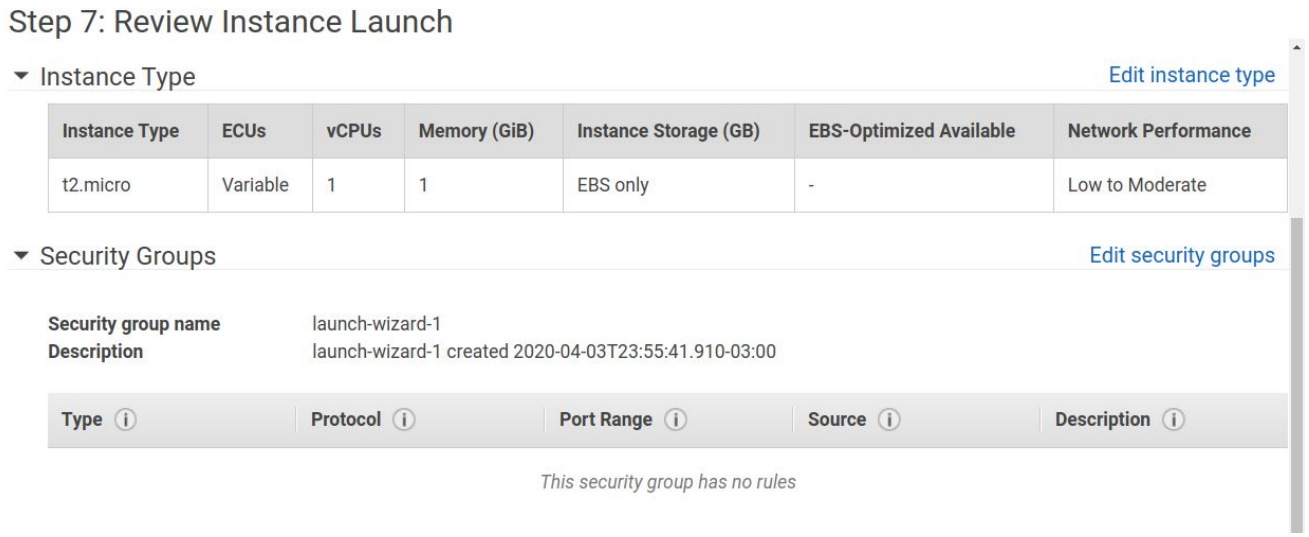
Configurando o cluster na AWS

Mário Olímpio de Menezes

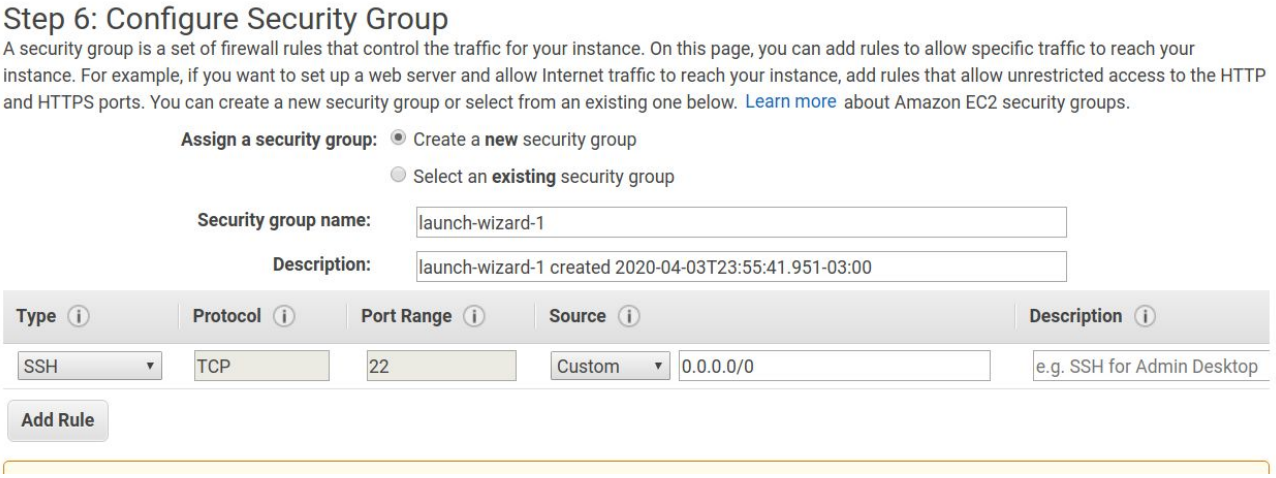
Criando as máquinas

Uma etapa fundamental ao criar as máquinas é utilizar um **security group** único para todas as máquinas. Isso é necessário para as configurações posteriores.

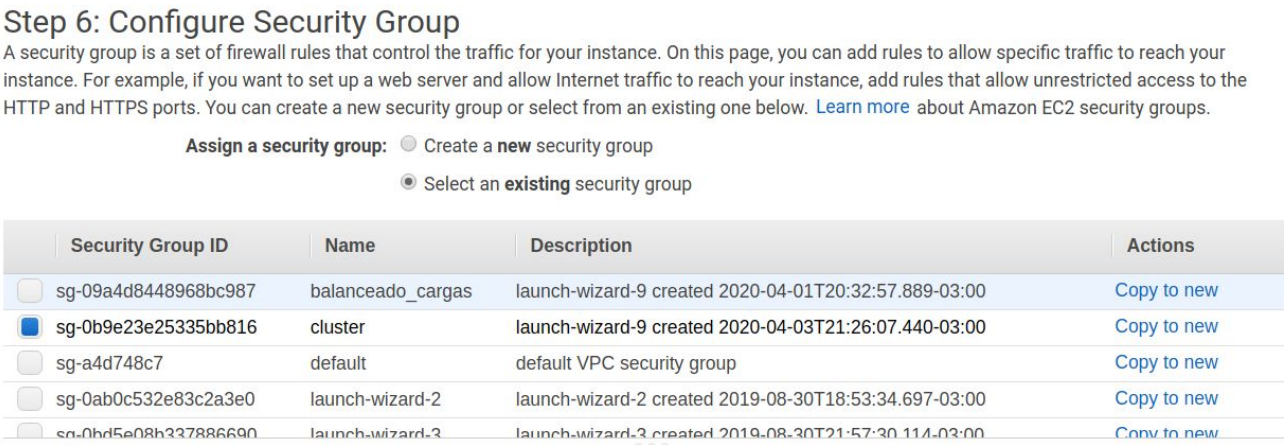
Depois de selecionar a AMI e o tipo de instância, chegamos nesta tela. **Selecione Edit security groups**



Dê um nome especial para o security group, para identificar o cluster



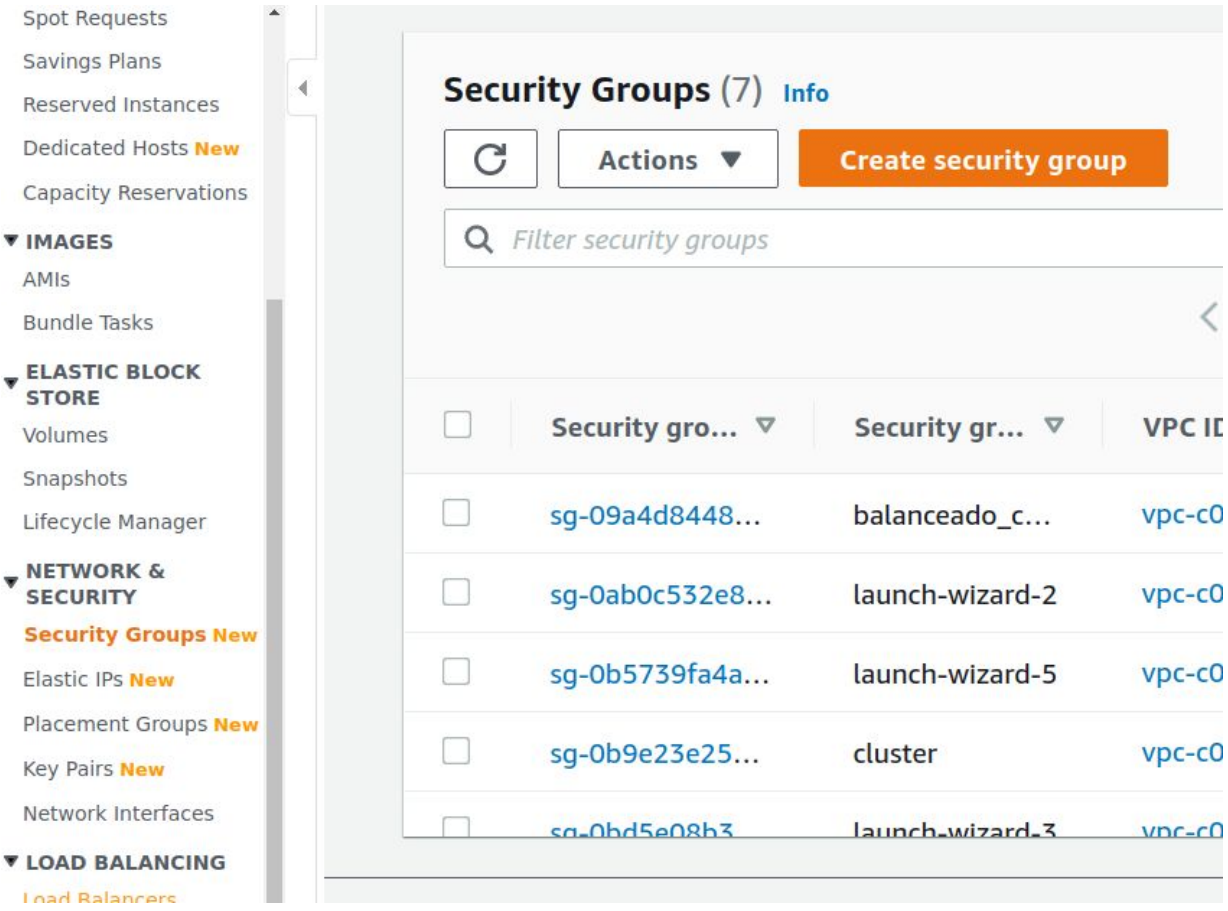
Quando estiver criando as outras máquinas do cluster, reutilize-o como mostrado na figura a seguir:



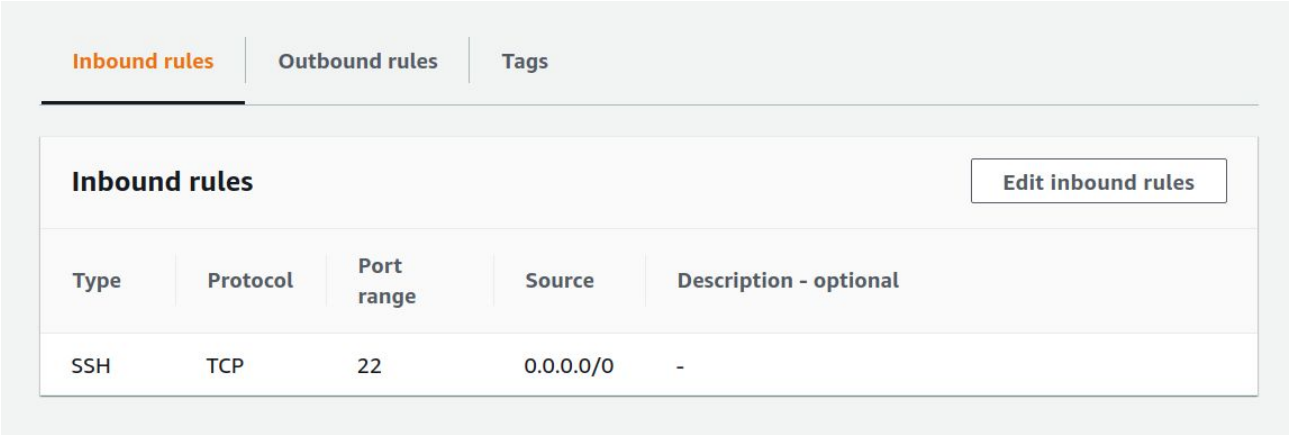
Abrindo portas TCP dentro do **security group** para o MPI

Para o MPI rodar, precisamos abrir as portas TCP livremente entre as máquinas do cluster. Como todas estão utilizando o mesmo **security group**, fazemos isso adicionando uma regra como abaixo:

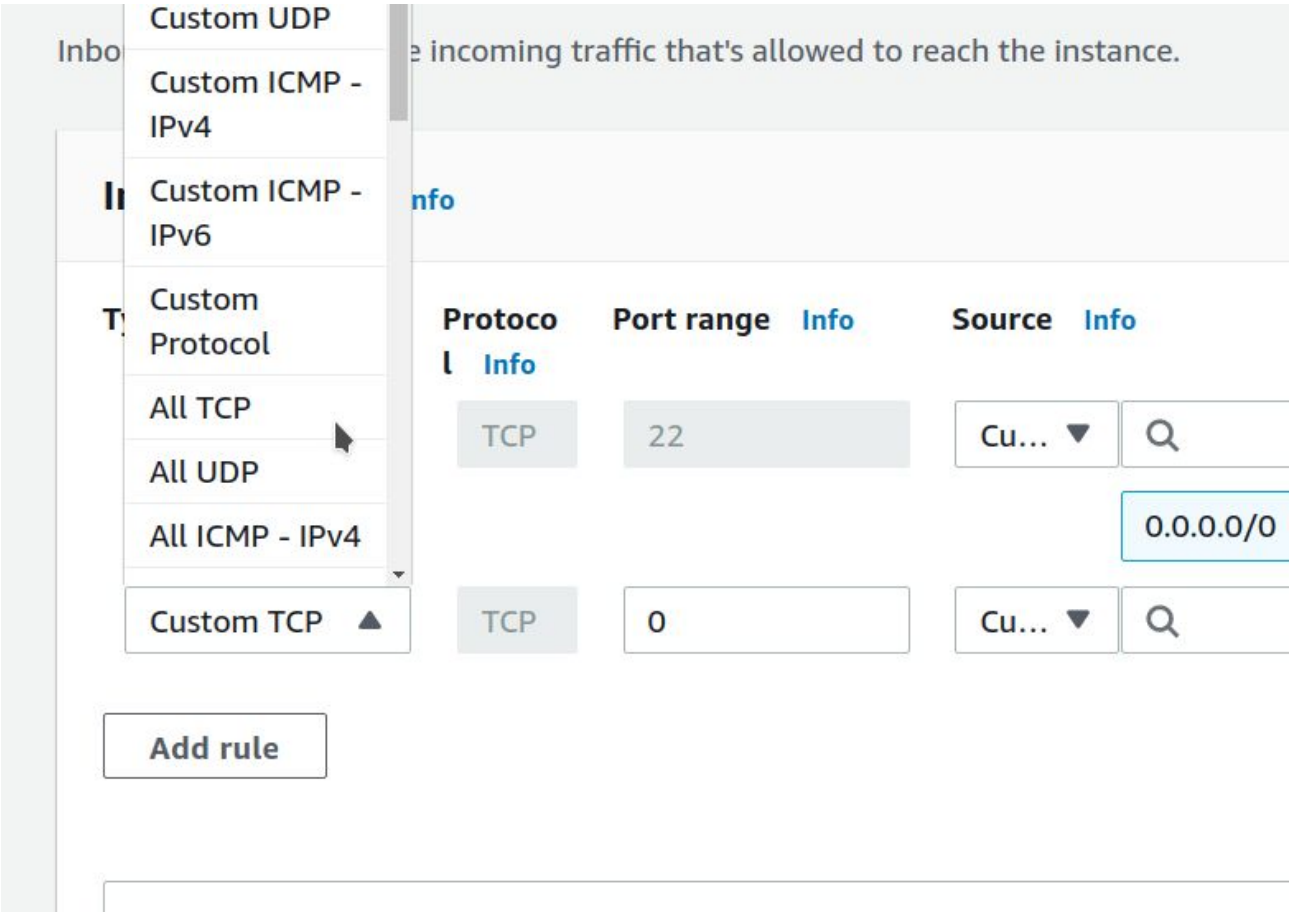
Primeiro, vamos na opção **NETWORK & SECURITY**, e selecionamos **Security Groups;** depois, selecionamos o **security group** correto



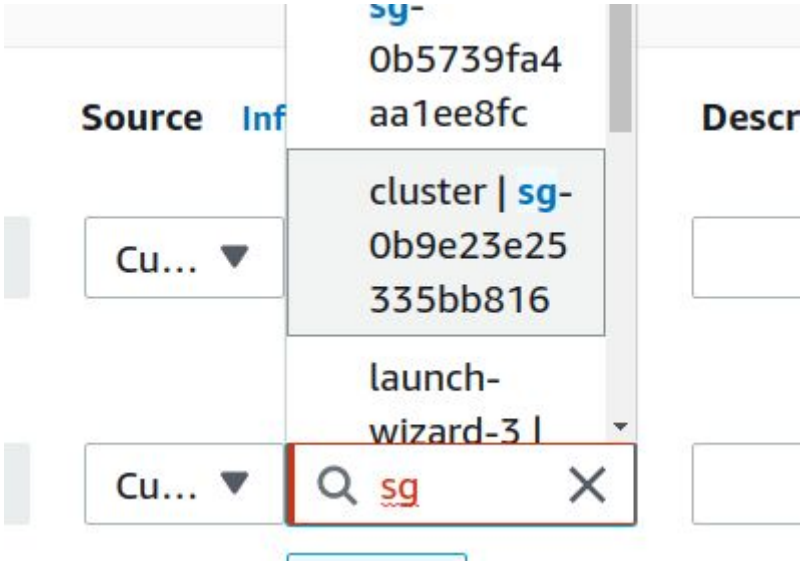
E vamos editar as regras de entrada: **Edit Inbound Rules**:



Agora, vamos Adicionar uma nova Regra **All TCP**



Depois de selecionar **All TCP** é preciso informar de onde podem vir as conexões (**Source**).
Neste campo, comece a digitar “**sg”** e vão aparecer as opções dos **security groups** para você poder selecionar:



Selecione o **security group** do seu cluster e salve as modificações.

Copiando a chave .pem para as máquinas

Quando uma máquina é criada, a AWS envia a chave **.pem** utilizada para conectar. Esta chave tem que ser enviada para cada máquina do cluster (utilize sempre a mesma chave).

Você deve utilizar o comando **scp** ou o **pscp** para fazer a cópia da chave: (o comando abaixo é em uma única linha)

```
$ scp -i ~/.ssh/mariomack.pem  ~/.ssh/mariomack.pem
ubuntu@ec2-18-223-28-95.us-east-2.compute.amazonaws.com:~/.ssh/
```

Agora precisamos adicionar nossa chave privada ao nosso agente de autenticação **ssh-agent**. **TEM QUE FAZER ISSO EM TODAS AS MÁQUINAS**

```
$ eval `ssh-agent`
$ ssh-add mariomack.pem
```

Colocar todas as máquinas no /etc/hosts utilizando o IP privado.

Edite o arquivo **/etc/hosts** adicionando estas linhas logo abaixo da linha do **localhost (1ª linha)**

```
$ sudo nano /etc/hosts

172.31.35.142 master
172.31.31.44 slave1
172.31.22.225 slave2
```

(Para finalizar, digite CRTL+O para salvar o arquivo e depois CRTL+X para sair)

Depois, da máquina master executar um comando remoto com o **ssh** em cada slave para tornar a máquina conhecida (pelo IP privado):

```
$ ssh ubuntu@slave1 hostname
The authenticity of host 'slave1 (172.31.22.225)' can't be established.
ECDSA key fingerprint is
SHA256:9YrhJaRMtgTBm/leoDCM8qhdsnnnrPPJPcd81nvoZ9o.
Are you sure you want to continue connecting (yes/no)? yes

$ ssh ubuntu@slave2 hostname
The authenticity of host 'slave2 (172.31.24.225)' can't be established.
ECDSA key fingerprint is
SHA256:9YrhJaRMtgTBm/leoDCM8qhdsnnnrPPJPcd81nvoZ9o.
Are you sure you want to continue connecting (yes/no)? yes
```

Na segunda vez que executar o comando acima, ele não deverá mais mostrar esta informação de autenticidade da máquina, etc. Vai mandar o resultado direto da execução do comando **hostname**. Você deve rodar o comando acima para todas as **slaves** do cluster.