

Criando um Classic Load Balancer e associando às instâncias

Transcrição

Nesse vídeo vamos criar um *Classic Load Balancer* a partir do AWS CLI. E para realmente testar o balanceamento vamos primeiro criar uma segunda instância com o comando `aws ec2` :

```
aws ec2 run-instances --image-id ami-8e681d98 --count 1 --instance-type t2.micro --key-name "catalogo"
```

Criação do Classic Load Balancer

Sabemos que o comando se chama de `aws` . Como queremos criar um balanceador de carga devemos usar o serviço `elb` (*Elastic Load Balancer*) como o comando `create-load-balancer` . Assim temos o comando parcial:

```
aws elb create-load-balancer
```

Além disso devemos definir o nome do load balancer:

```
aws elb create-load-balancer --load-balancer-name "aws-cli-lb"
```

Também queremos que nosso load balancer recebe requisições HTTP e para tal devemos definir um listener HTTP:

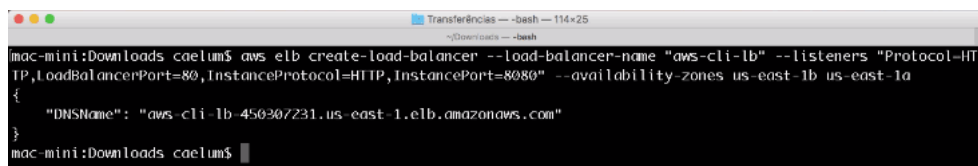
```
aws elb create-load-balancer --load-balancer-name "aws-cli-lb" --listeners "Protocol=HTTP,LoadBalancerPort=80,InstanceProtocol=HTTP,InstancePort=8080" --availability-zones us-east-1b us-east-1a
```

Repare que definimos o protocolo HTTP na porta 80 para receber as requisições e a porta 8080 para as instâncias no EC2.

O último passo é declarar as zones de disponibilidade:

```
aws elb create-load-balancer --load-balancer-name "aws-cli-lb" --listeners "Protocol=HTTP,LoadBalancerPort=80,InstanceProtocol=HTTP,InstancePort=8080" --availability-zones us-east-1b us-east-1a
```

Já podemos testar o comando que deve criar o novo balanceador de carga:



```
mac-mini:Downloads caelum$ aws elb create-load-balancer --load-balancer-name "aws-cli-lb" --listeners "Protocol=HTTP,LoadBalancerPort=80,InstanceProtocol=HTTP,InstancePort=8080" --availability-zones us-east-1b us-east-1a
{
  "DNSName": "aws-cli-lb-456307231.us-east-1.elb.amazonaws.com"
}
mac-mini:Downloads caelum$
```

E assim ver se o load balancer foi criado através da console web:

Name	DNS name	State	VPC ID	Availability Zones	Type
aws-cli-lb	aws-cli-lb-450307231.us-east-1a.elb.amazonaws.com	active	vpc-e9d6f08f	us-east-1a, us-east-1b	classic
catalogo-application-lb	catalogo-application-lb-20748.us-east-1a.elb.amazonaws.com	active	vpc-e9d6f08f	us-east-1b, us-east-1a	application

Associar o balanceador com as instâncias

Criamos duas instâncias e criamos o load balancer, falta apenas associá-los. Novamente você pode ver a documentação completa referente ao comando utilizado:

<http://docs.aws.amazon.com/elasticloadbalancing/latest/classic/elb-deregister-register-instances.html>

(<http://docs.aws.amazon.com/elasticloadbalancing/latest/classic/elb-deregister-register-instances.html>)

Para registrar instâncias com um load balancer devemos continuar usando o serviço `elb` mas agora com o comando `register-instances-with-load-balancer` :

```
aws elb register-instances-with-load-balancer
```

Agora só falta especificar o nome do nosso balanceador junto com as Ids das instâncias do EC2. O comando completo fica:

```
aws elb register-instances-with-load-balancer --load-balancer-name "aws-cli-lb" --instances i-0a1a63714866cfe8d i-0dff6cd94176d59b5
```

Executando o comando no terminal:

```
mac-mini:Downloads caelum$ aws elb register-instances-with-load-balancer --load-balancer-name "aws-cli-lb" --instances i-0a1a63714866cfe8d i-0dff6cd94176d59b5
{
  "Instances": [
    {
      "InstanceId": "i-0dff6cd94176d59b5"
    },
    {
      "InstanceId": "i-0a1a63714866cfe8d"
    }
  ]
}
```

Um vez criado o load balancer e associado as instâncias podemos testá-lo e usar o nome DNS dele para jogar no navegador.