**Getting Started with Istio: Install Istio with Helm in 2024**



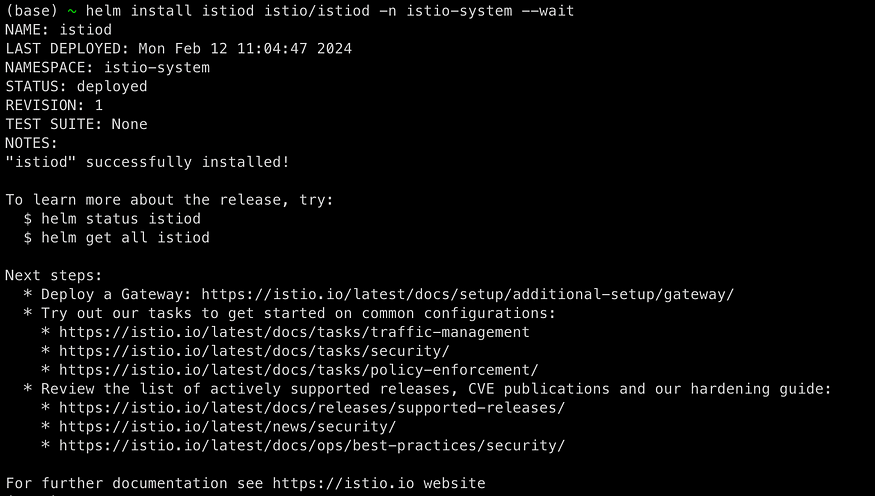
**Prerequisites**

1. [Install the Helm client](https://helm.sh/docs/intro/install/), version 3.6 or above.
2. A Kubernetes Cluster, I am using GKE.
3. [Install Kubectl Client Tool](https://kubernetes.io/docs/tasks/tools/install-kubectl-macos/), version 1.26.1 or above.

**Installation steps**

**Install with Helm**

helm repo add istio https://istio-release.storage.googleapis.com/charts  
helm repo update  
kubectl create namespace istio-system  
helm install istio-base istio/base -n istio-system --set defaultRevision=default  
helm ls -n istio-system  
helm install istiod istio/istiod -n istio-system --wait  
helm ls -n istio-system  
kubectl get deployments -n istio-system --output wide  
kubectl create namespace istio-ingress  
helm install istio-ingress istio/gateway -n istio-ingress --wait





**Deploy the sample application**

1. **Deploy the**Bookinfo sample application

kubectl label namespace default istio-injection=enabled  
  
kubectl apply -f https://raw.githubusercontent.com/istio/istio/release-1.20/samples/bookinfo/platform/kube/bookinfo.yaml  
kubectl get services  
kubectl get pods

**2. Verify everything is working correctly up to this point. Run this command to see if the app is running inside the cluster and serving HTML pages by checking for the page title in the response**

kubectl exec "$(kubectl get pod -l app=ratings -o jsonpath='{.items[0].metadata.name}')" -c ratings -- curl -sS productpage:9080/productpage | grep -o "<title>.\*</title>"

https://miro.medium.com/v2/resize:fit:875/1*kkghMjG8M-HR-r-aPKDd_Q.png

**Open the application to outside traffic**

**The Bookinfo application is deployed but not accessible from the outside. To make it accessible, you need to create an**[Istio Ingress Gateway](https://istio.io/latest/docs/concepts/traffic-management/#gateways)**, which maps a path to a route at the edge of your mesh.**

**Associate this application with the Istio gateway**

kubectl apply -f bookinfo-gateway.yaml

bookinfo-gateway.yaml

apiVersion: networking.istio.io/v1alpha3  
kind: Gateway  
metadata:  
 name: bookinfo-gateway  
spec:  
 # The selector matches the ingress gateway pod labels.  
 # If you installed Istio using Helm following the standard documentation, this would be "istio=ingress"  
 selector:  
 istio: ingress # use istio default controller  
 servers:  
 - port:  
 number: 8080  
 name: http  
 protocol: HTTP  
 hosts:  
 - "\*"  
---  
apiVersion: networking.istio.io/v1alpha3  
kind: VirtualService  
metadata:  
 name: bookinfo  
spec:  
 hosts:  
 - "\*"  
 gateways:  
 - bookinfo-gateway  
 http:  
 - match:  
 - uri:  
 exact: /productpage  
 - uri:  
 prefix: /static  
 - uri:  
 exact: /login  
 - uri:  
 exact: /logout  
 - uri:  
 prefix: /api/v1/products  
 route:  
 - destination:  
 host: productpage  
 port:  
 number: 9080

**Determining the ingress IP and ports**

kubectl get svc -n istio-ingress  
  
NAME TYPE CLUSTER-IP EXTERNAL-IP PORT(S) AGE  
istio-ingress LoadBalancer 10.80.4.249 34.151.121.69 15021:31251/TCP,80:30600/TCP,443:30981/TCP 3m52s  
  
export INGRESS\_HOST=34.151.121.69  
export INGRESS\_PORT=80  
export GATEWAY\_URL=$INGRESS\_HOST:$INGRESS\_PORT

**Verify external access**

echo "http://$GATEWAY\_URL/productpage"

**View the dashboard**

Istio integrates with [several](https://istio.io/latest/docs/ops/integrations) different telemetry applications. These can help you gain an understanding of the structure of your service mesh, display the topology of the mesh, and analyze the health of your mesh.

Use the following instructions to deploy the [Kiali](https://istio.io/latest/docs/ops/integrations/kiali/) dashboard, along with [Prometheus](https://istio.io/latest/docs/ops/integrations/prometheus/), [Grafana](https://istio.io/latest/docs/ops/integrations/grafana), and [Jaeger](https://istio.io/latest/docs/ops/integrations/jaeger/).

1. Kiali

kubectl apply -f https://raw.githubusercontent.com/istio/istio/release-1.20/samples/addons/kiali.yaml

2. Prometheus

kubectl apply -f https://raw.githubusercontent.com/istio/istio/release-1.20/samples/addons/prometheus.yaml

3. Jaeger

kubectl apply -f https://raw.githubusercontent.com/istio/istio/release-1.20/samples/addons/jaeger.yaml

4. Grafana

kubectl apply -f https://raw.githubusercontent.com/istio/istio/release-1.20/samples/addons/grafana.yaml

Access the Kiali dashboard.

istioctl dashboard kiali

To see trace data, you must send requests to your service. The number of requests depends on Istio’s sampling rate and can be configured using the [Telemetry API](https://istio.io/latest/docs/tasks/observability/telemetry/). With the default sampling rate of 1%, you need to send at least 100 requests before the first trace is visible. To send 100 requests to the productpage service, use the following command:

for i in $(seq 1 100); do curl -s -o /dev/null "http://$GATEWAY\_URL/productpage"; done

