AWS EKS - KUBERNETES

Fernando C. Menardi

GitHub Repo: https://github.com/Sargastico/ally-aws-kubernetes

Configuração do IAM

É necessário e recomendado que se crie um usuário específico para o gerenciamento EKS no AWS. Para isso, prossiga para o painel do IAM e acesse a página de "Usuários".

1) Clique em "Adicionar usuário".
2) De um nome ao novo usuário.
3) Conceda acesso "Programático" e ao "Console de Gerenciamento da AWS".
4) Na página de permissões, prossiga com "Anexar políticas existentes".
5) Pesquise e adicione as seguintes políticas:

    a. AWSQuickSightListIAM
    b. AmazonEC2FullAccess
    c. IAMFullAccess
    d. AmazonS3FullAccess
    e. AmazonSNSReadOnlyAccess
    f. AWSImageBuilderFullAccess
    g. AmazonEKSFargatePodExecutionRolePolicy

2. Será necessário criar mais duas políticas. Ainda em "Anexar políticas existentes", acesse "Criar política".
3. Na página "Criar política", acesse a opção "JSON", copie e cole o código a seguir:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": "cloudformation:*",
            "Resource": "*"
        }
    ]
}
```

1) Clique em "Revisar política" e de um nome e uma descrição à nova política.
2) Clique em criar política. Agora ela está disponível junto as politicas existentes para ser adicionada.
3) Repita os passos de 6-9 mas utilizando o código JSON a seguir:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource":
"arn:aws:iam::091481594548:role/eksServiceRole"
        }
    ]
}
```

ATENÇÃO: Todas essas políticas concedem permissões potencialmente perigosas ao sistema, caso sejam atribuídas à um usuário não confiável. Revise-as e certifique-se de que estão sendo garantidas apenas as permissões necessárias. Este exemplo está generalizado, em produção é preciso utilizar permissões mais cautelosas.

Prosseguindo para "criar usuário", você deve ser direcionado para uma página exibindo uma mensagem de "Êxito". Nessa página, faça o download das credenciais de acesso clicando no botão "Fazer download .csv".

ATENÇÃO. Essa etapa é importante, pois não será possível ter acesso a "Chave de acesso secreta" novamente. Guarde-a de forma segura e responsável.

Dentro do arquivo CSV, temos respectivamente:

```
<Nome do usuário>, <ID da Chave de acesso>, <Chave de acesso secreta>,
<Senha>, <Link para login no console>
```

Acesse o "Link para login no console" e prossiga com login e senha.

Virtual Private Cloud (VPC)

Para a criação de uma VPC utilize o serviço "CloudFormation" do AWS.

1) Acesse "Stacks"
2) "Create Stacks" > "With new resources (standard)"
3) Mantenha a opção: "Template is ready"
4) Mantenha a opção de "Amazon S3 URL" no "Template Source"

Copie e cole o seguinte link no campo "Amazon S3 URL":

```
https://amazon-eks.s3-us-west-2.amazonaws.com/cloudformation/2018-11-
07/amazon-eks-vpc-sample.yaml
```

O arquivo YAML apontato por esse link estabelece as características desejadas da VPC. Nesse caso, com três subnets:

- VPC Block: 192.168.0.0/16

- Subnet 01: 192.168.64.0/18
- Subnet 02: 192.168.128.0/18
- Subnet 03: 192.168.192.0/18

Prosseguindo com o processo, certifique-se de que os IPs foram arranjados automaticamente e de um nome para a stack.

Nas próximas duas telas não é necessário mudar nenhuma configuração.

Clique em "Create Stack", e espere até que o status da stack na página inicial fique como "create complete".

Configurações do Cluster

Para criar um cluster, acesse o serviço "EKS" no painel de serviços do AWS e clique em "Create cluster":

1) De um nome para o novo cluster.
2) Selecione a versão do kubernetes mais recente.
3) Selecione "eksServiceRole" em "Role name".
4) Selecione o VPC Block criado anteriormente em VPC.
5) Selecione todas as Subnets carregadas.
6) Selecione o "Security Group" contendo "ControlPlaneSecurity" no nome.

ATENÇÃO: Certifique-se de que o Cluster EKS está sendo criado utilizando o mesmo usuário IAM que será utilizando para administrar o cluster a partir do terminal (próximas etapas).

7) Clique em "Create" no final da página, e espere até que o status do cluster na página inicial esteja como "ACTIVE".

Configuração do Kubectl para o AWS

Para instalar o Kubectl copie e cole os comandos a seguir no terminal:

```
curl -o kubectl https://amazon-eks.s3-us-west-
2.amazonaws.com/1.10.3/2018-07-26/bin/linux/amd64/kubectl

chmod +x ./kubectl


mkdir $HOME/bin && cp ./kubectl $HOME/bin/kubectl && export
PATH=$HOME/bin:$PATH

echo 'export PATH=$HOME/bin:$PATH' >> ~/.bashrc
```

Instale também o “aws-iam-authenticator”:

```
curl -o aws-iam-authenticator https://amazon-eks.s3-us-west-
2.amazonaws.com/1.10.3/2018-07-26/bin/darwin/amd64/aws-iam-
authenticator

chmod +x ./aws-iam-authenticator
```

Instale o “aws-cli” seguindo as instruções apresentadas no link a seguir:

1. https://docs.aws.amazon.com/pt_br/cli/latest/userguide/install-cliv2-linux-mac.html

Com o “aws-cli” configurado corretamente, prossiga para a autenticação. Execute no terminal:

```
aws configure
```

Este comando irá requisitar que sejam inseridas as credenciais de acesso do usuário IAM, criado anteriormente. As credenciais estão armazenadas no arquivo CSV. Também é preciso definir a região de operação (dever ser a mesma na qual foram criados a VPC e o Cluster).

Utilize o comando:

```
aws eks update-kubeconfig --name <nome do cluster>
```

Work Nodes

Acesse novamente o “EKS” no painel de serviços do AWS. Clique no nome do cluster que foi criado anteriormente. Na sessão “Node Groups”, clique em “Add node group”.

Escolha um nome para dar ao grupo de Nodes, e uma IAM Role. Para criar uma IAM role acesse:

2. https://docs.aws.amazon.com/eks/latest/userguide/worker_node_IAM_role.html

Adicione as três Subnets, caso não tenham sido vinculadas automaticamente.

Marque a check-box quer permite o acesso remoto aos nodes.

Selecione uma chave SSH já existente no campo “SSH Key pair” (será usada para eventuais acessos diretos às instancias/nodes que compõe o cluster).

Prossiga para a tela seguinte e selecione o tipo de AMI (com GPU/sem GPU), o tipo de instância e o tamanho do disco individual para cada máquina.

NOTA: Não é possível alterar especificações de hardware da máquina depois do "Node-Group" ser criado. Para isso, é necessário apagar o grupo por completo e recriar outro com as novas especificações.

Prossiga para a próxima tela. Nessa etapa, escolha quantas instancias deverão ficar sempre ligadas, definindo o "Minimum Size". Escolha quantas deverão ser ligadas no máximo, definindo o "Maximum size" e defina com quantas máquinas o cluster deve operar inicialmente, definindo o "Desired size".

NOTA: Mesmo depois de criado, ainda é possível alterar os parâmetros de quantidade de instâncias a qualquer momento, clicando em "Edit" no campo "Node-Groups" na página de configuração do cluster.

Clique em "create" e aguarde o status do novo grupo de nodes ser alterado para "active".

Para se certificar de que tudo está funcionando corretamente, rode o seguinte comando no terminal:

```
kubectl get nodes
```

O output deve ser semelhante a este (para dois nodes em um node group):

```
NAME                                STATUS   ROLES    AGE   VERSION
ip-192-168-133-80.ec2.internal      Ready    <none>   18h   v1.14.7-eks-1861c5
ip-192-168-231-174.ec2.internal     Ready    <none>   18h   v1.14.7-eks-1861c5
```

Kubernetes Dashboard

O Kubernetes Dashboard é uma GUI para controle e gerenciamento do kubernetes. Para instalar execute os seguintes comandos:

```
kubectl create –f
https://raw.githubusercontent.com/kubernetes/dashboard/v2.0.0-
rc4/aio/deploy/recommended.yamlhttps://raw.githubusercontent.com/kuber
netes/dashboard/v2.0.0-rc4/aio/deploy/recommended.yaml

kubectl proxy &
```

Para se autenticar no dashboard é necessário ter um token. O token expira relativamente rápido e força o usuário a realizar uma nova autenticação. Para gerar um token execute o script >> ./dashboard/commandToken.sh

Helm - Gerenciador de Pacotes

Para instalar o helm, execute o script >> ./helm/get_helm.sh

Execute o comando a seguir no diretório "/helm" :

```
kubectl apply -f tiller-rbac.yaml
```

Utilize o "Tiller" como conta de serviço:

```
helm init --service-account tiller
```

Prometheus - Monitoramento do sistema

Antes de prosseguir com a instalação do "Prometheus", vamos instalar o "Metrics Server". Execute:

```
helm install stable/metrics-server --name metrics-server --version
2.0.4 --namespace metrics`

kubectl get apiservice v1beta1.metrics.k8s.io -o yaml
```

NOTA: Sempre defina um novo namespace para cada instalação para que manter as instalações organizadas.

O "Prometheus" necessita de espaço para armazenamento, para criar uma nova "StorageClasse", execute o comando a seguir no diretório "/prometheus":

```
kubectl create -f prometheus-storageclass.yaml
```

Será necessário realizar mudanças no arquivo de configuração do prometheus. Com algum editor de código abra: "/prometheus/prometheus-values.yaml"

Na linha 726, insira dentro do array "externalIPs", o IPv4 público da instancia que irá expor o serviço do para rede na porta 30900. Salve a mudança.

Para instalar o Prometheus:

```
helm install -f prometheus-values.yaml stable/prometheus --name
prometheus --namespace prometheus
```

Para verificar se todos os componentes do prometheus estão funcionando como o esperado:

```
kubectl get all -n prometheus
```

Grafana – Interface GUI de Supervisão

Antes de realizar o deploy da grafana no cluster, é precisso alterar a senha do painel administrativo no arquivo "/grafana/grafana-values.yaml".

Abra o arquivo com algum editor de código e vá para a linha 118.

Na linha 118 e 119, defina o nome de usuário e a senha do administrador:

```
118 adminUser: admin
119 adminPassword: StrongPassword # Use a strong password here!
```

Verifique se todos os componentes da grafana estão funcionando corretamente rodando:

```
kubectl get all -n grafana
```

Para obter a URL de acesso ao painel de login do sistema, rode o script >> ./grafana/grafanaURL