# **Documentação de Criação e Configuração**

## **AWS e Linux - Versão 2.0**

*Ana Maria Cavasin Data de Atualização: 04/09/2023*

### **Visão Geral**

A apresente documentação fornece instruções passo a passo para a instalação e configuração o de uma instância EC2 na AWS juntamente com a configuração do Linux. A seguir serão apresentados os tópicos que serão apresentados nessa documentação:

**AWS:**

* Geração de uma chave pública para acesso ao ambiente;
* Criação de uma instância EC2 com o sistema operacional Amazon Linux 2 (Família t3.small, 16 GB SSD);
* Geração de um Elastic IP e anexá-lo à instância EC2;
* Liberação das portas de comunicação para acesso público: (22/TCP, 111/TCP e UDP, 2049/TCP/UDP, 80/TCP, 443/TCP).

**Linux:**

* Configuração do NFS entregue;
* Criação um diretório dentro do filesystem do NFS com seu nome;
* Subir um apache no servidor - o apache deve estar online e rodando;
* Criação um script que valide se o serviço está online e envie o resultado da validação para o seu diretório no NFS.
  + O script contém a - Data Hora + nome do serviço + Status + mensagem personalizada de online ou offline. Além disso, o script gera 2 arquivos de saída: 1 para o serviço online e 1 para o serviço offline;
* Preparação de uma execução automatizada do script a cada 5 minutos.

### **Procedimento AWS**

Siga estas etapas para realizar as configurações. Em cada passo será apresentado uma instrução escrita passo a passo.

**Passo 1: Gerar uma chave pública para acesso ao ambiente**

* No Console de Gerenciamento da AWS, acesse a opção **EC2**.
* Agora busque a seção **Rede e segurança e** selecione a opção **Pares de chaves**
* Após isso, clique em **Criar par de chaves**.
* Insira um **nome para a chave** e escolha o tipo de par de chaves, o formato de arquivo da chave privada, se quiser pode adicionar tags.
* Por fim clique em **Criar par de chaves,** isso fará o download da chave privada automaticamente.
* **Guarde** essa chave privada em um l**ocal seguro**, pois você precisará dela para se conectar à instância EC2.

**Passo 2: Criar de uma instância EC2 com o sistema operacional Amazon Linux 2**

* No Console de Gerenciamento da AWS, acesse **EC2**.
* Acesse a opção **Instâncias**
* Clique em **Executar instâncias** para iniciar o processo de criação de uma instância.
* Configure as **Tags** da instância (**Name, Project e CostCenter**) para i**nstâncias e volumes.**
* Escolha a imagem **Amazon Linux 2 AMI (HVM), SSD Volume Type.**
* Na sessão Tipo de instância, selecione a opção **t3.small**
* Agora em Par de chaves, selecione a chave gerada no Passo 1.
* Em configuração de rede, verifique se a sub-rede de criação é pública. Se não for, acesse **Criar nova sub-rede** e siga o **passo 3.**
* Ainda em configuração de rede, agora será configurado o firewall (grupo de segurança), pode-se **criar grupo de segurança** ou **selecionar grupo de segurança existente**.
* Nesse caso será **criado um no grupo de segurança,** pode-se r**ealizar as configurações** nesse momento clicando em editar e seguindo o **passo 6**, ou posteriormente.
* Agora em **armazenamento (volumes)**, altere o tamanho para **16 GiB** e o Tipo de volume em **gp2**.
* Agora revise as informações sobre a instância que será criada e click em Executar instância.
* Em poucos segundos a instância estará pronta para ser acessada.

**Passo 3: Criar uma sub-rede pública**

Se em sua VPC não possui um sub-rede pública, siga os seguintes passos para cria-la.

* No Console de Gerenciamento da AWS, acesse VPC.
* Acesse a opção **Sub-redes**
* Clique em **Criar sub-rede** para iniciar o processo de criação.
* Selecione o **id da VPC** em que você deseja criar a sub-rede.
* Agora configure o **nome da sua sub-rede** e em qual **Zona de disponibilidade** deseja **residir a sua sub-rede**.
* Após isso, escolha o **bloco CIDR IPv4**. Os tamanhos dos blocos IPv4 devem estar entre uma máscara de rede /16 e uma máscara de rede /28 e podem ser do mesmo tamanho da VPC.
* Após isso, click em **Criar sub-rede**.

**Passo 4: Criar um internet gateware**

Antes de associarmos um Elastic ip em nossa máquina, precisamos criar um internet gateware.

* No Console de Gerenciamento da AWS, acesse VPC.
* Acesse a opção **Gateways da Internet**
* Clique em **Criar gateways da Internet** para iniciar o processo de criação.
* Escolha a Tag de nome do seu gateway da Internet e click em **Criar gateways da Internet**
* Com o seu gateway criado, agora vamos associá-lo a uma VPC. Para isso selecione o gateway criado e acesse as **ações.**
* Ao clicar em ações, aparecerá a opção **Associar à VPC.**
* Ao clicar nele, você deve selecionar a **VPC da instância EC2** criada anteriormente e clicar em **Associar**.

**Passo 5: Configurar rota de internet**

Agora é preciso configura a rota de internet para que a nossa instância EC2 tenha acesso a internet

* No Console de Gerenciamento da AWS, acesse VPC.
* Acesse a opção **Tabelas de rotas**
* Selecione a **tabela de rotas da VPC da instância EC2** criada anteriormente.
* Clique em **Ações** > **Editar rotas**.
* Clicar em **Adicionar rota**.
* Configurar da seguinte forma:
  + **Destino**: 0.0.0.0/0
  + **Alvo:** Selecionar o gateway de internet criado anteriormente
* Clicar em **Salvar alterações**.

**Passo 6: Configurar o Security Group para a instância EC2**

* Acessar a AWS no serviço EC2,
* Busque a seção **Rede e segurança e** selecione a opção **Security group**
* Agora acesse o **grupo de segurança criado anteriormente**, vá em **Regras de Entrada** e clique em **Editar regras de entrada**.
* Após isso, clique em **Adicionar regra** para adicionar as regras de entrada necessárias para liberar as portas de comunicação.
* Agora você deve **criar todas as regras** apresentadas na **tabela** a seguir:

|  |  |  |  |
| --- | --- | --- | --- |
| **Tipo** | **Protocolo** | **Intervalo de portas** | **Origem** |
| UDP personalizado | UDP | 111 | 0.0.0.0/0 |
| HTTPS | TCP | 443 | 0.0.0.0/0 |
| HTTP | TCP | 80 | 0.0.0.0/0 |
| SSH | TCP | 22 | 0.0.0.0/0 |
| UDP personalizado | UDP | 2049 | 0.0.0.0/0 |
| TCP personalizado | TCP | 111 | 0.0.0.0/0 |
| NFS | TCP | 2049 | 0.0.0.0/0 |

* Após configurar todas as regras de entrada, clique em **Visualizar alterações** para revisar suas configurações.
* Revise suas configurações na página de revisão e, se tudo estiver correto, clique em **Salvar regras** para iniciar a criação da instância EC2 com as configurações de segurança definidas.

**Passo 7: Criando um Elastic ip e associando a uma instancia EC2**

* Acessar a AWS na página do serviço EC2.
* Busque a seção **Rede e segurança** e selecione a opção **IPs elásticos**
* Acesse a opção **Alocar endereço IP elástico**
* Selecione o **Grupo de Borda de Rede** e click em **Alocar**
* Após isso, vamos **alocar o IP a uma instância EC2**
* Para fazer isso basta selecionar o **IP**, ir em ações e selecionar **Alocar IP elástico**
* Agora basta selecionar o **tipo de recurso** que você gostaria de associar, instância ou interface de rede. Nesse caso **selecione instância**.
* Depois selecione **qual instância** você gostaria de associar e o seu **IP privado**.
* Agora clique em **Associar instância**.

### **Procedimento Linux**

Siga estas etapas para realizar as configurações. Em cada passo será apresentado uma instrução escrita, bem como um imagens passo a passo.

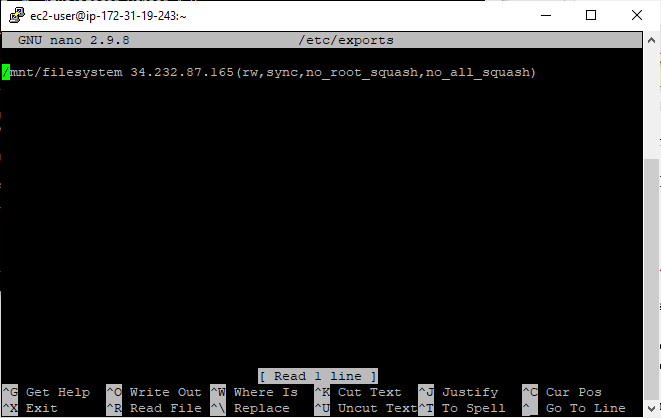
**Pré-requisitos**

* **2 instâncias EC2** com o Sistema Operacional Amazon Linux 2, criadas na mesma de preferência na mesma sub-rede.
* A **configuração das instâncias** deve ser feita conforme a **sessão anterior de procedimento AWS.**

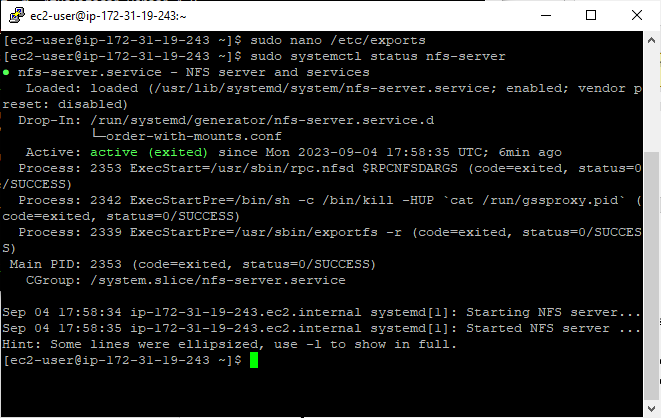
**Passo 8: Configurar o NFS – Servidor**

**Acesse** **uma da instância** criadas previamente via **SSH**, faça as seguintes configurações para criarmos um **servidor NFS**.

* Primeiramente vamos **atualizar o sistema** com o *sudo yum update -y*
* Após isso vamos **instalar os pacotes** para a configuração do **NFS** *sudo yum install nfs-utils –y*
* Precisamos **criar uma pasta** onde os arquivos serão compartilhados entre as duas máquinas *sudo mkdir /mnt/filesystem*
* Agora é necessário fazemos as **configurações** do **NFS Server**. Para isso, precisamos acessar o seguinte arquivo *sudo nano /etc/exports*
* Dentro desse arquivo vamos colocar a seguinte */mnt/filesystem [IP\_SERVIDOR\_CLIENT](rw,sync,no\_root\_squash\_all\_squash)*



* Salve o arquivo e saia.
* Agora é necessário **ativar o serviço NFS** com o seguinte comando *sudo systemctl enable nfs-server*
* Agora vamos **iniciar o NFS** com o seguinte comando *sudo systemctl start nfs-server*
* Para **verificarmos se ele está iniciado** basta digitar o seguinte comando *sudo systemctl status nfs-server*

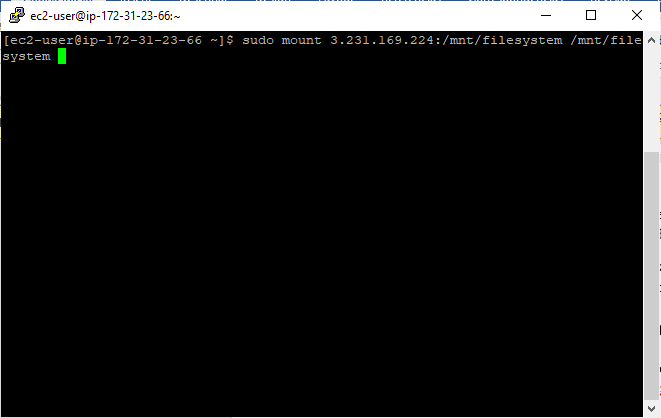


* Agora precisamos alterar as permissões da pastacom os seguinte comando *sudo sudo chmod -R 777 /mnt/filesystem/* e o *sudo chmod -R 777 /mnt/*

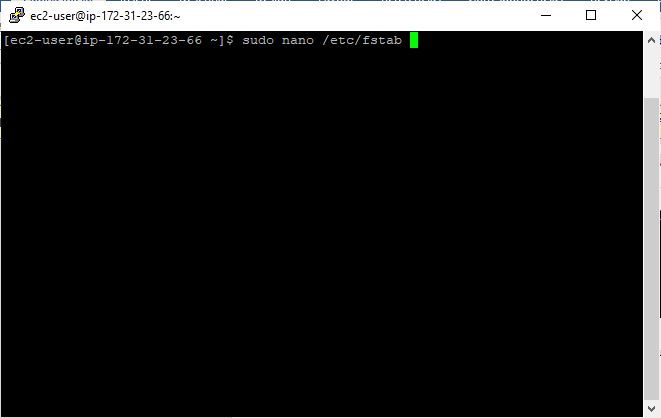
**Passo 9: Configurar o NFS – Cliente**

Acesse a **outra instância** criada previamente via **SSH**, faça as seguintes configurações para criarmos um **Servidor Cliente**, onde será desenvolvida a atividade.

* Primeiramente vamos **atualizar o sistema** com o *sudo yum update -y*
* Após isso vamos **instalar os pacotes** para a configuração do **NFS** *sudo yum install nfs-utils –y*
* Precisamos **criar uma pasta** onde os arquivos serão compartilhados *sudo mkdir /mnt/filesystem*
* Agora vamos **montar o compartilhamento NFS** no diretório criado anteriormente com o seguinte comando *sudo mount [IP\_SERVIDOR\_NFS]:/mnt/filesystem /mnt/filesystem*



* Agora é preciso **montar automaticamente no boot**, para isso acesse o seguinte arquivo *sudo nano /etc/fstab*



* Dentro do arquivo adicione as seguintes configurações *[IP\_SERVIDOR\_NFS]:/mnt/filesystem /mnt/filesystem nfs defaults 0 0*

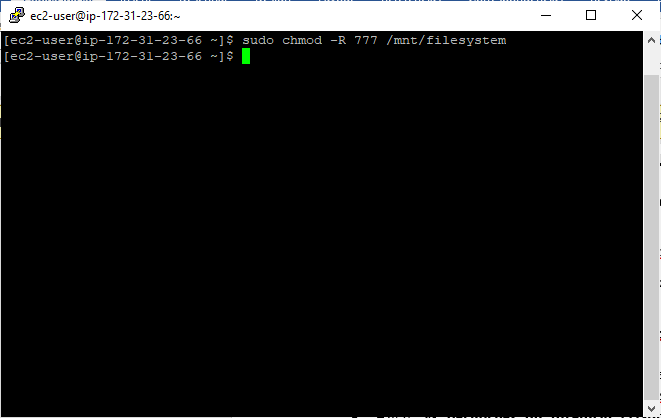


* Salve o arquivo e saia.
* Após isso, devemos **montar todas as entradas do arquivo /etc/fstab** com o seguinte comando *sudo mount –a*

**Passo 10: Criar um diretorio dentro do filesystem do NFS com seu nome**

Ainda no servidor cliente faça os seguintes comandos para a criação de uma pasta com seu nome.

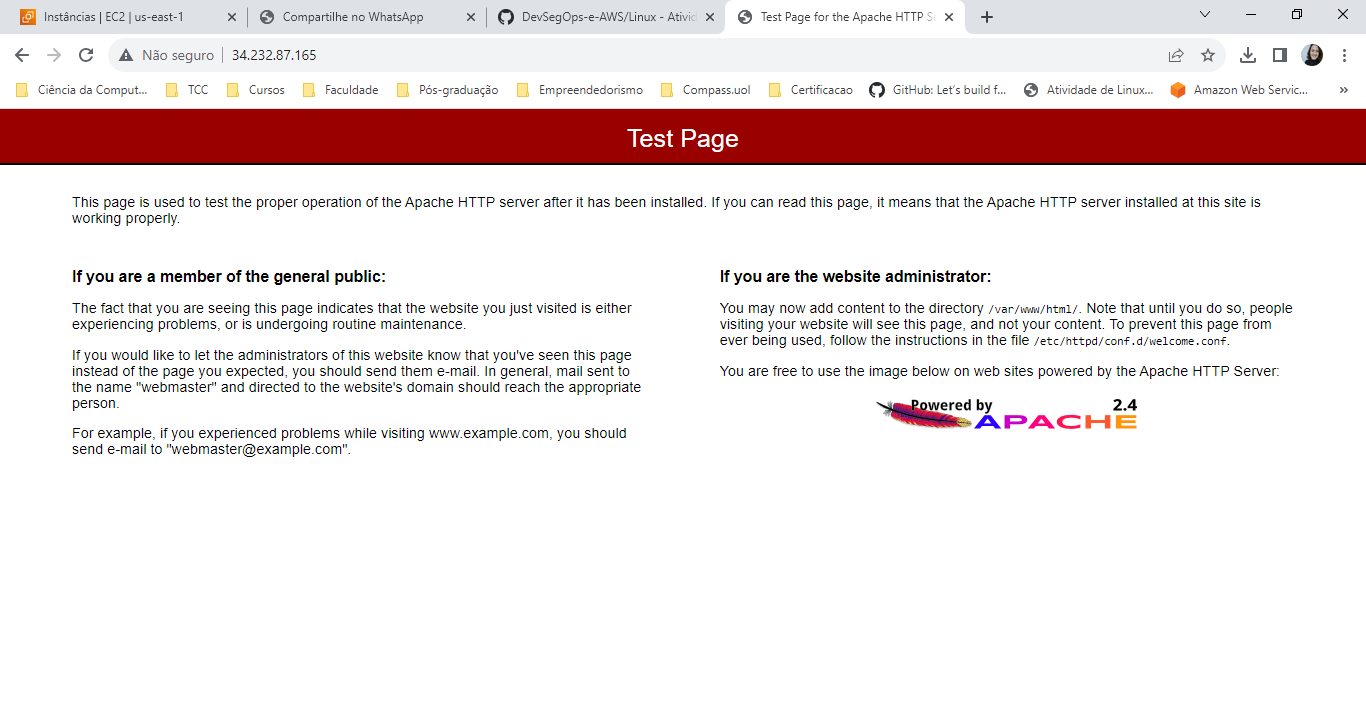
* Primeiramente precisamos **alterar as permições** do diretório criada anteriormente *sudo chmod -R 777 /mnt/filesystem*



* Após isso, crie a **nova pasta com seu nome** *sudo mkdir /mnt/filesystem/[SEU\_NOME]*
* Altere as **permições do diretório criado com seu nome** *sudo chmod -R 777 /mnt/filesystem/[SEU\_NOME]/*

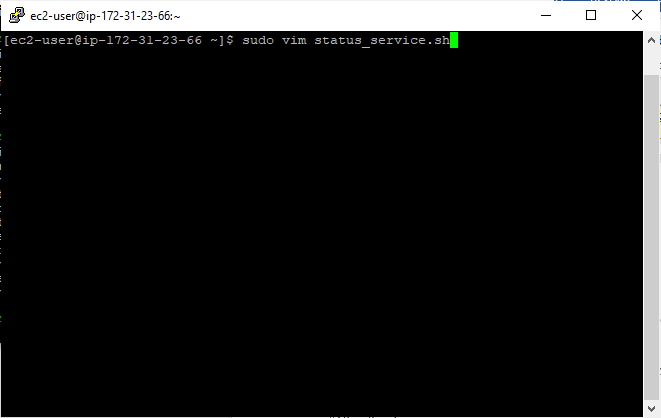
**Passo 11: Subir um apache no servidor**

A partir desse momento todos os passos devem ser feitos na instância EC2 do Servidor Cliente.

* Primeiramente vamos **instalar os pacotes** para a **configuração do Apache** *sudo yum install httpd*
* Agora é necessário a**tivar o serviço do Apache** com o seguinte comando *sudo systemctl enable httpd*
* Agora vamos **iniciar o Apache** com o seguinte comando *sudo systemctl start httpd*
* Para **verificarmos se ele está iniciado** basta digitar o seguinte comando *sudo systemctl status httpd*
* Também podemos acessa-lo via navegador, conforme a imagem a seguir

**Passo 12: Criar um script que valide se o Apache está online e envie o resultado da validação para o diretório no NFS**

* Para **criarmos o script** utilizaremos o seguinte comando *sudo vim status\_service.sh*



* Agora **dentro do arquivo** criamos o script

*#!/bin/bash*

*service\_name="Apache"*

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

*status=$(systemctl is-active httpd)*

*if [ "$status" == "active" ]; then*

*message="Online"*

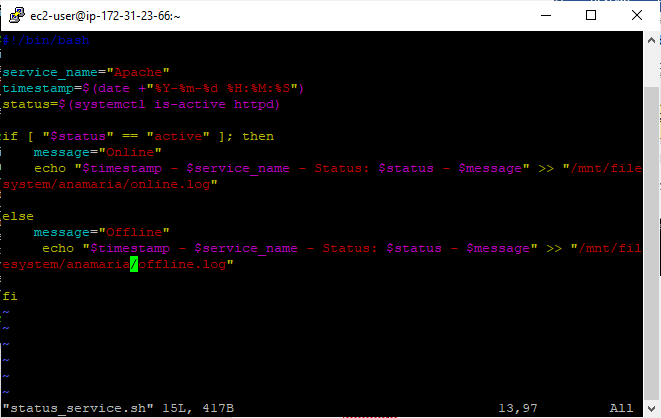
*echo "$timestamp - $service\_name - Status: $status - $message" >> "/mnt/filesystem/anamaria/online.log"*

*else*

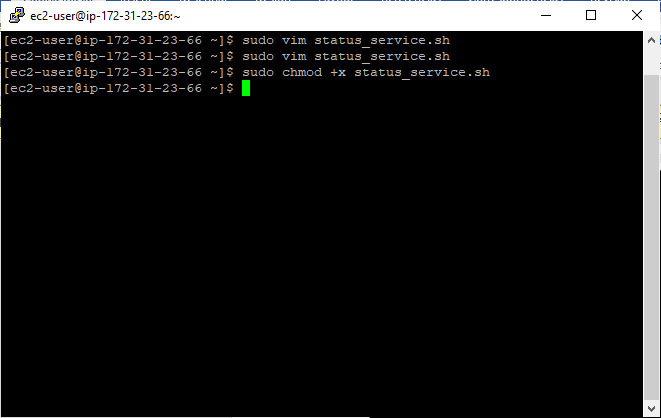
*message="Offline"*

*echo "$timestamp - $service\_name - Status: $status - $message" >> "/mnt/filesystem/anamaria/offline.log"*

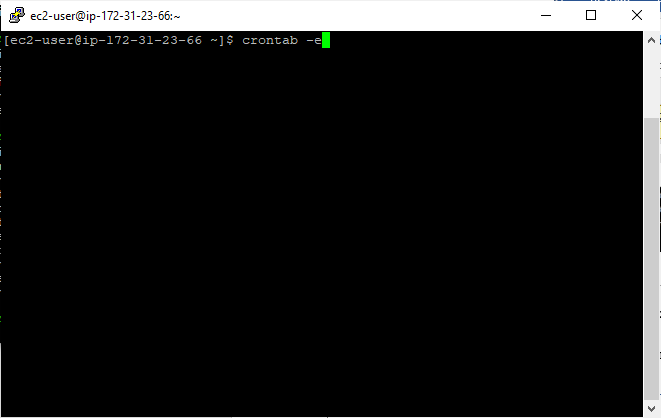
*fi*

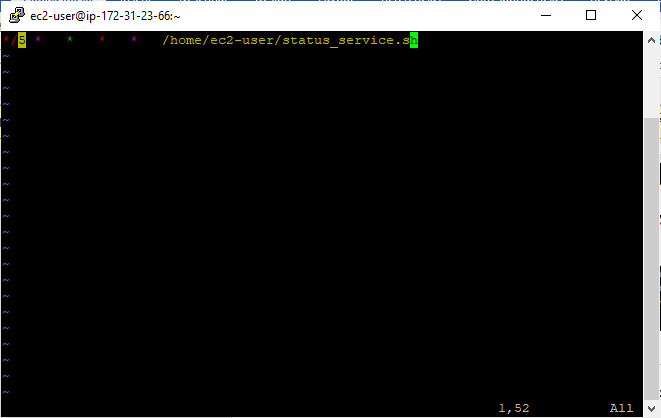


* Salve o arquivo do script e saia.
* Agora precisamos **deixar o script execultavel** com o seguinte comando *sudo chmod +x status\_service.sh*

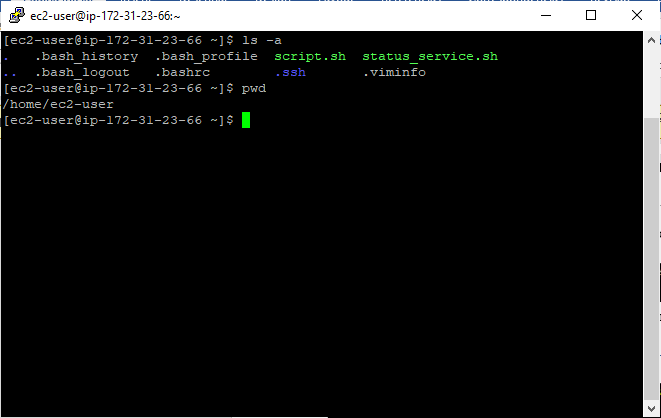


**Passo 13: Execução automatizada do script a cada 5 minutos**

* Para deixarmos o script executável automaticamente precisamos **acessar** o *crontab -e*
* Agora vamos **adiciona**r o seguinte comando *\*/5 \* \* \* \* /home/ec2-user/status\_service.sh*

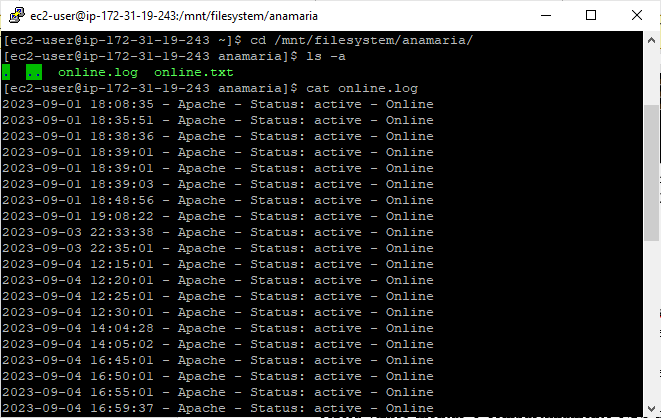


* Salvar e sair
* Para funcionar corretamente é necessário que seja colocado o **caminho completo do arquivo**. Para saber ele utilize o comando *pwd*

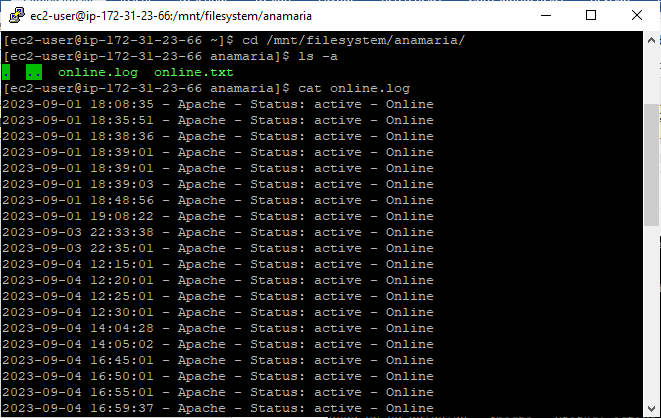


### **Conclusão**

Com esse tutorial teremos o seguinte resultado. Na instancia EC2 do Servidor Cliente termos o script que verifica o estado do Apache sendo executando a cada 5 min e envido o resultado pra o diretório com o seu nome no NFS e podendo ser visto no Servidor NFS e outras maquinas.



Resultado Final do Servidor NFS



Resultado Final Servidor Cliente

Pronto! Agora seus servidores estão configurados. Caso haja alguma dúvida ou problema acesse a documentação oficial da AWS a seguir <https://docs.aws.amazon.com/> ou na documentação do Amazon Linux 2 <https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/amazon-linux-ami-basics.html>