

Alta disponibilidade com PostgreSQL

Em ambientes críticos, a tolerância à falhas é essencial ainda mais quando falamos de banco de dados. É muito importante que os seus dados estejam protegidos à falhas de hardware, eletricidade, etc., para que seu negócio tenha um plano de continuidade. A replicação de dados e backup são ótimas práticas e são adotadas em quase todos os ambientes nos dias de hoje, neste post veremos um pouco além, veremos aqui como implementar um o cluster de PostgreSQL em alta disponibilidade (HA) utilizando a ferramenta Repmgr e utilizaremos também um balanceador de carga para distribuir as queries entre as instâncias com o PgPool2.

Pré-requisitos:

- Utilizar um SO Linux
 - GIT (<https://git-scm.com/book/en/v2/Getting-Started-Installing-Git>)
 - Vagrant (<https://www.vagrantup.com/downloads.html>)
 - Virtual Box (<https://www.virtualbox.org/wiki/Downloads>)
 - RAM: 2GB (livre)
-

Provisionando o Lab

Para realizar o provisionamento deste laboratório usaremos o Vagrant, que é uma ferramenta muito prática para o provisionamento de ambientes virtuais de testes e desenvolvimento, em nosso blog você pode aprender um pouco mais sobre ele e como instalar acessando [aqui](#). Os arquivos necessários passo o nosso laboratório estão disponíveis em um repositório do GitHub, então em sua máquina com o GIT instalado execute o comando:

```
git clone https://github.com/Caeir0ps/postgresql-ha.git
```

Acesse o diretório criado, pois iremos adotar este como workspace para executar o Vagrant. Se listarmos o conteúdo deste diretório veremos o Vagrantfile, arquivo do Vagrant utilizado para criar as VMs com as configurações necessárias de recurso e permissões para o nosso laboratório. Para criar as máquinas virtuais execute o comando:

```
vagrant up
```