Universidade Federal do ABC

Sistemas Distribuídos

Sistema de Backup Distribuído

Alunos:

- Gabriel Yuto Munakata / RA: 11201721879
- Gabriel Fernandes / RA: 11201720718

Descrição do projeto:

- O objetivo do projeto é criar um sistema distribuído responsável por realizar backups de arquivos com suporte a replicação de conteúdo.
- O sistema é composto de servidores, um gerenciador e um ou mais clientes.
- Qualquer um dos servidores pode receber o arquivo a ser armazenado. Entretanto, o servidor que armazena o arquivo é escolhido pelo gerenciador. Além de escolher o servidor que armazena o arquivo, o gerenciador também escolhe um outro servidor para receber uma réplica do arquivo que foi enviado ao sistema de backup. Tal informação de replicação é enviada no cabeçalho do arquivo.
- Outro aspecto, é que o servidor que foi escolhido para receber o arquivo é quem envia a réplica para o servidor de réplica. Quaisquer servidores podem ser o servidor principal e o servidor de réplica de uma operação de backup.
- Para o cliente, é exibido um menu onde a operação de backup será iniciada. Assim, por meio dessa interface do cliente, devemos informar o nome do arquivo que será enviado ao sistema de backup.
- Os arquivos que poderão ser enviados ao sistema de backup estão dentro de um diretório raíz de cada cliente.

Configuração:

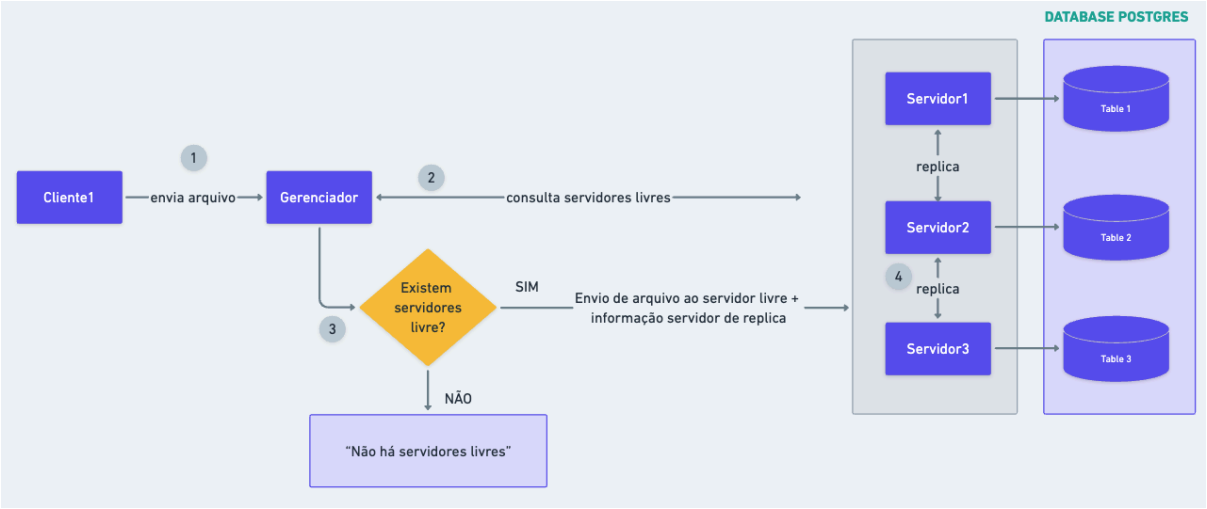
Para configurar o projeto, utilizamos o docker como recurso para a criação do container responsável pelo banco de dados PostgreSQL. Optamos por utilizar tal recurso como forma de armazenar os arquivos salvos pelos servidores.

Junto a isso, também utilizamos o ambiente virtual do python como forma de manter a consistência das bibliotecas para a comunicação com o banco de dados e

demais recursos. Ressaltamos que não foram utilizadas nenhuma biblioteca ou framework para fazer a comunicação, que se baseou em sockets. Apenas utilizamos bibliotecas nativas do python e o "psycopg2", para comunicação com o banco de dados.

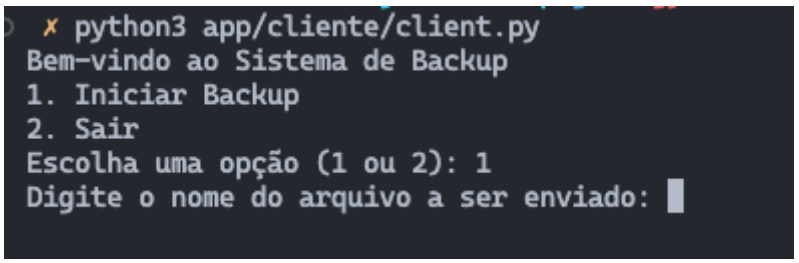
Arquitetura do sistema

Na figura abaixo, temos um detalhamento da arquitetura do sistema distribuído de backup de arquivos. Ela exemplifica o comportamento que foi desenvolvido para a comunicação entre o cliente, o gerenciador e os servidores.



Cada etapa do funcionamento é descrito abaixo:

1. Cliente Envia Arquivo (Passo 1):
   - O Cliente1 é responsável por enviar um arquivo ao gerenciador. Neste momento, é exibido uma interface no terminal para que o cliente opte pelo envio do arquivo ou cancelar a operação. Caso decida por enviar o arquivo, é feito uma pergunta sobre o nome do arquivo a ser enviado, de modo que o arquivo se encontra no mesmo diretório do cliente.



2. Consulta de Servidores Livres (Passo 2):
    - Ao receber o arquivo, o Gerenciador verifica quais servidores estão livres para armazenar o novo arquivo. Ele consulta a disponibilidade dos servidores para determinar onde o arquivo pode ser armazenado. Nesta etapa, o gerenciador, além de escolher um primeiro servidor livre, também busca por um segundo servidor com armazenamento livre para enviar no cabeçalho do arquivo, o qual servirá de servidor de réplica para que o primeiro servidor escolhido encaminhe o arquivo.
3. Verificação de Disponibilidade e Envio de Arquivo (Passo 3):
    - O Gerenciador verifica se existem servidores livres para armazenar o arquivo. Caso um servidor livre e um servidor de réplica sejam encontrados, o arquivo é enviado para o primeiro servidor, junto a um cabeçalho informando qual o servidor de réplica a ser utilizado nesse processo.
    - Se não houver servidores livres, depois de algumas tentativas de busca, o gerenciador informa a mensagem de que não há servidores livres.
    - Depois de receber o arquivo do gerenciador, o servidor escolhido como backup salva o arquivo em uma tabela presente do banco de dados. No banco de dados, existem tabelas específicas para cada servidor.
4. Replicação de Dados (Passo 4):
    - Uma vez que o arquivo é armazenado no primeiro servidor livre (por exemplo, Servidor1), este servidor realiza a replicação do arquivo para o próximo servidor que foi escolhido pelo gerenciador e enviado como informação no cabeçalho do arquivo (por exemplo, do Servidor1 para o Servidor2), garantindo que existam duas cópias do arquivo em dois servidores distintos, o que aumenta a segurança e a confiabilidade do backup.

Como executar a aplicação:

Para executar a aplicação, é necessário que tenha o docker instalado na máquina para configurar o banco de dados PostgreSQL (informação de instalação do docker neste link). Com o docker instalado, devemos executar o arquivo docker-compose, para que seja criado o container com o banco de dados. Para isso, basta executar o comando "docker-compose up -d".

Antes de iniciar os scripts python, é importante utilizar os ambientes virtuais do python. Sendo assim, em cada terminal, antes de proceder com a execução do

arquivo, deve ser carregado o ambiente com o comando "source venv/bin/activate".

Com isso, foi criado um script python para criar as tabelas no banco. O script se encontra na pasta "app/preparedb.py". Assim, basta executar o comando com "python3 app/preparedb.py".

Assim, temos já o ambiente configurado para executar o projeto. Dessa forma, de posse de 5 terminais distintos, devemos executar em cada terminal os arquivos referente ao sistema. Em um terminal, executamos o *client* que se encontra na pasta "app/cliente/client.py". Nessa pasta, também há os arquivos a serem encaminhados, conforme descrito na documentação solicitada do projeto.

Em outro terminal, devemos executar o *manager*, que se encontra na pasta "app/gerenciador/manager.py". Por fim, nos demais três terminais, executamos em cada um o *server1*, *server2* e *server3*, que se encontram na pasta "app/servidores".

Com isso, podemos iniciar o envio de arquivos pelo *client*, e analisar o comportamento dos dados sendo armazenados e replicados.

Link do repositório git: https://github.com/gabrielyuto/UFABC-SD-Projeto-Backup