

IBM开源技术 Istio训|练营

2019.1.13

Iris Ding (dingsj@cn.ibm.com)

Istio训练营上手实操内容

- 在k8s上安装Istio
- guestbook应用程序介绍
- 部署guestbook 相关微服务应用程序
- 使用Istio进行微服务流量
- 使用Istio来监控你的微服务

训练营的基本环境

- IKS

训练营的基本环境 -IKS

- 一个 IBM Cloud Kubernetes Service Cluster
- 所需的软件**与工具**
 - 使用您喜爱的终端，比如， iTerm2

注册并登录IBM Cloud，创建 kubernetes cluster

- <https://console.bluemix.net/>, 注册Trial, Pay-As-You-Go, 或者 Subscription 账户
- 选择Catalog/Compute/Kubernetes Service
- 点击create，选择 Free
- 取一杯coffee 时间(< 2 分钟)，等待cluster部署完成(显示为正常状态)



The screenshot shows the 'Create new cluster' form on the IBM Cloud website. The steps are as follows:

- Resource Group: default
- Cluster type: Free (selected)
- Cluster name: mycluster
- Create Cluster button

Additional text on the page states: 'New to Kubernetes? Create a cluster with 1 worker node to explore the capabilities.' and 'Learn more about the differences between Free and Standard clusters in our [docs](#)'.

下载kubernetes配置文件访问你的IKS cluster

- 下载 <https://kubernetes.io/docs/tasks/tools/install-kubectl/#install-kubectl>

E.g., Ubuntu:

```
sudo apt-get update && sudo apt-get install -y apt-transport-https  
curl -s https://packages.cloud.google.com/apt/doc/apt-key.gpg | sudo apt-key add -  
echo "deb http://apt.kubernetes.io/ kubernetes-xenial main" | sudo tee -a /etc/apt/sources.list.d/kubernetes.list  
sudo apt-get update  
sudo apt-get install -y kubectl
```

- 下载并解压目标cluster的配置及证书文件

- export 环境变量KUBECONFIG

- export
KUBECONFIG=/Users/iris/oti/DevAdvocacy/istioworkshopBeijing/kubeConfig146275655/ku
be-config-mel01-istiocluster.yml

- 查看cluster 信息

- kubectl get node
 - kubectl get svc,deploy,po —all-namespaces

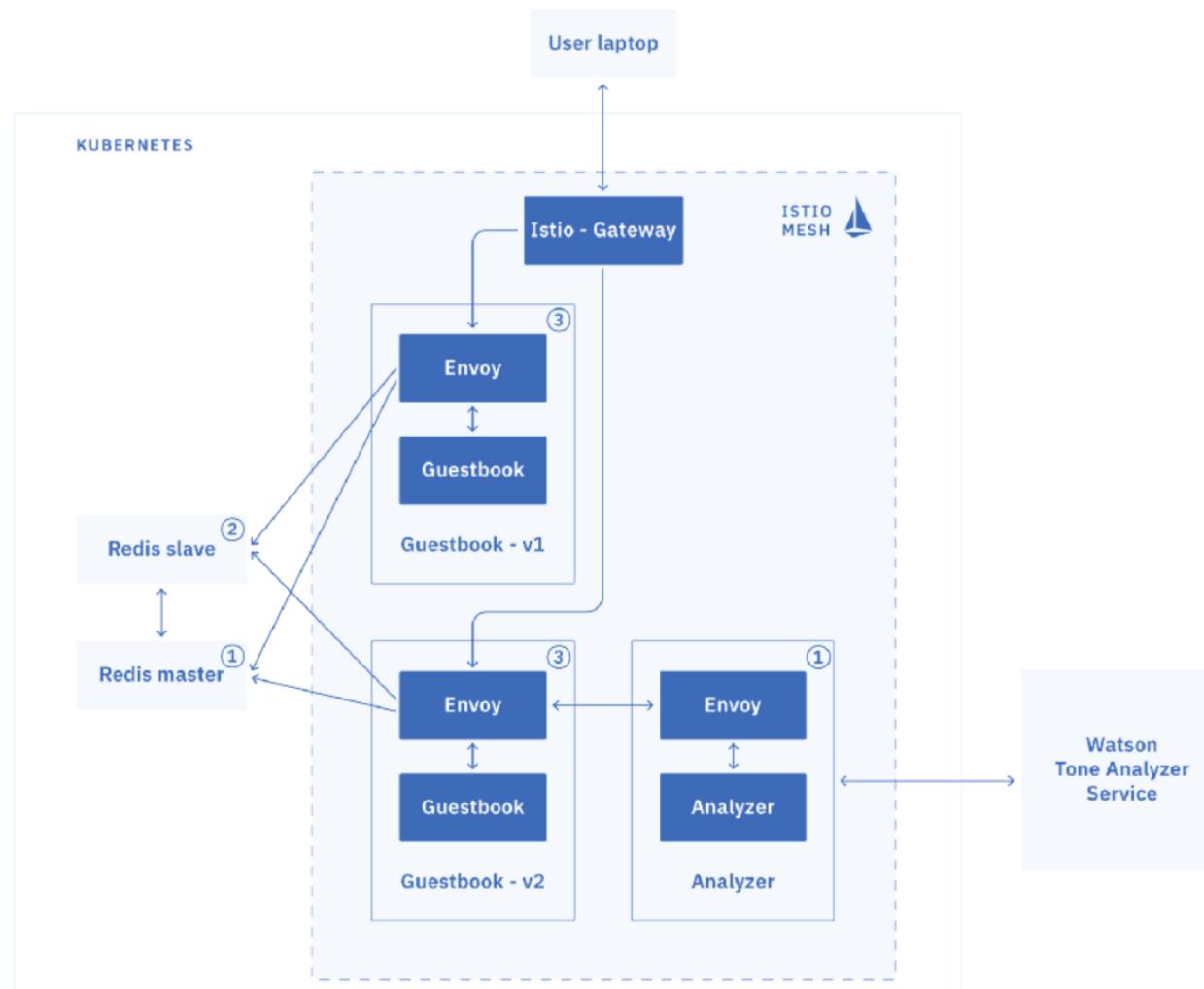
部署Istio

- 下载Istio
 - curl -L https://git.io/getLatestIstio | sh -
- 配置 Istioctl CLI
 - export PATH=\$PWD/istio-1.0.5/bin:\$PATH
- 安装
 - kubectl apply -f install/kubernetes/helm/istio/templates/crds.yaml
 - kubectl apply -f install/kubernetes/istio-demo.yaml
- 验证
 - kubectl get pods –n istio-system
 - kubectl get svc -n istio-system

获取 istio101 workshop 代码 和 guestbook 代码

- istio101
 - git clone <https://github.com/IBM/istio101>
- guestbook
 - git clone <https://github.com/IBM/guestbook.git>

Guestbook应用程序



<https://github.com/IBM/guestbook>

部署guestbook

1. 部署redis

- cd guestbook/v2
- kubectl create -f redis-master-deployment.yaml
kubectl create -f redis-master-service.yaml
kubectl create -f redis-slave-deployment.yaml
kubectl create -f redis-slave-service.yaml

2. 验证 redis 成功部署

- kubectl get pods
- kubectl get svc
- kubectl get deployment

3. 手动注入side-car,并安装guestbook

- kubectl apply -f <(istioctl kube-inject -f ..v1/guestbook-deployment.yaml)
- kubectl apply -f <(istioctl kube-inject -f guestbook-deployment.yaml)

4. 创建guestbook服务

- kubectl create -f guestbook-service.yaml
- Verify that the service was created.
- kubectl get svc
- Verify that the pods are up and running.
- kubectl get pods

- 使用Istio Service Entry 和 Istio Virtual Service 配置guestbook访问IKS Cluster 之外的 Watson Tone Analyzer服务
 - cd istio101/workshop/plans
 - 添加 Service Entry 和 Virtual Service
 - Kubectl apply –f serviceentry-tone.yaml

部署Analyzer

1. <https://cloud.ibm.com/catalog>->AI->Tone Analyzer->:

1. Service Name: my-tone-analyzer-service
2. Location: Sydney -> create

2. 创建证书

1. Name: tone-analyzer-key
2. Role: manager
3. Add Inline Configuration Parameters (Optional)::
{"instance-name ":"my-tone-analyzer-service"}
4. copy the apikey and url

3. 编辑 guestbook/v2/analyzer-deployment.yaml

```
env:  
  - name: VCAP_SERVICES_TONE_ANALYZER_API_KEY  
    value:  
  - name : VCAP_SERVICES_TONE_ANALYZER_TOKEN_ADDRESS  
    value: "https://iam.bluemix.net/identity/token"  
  - name: VCAP_SERVICES_TONE_ANALYZER_SERVICE_API  
    value:
```

4. 部署Analyzer

- kubectl apply -f <(istioctl kube-inject -f analyzer-deployment.yaml)
- kubectl apply -f analyzer-service.yaml

Analyzer 以apikey与 创建的Watson Tone Analyzer 服务认证通讯以帮助分析用户文字中的语气

5. 创建Egress rule:

```
cd ../../istio101/workshop/plans  
kubectl apply -f analyzer-egress.yaml
```

使用Istio对guestbook进行流量管控

- 使用Istio Ingress gateway 暴露guestbook服务
 - cd istio101/workshop/plans
 - kubectl create -f guestbook-gateway.yaml
- 获取 Node Port 以及 External IP
 - kubectl get service istio-ingressgateway -n istio-system
- 把所有流量导入v1版本的guestbook
 - cd istio101/workshop/plans/
 - kubectl replace -f virtualservice-all-v1.yaml
 - kubectl create -f guestbook-destination.yaml
- 把来自firefox的流量导入v2版本，其他流量导入v1版本
 - kubectl replace -f virtualservice-test.yaml
- 把80%的流量导入v1版本，20%的流量导入v2版本
 - kubectl replace -f virtualservice-80-20.yaml

观察guestbook遥测数据 (Telemetry) cont'd

- 验证 Grafana, Prometheus, ServiceGraph and Jaeger 已经被成功安装
 - `kubectl get pods -n istio-system` `kubectl get services -n istio-system`
- 创建规则并配置Istio自动接受监控数据
 - `cd istio101/workshop/plans`
 - `kubectl create -f guestbook-telemetry.yaml`
- 访问guestbook来产生一些工作负载 (External IP, Node Port).
 - `while sleep 0.5; do curl http://<External IP>:<Node Port>/; done`

观察guestbook遥测数据 (Telemetry)

- 在laptop上运行

- Jaeger

- `kubectl port-forward -n istio-system $(kubectl get pod -n istio-system -l app=jaeger -o jsonpath='{.items[0].metadata.name}') 16686:16686 &`

- Grafana

- `kubectl -n istio-system port-forward $(kubectl -n istio-system get pod -l app=grafana -o jsonpath='{.items[0].metadata.name}') 3000:3000 &`

- Prometheus

- `kubectl -n istio-system port-forward $(kubectl -n istio-system get pod -l app=prometheus -o jsonpath='{.items[0].metadata.name}') 9090:9090 &`
 - 访问 <http://localhost:9090/graph>

- Service Graph

- `kubectl -n istio-system port-forward $(kubectl -n istio-system get pod -l app=servicegraph -o jsonpath='{.items[0].metadata.name}') 8088:8088 &`
 - 浏览 <http://localhost:8088/dotviz>

欢迎大家继续关注IBM开源技术微讲堂的活动

- 每周四晚8点Istio系列
- 本年度其他的微讲堂活动
- 请使用 bluemix.net 注册IBM Cloud使用账号