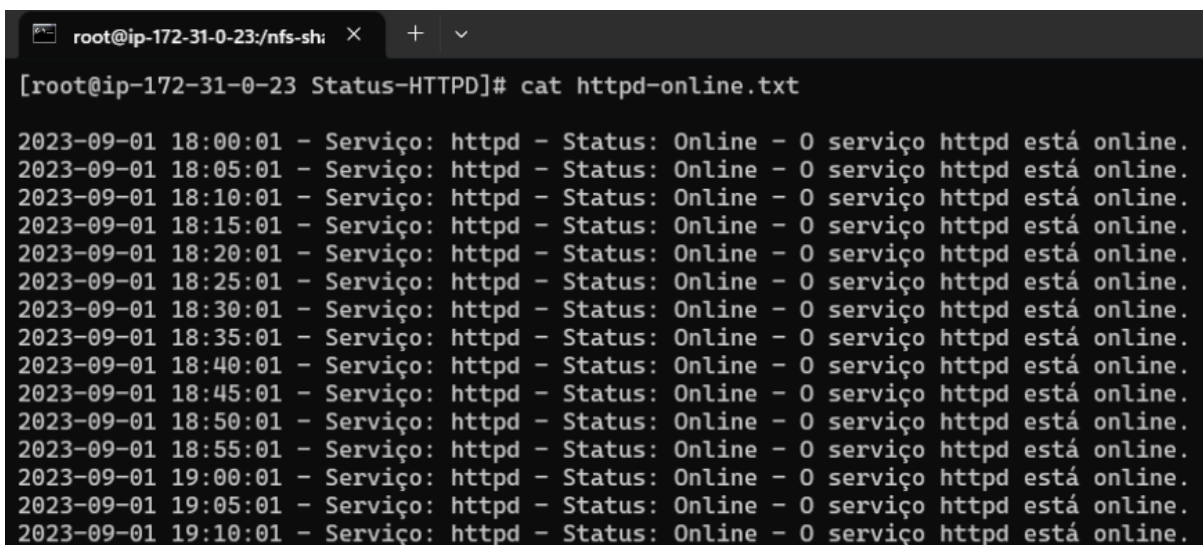
```

Figura 04: Script criado Fonte: Captura de tela do próprio autor

Após criado, devemos dar permissões de execução ao arquivo .sh, para isso usamos o comando **“chmod =x /diretório/script.sh”**, onde diretório é o diretório em que o arquivo .sh está localizado.

Após a permissão de execução do script é necessário configurá-lo para a execução em um tempo desejado, que no caso foi de 5 em 5 minutos. Para a configuração devemos acessar o crontab, que é como se fosse um agendador de tarefas do Linux. Para acessá-lo usamos o comando **“crontab -e”**, na tela que abriu adicionamos o seguinte comando **“*/5 * * * * /nfs-share/scripts/HTTPD-ON-OFF.sh”**. Esse comando configura a execução do script criado a cada 5 minutos. O crontab utiliza o mesmo mecanismo do editor de texto vi, então o comando para salvar e sair é o mesmo.

Abaixo podemos observar um exemplo do arquivo de texto que contém os resultados da execução automática do script pelo crontab. A execução ocorreu a cada 5 minutos, neste caso em todas as execuções o serviço estava online.

A screenshot of a terminal window with a dark background. The title bar at the top shows 'root@ip-172-31-0-23:/nfs-sh:'. The terminal content shows the command '[root@ip-172-31-0-23 Status-HTTPD]# cat httpd-online.txt' followed by a list of 15 log entries. Each entry shows a timestamp, the service name 'httpd', its status 'Online', and a count '0', followed by a human-readable message 'serviço httpd está online.'.

```
root@ip-172-31-0-23:/nfs-sh: X + v
[root@ip-172-31-0-23 Status-HTTPD]# cat httpd-online.txt
2023-09-01 18:00:01 - Serviço: httpd - Status: Online - 0 serviço httpd está online.
2023-09-01 18:05:01 - Serviço: httpd - Status: Online - 0 serviço httpd está online.
2023-09-01 18:10:01 - Serviço: httpd - Status: Online - 0 serviço httpd está online.
2023-09-01 18:15:01 - Serviço: httpd - Status: Online - 0 serviço httpd está online.
2023-09-01 18:20:01 - Serviço: httpd - Status: Online - 0 serviço httpd está online.
2023-09-01 18:25:01 - Serviço: httpd - Status: Online - 0 serviço httpd está online.
2023-09-01 18:30:01 - Serviço: httpd - Status: Online - 0 serviço httpd está online.
2023-09-01 18:35:01 - Serviço: httpd - Status: Online - 0 serviço httpd está online.
2023-09-01 18:40:01 - Serviço: httpd - Status: Online - 0 serviço httpd está online.
2023-09-01 18:45:01 - Serviço: httpd - Status: Online - 0 serviço httpd está online.
2023-09-01 18:50:01 - Serviço: httpd - Status: Online - 0 serviço httpd está online.
2023-09-01 18:55:01 - Serviço: httpd - Status: Online - 0 serviço httpd está online.
2023-09-01 19:00:01 - Serviço: httpd - Status: Online - 0 serviço httpd está online.
2023-09-01 19:05:01 - Serviço: httpd - Status: Online - 0 serviço httpd está online.
2023-09-01 19:10:01 - Serviço: httpd - Status: Online - 0 serviço httpd está online.
```

Figura 05: Arquivo status online: Captura de tela do próprio autor

2.6 CONSIDERAÇÕES FINAIS

Com a execução de todas as atividades pude pôr em prática muitos dos conteúdos explicados nos cursos de Linux e AWS assistidos até agora. Isso tudo serviu para aprimorar o conhecimento e fixar os conteúdos aprendidos.

Aprendi que a combinação de habilidades em Linux e AWS oferece um conjunto poderoso de ferramentas para implantar, gerenciar e escalar aplicativos na nuvem de forma eficiente e segura. Habilidades como implantar um servidor Apache, criar instâncias na AWS ou configurar servidores nfs no Linux são altamente valiosas no mundo da tecnologia e podem abrir oportunidades significativas.

Com as próximas atividades poderei expandir ainda mais meu conhecimento em ambas as áreas e explorar casos de uso mais avançados à medida que continuarei minha jornada na administração de sistemas e na computação em nuvem.