

Portfólio de Ettory Martins de Souza

Resumo Profissional

Sou um profissional com sólida experiência em administração de sistemas e infraestrutura, focado na implementação de soluções eficientes e seguras para ambientes corporativos. Possuo habilidade comprovada em diagnosticar e resolver problemas técnicos complexos, além de desenvolver documentações e processos que otimizam a operação de TI. Sou comprometido com a evolução contínua, aplicando novas tecnologias e práticas para entregar resultados que agregam valor ao negócio.

Objetivo do Projeto

O objetivo central do projeto foi configurar, implementar e provisionar um banco de dados PostgreSQL em um servidor Ubuntu Server 24.04 LTS, para armazenamento e gerenciamento efetivos dos dados. O sistema foi devidamente virtualizado utilizando de um virtualizador tipo II (VirtualBox), enquanto o gerenciador do banco de dados PgAdmin, foi instalado e configurado na máquina física do administrador, de maneira externa ao servidor. O projeto teve enfoque na configuração do servidor, ajustes de segurança de serviço na rede e tuning do banco PostgreSQL para maior desempenho e eficiência.

Logo abaixo estarão descritos os componentes do cenário apresentado:

1. Máquina Virtual: Ubuntu Server (CLI) 24.04 LTS, devidamente configurado no virtualizador VirtualBox.
2. Banco de Dados: PostgreSQL, instalado localmente na máquina virtual Ubuntu Server 24.04 LTS.
3. Ferramenta de Administração: pgAdmin, instalado na máquina física do DBA (Database Administrator) .
4. Rede: Configurações de rede entre as máquinas virtual e física para acesso ao banco de dados via gerenciador PgAdmin serão tratadas de maneira mais específica ao decorrer do processo de desenvolvimento do projeto.

Instalando o Sistema Operacional

Daremos uma breve descrição de como o sistema operacional da virtual machine foi instalado e configurado, passando por sessões significativas como particionamento de disco, configurações de rede e de segurança por firewall. O firewall utilizado será o UFW para fácil compreensão do funcionamento da segurança, e pela facilidade visual dos bloqueios e liberações de portas.

Para visualização dos recursos computacionais de forma resumida da máquina virtual:

Armazenamento Primário (RAM): 4 GB

Armazenamento Secundário (HDD): 100 GB

Interface de rede: Modo Bridge para contato na mesma LAN simulando ambiente real de infraestrutura empresarial local.

Particionamento de Disco

Aqui, utilizamos um disco de 100 GB para este projeto, alinhado à duas tecnologias muito conhecidas de sistemas de kernel Linux, o LVM (Logical Volume Manager) que nos possibilita ter uma vantagem de flexibilidade ao decorrer do uso constante do servidor e o preenchimento do disco, bem como aumentar a escalabilidade, e o LUKS (Linux Unified Key Setup) que faz com que o disco seja criptografado para proteção dos dados no disco de maneira integral.

Com os conceitos já apresentados, iremos para a parte do particionamento que seguirá o padrão abaixo:

- PV (Physical Volume):
 - LV (Logical Volume):
 - VG (Volume Group):
 - Raiz (/): 30 GB → Sistema operacional principal.
 - Home (/home): 50 GB → Dados de usuários.
 - Var (/var): 9 GB → Logs, cache, entre outros.
 - Opt (/opt): 10 GB → Pacotes de aplicativos adicionais.

- Boot (/boot): 1 GB → Partição separada do LVM, onde ficam armazenados arquivos essenciais de inicialização, como o kernel.

Configurações de rede

Para a configuração de rede, iremos utilizar de um recurso nativo do sistema Ubuntu Server 24.04 que está presente desde sua versão 20.04, o `netplan`. Este é um arquivo de configuração presente no seguinte path `/etc/netplan/`, e utiliza a extensão `.yaml` para sua configuração.

Precisamos encontrar o nome da interface de rede com o comando `ip a` e, logo em seguida, devemos editar o arquivo com as seguintes informações:

```
network:
  version: 2
  renderer: networkd
  ethernets:
    enp0s3:
      dhcp4: no
      addresses:
        - 192.168.3.3/24
      gateway4: 192.168.3.1
      nameservers:
        addresses:
          - 8.8.8.8
          - 8.8.4.4
```

Logo após definir estas configurações, devemos executar o comando `sudo netplan apply` para aplicar as configurações ao sistema, e utilizar o comando `ip a` novamente para checar as alterações.

Configurações de Firewall

Devemos agora aplicar todas as regras de firewall presentes no projeto, e também instalar o SSH. Isso será descrito abaixo em forma de um `shell script` para melhor compreensão e leitura dos passos realizados:

```
#!/bin/bash

sudo ufw enable

sudo ufw default deny incoming

sudo ufw default allow outgoing

sudo ufw allow from 192.168.3.111 to any port 22

sudo ufw allow from 192.168.3.111 to any port 5432

sudo apt install ssh -y
```

Isto fará com que as portas 5432 e 22 sejam acessíveis apenas para o administrador do banco de dados por meio da filtragem de portas.

Para realizar a checagem, devemos executar o comando `"sudo ufw status"` e este deverá retornar a saída abaixo:

```
Status: active
```

To	Action	From
--	-----	----
22	ALLOW	192.168.3.111
5432	ALLOW	192.168.3.111

Instalando o PostgreSQL

Para instalar o serviço do PostgreSQL vamos utilizar o `shell script` abaixo para simplificar a leitura do procedimento:

```
#!/bin/bash
```

```
sudo apt update
```

```
sudo apt install postgresql postgresql-contrib -y
```

Para checar se tudo correu bem, vamos fazer o uso do comando `"sudo systemctl status postgresql"`. Devemos obter uma resposta como a descrita abaixo:

- `postgresql.service` - PostgreSQL RDBMS

```
Loaded: loaded (/lib/systemd/system/postgresql.service;  
enabled; vendor preset: enabled)
```

```
Active: active (exited) since Tue 2024-12-19 10:33:21  
UTC; 1min 30s ago
```

```
Main PID: 1234 (code=exited, status=0/SUCCESS)
```

```
Tasks: 0 (limit: 2344)
```

```
Memory: 5.0M
```

```
CGroup: /system.slice/postgresql.service
```

Para acessar a ferramenta de gerenciamento do PostgreSQL, devemos utilizar os comandos abaixo em ordem:

```
sudo -i -u postgres
```

```
psql
```

Para sair da ferramenta CLI de gerenciamento do PostgreSQL, utilizamos o comando `"\q"`.

Configurando o PostgreSQL Para Aceitar Conexões Externas

Para realizar este passo, iremos fazer o uso do arquivo localizado em `"/etc/postgresql/16/main/postgresql.conf"` e editar a linha descrita abaixo para a sua respectiva linha abaixo:"

```
listen_address = 'localhost'
listen_address = '*'
```

Agora, o PostgreSQL irá escutar conexões de qualquer endereço IP.

Vamos permitir agora que o administrador possa se conectar ao banco informando seu endereço IP. Para isso adicionamos a linha descrita abaixo no arquivo localizado em `"/etc/postgresql/16/main/pg_hba.conf"`:

```
host      all             all             192.168.3.111/24          md5
```

Após isso, reiniciaremos o serviço do PostgreSQL com o comando `"sudo systemctl restart postgresql"`.

Criando Novo Usuário e Banco de Dados

Para realizar este passo, iremos entrar na ferramenta CLI de gerenciamento do PostgreSQL através do comando `"sudo -i -u postgres psql"` e seguir os seguintes passos abaixo para a criação de um banco de dados e de um novo usuário, respectivamente:

```
CREATE DATABASE novo_banco;
```

```
CREATE USER novo_usuario WITH PASSWORD 'senha';  
GRANT ALL PRIVILEGES ON DATABASE novo_banco TO novo_usuario;
```

Conclusão

Concluimos este processo simples de criação de um servidor de banco de dados PostgreSQL 16 para fins de laboratório e de amostragem das devidas práticas, a fim de exemplificar todo o processo.