

Tracing distributed microservices via Kiali dashboard

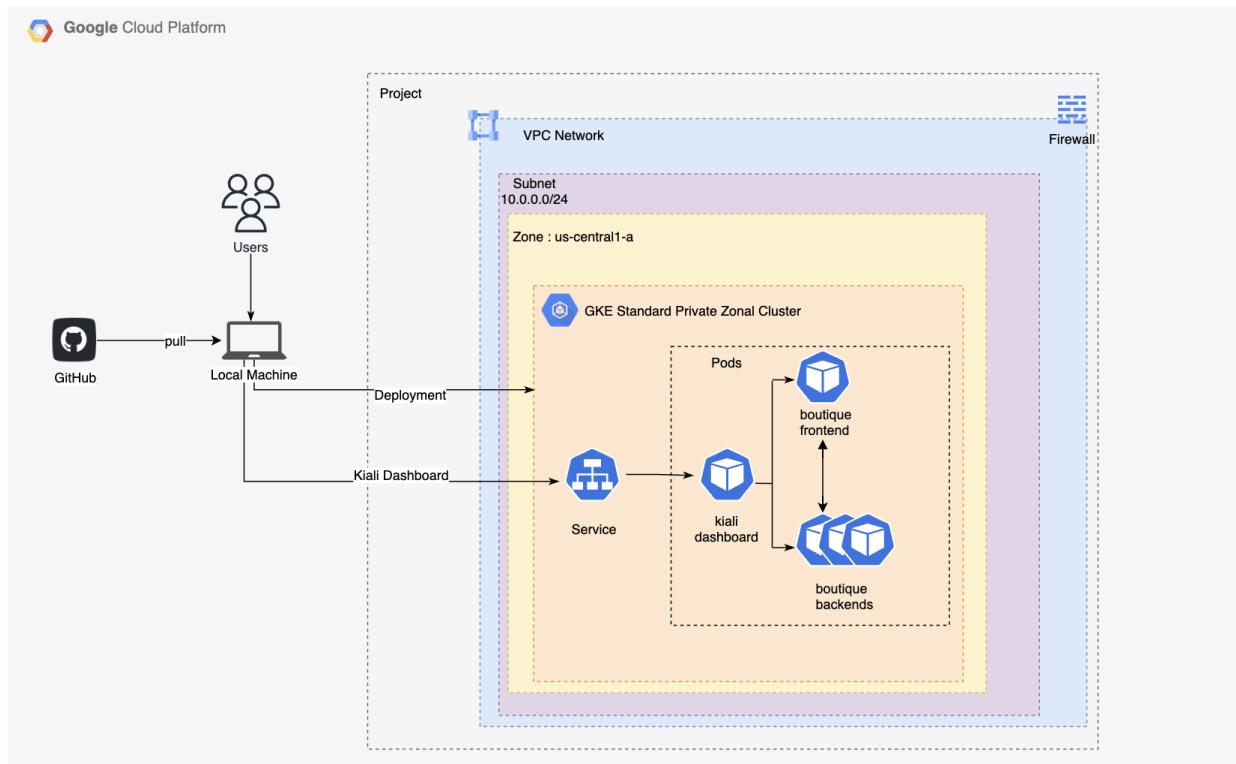


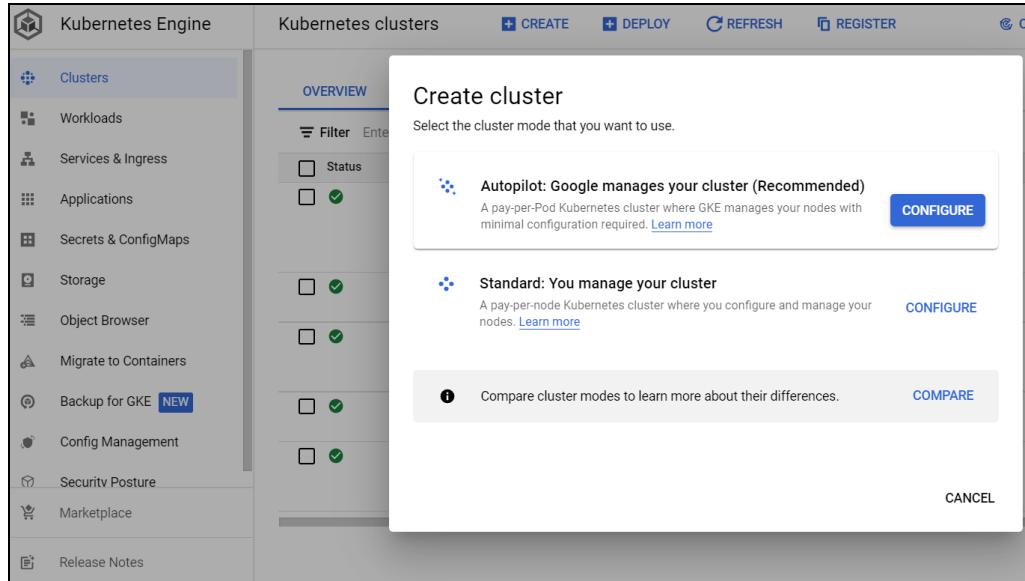
Figure: Lab_7 - Tracing Microservice using Istio Kiali dashboard

These steps require you to have a cluster running a supported version of Kubernetes(GKE).

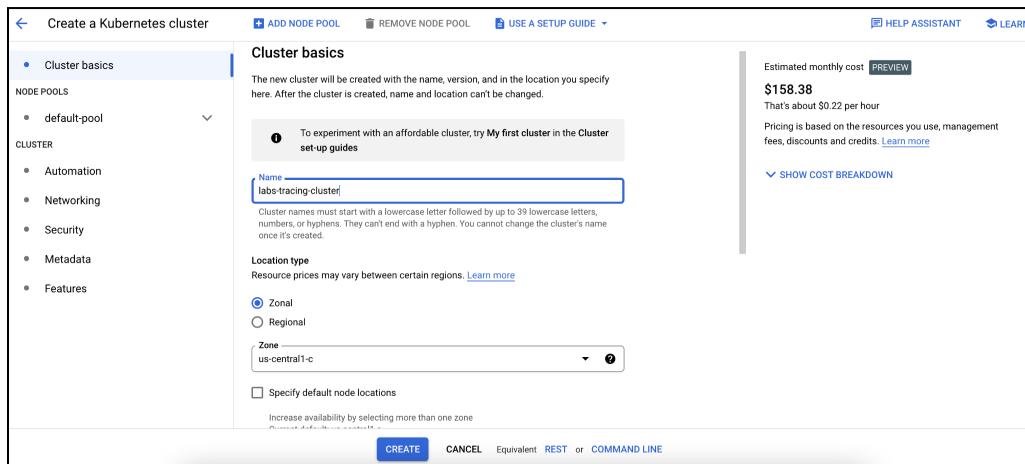
Enable the Kubernetes Engine API - [Kubernetes Engine API](#)

Follow these steps to get started with Istio:

- Create Google Kubernetes Engine Cluster
 - In the Cloud Console, on the **Navigation menu** (≡), click **Kubernetes Engine > Cluster**.
 - To create a new cluster, click **CREATE**.
 - A popup window will appear to select the cluster mode, select **Standard** mode.



- On Cluster basics, Enter the **Name** of cluster. Select location type as **Zonal** and in zone select **us-central1** from dropdown.



- Rest of the fields can be left with default values.

The screenshot shows the 'Create a Kubernetes cluster' interface. On the left, a sidebar lists 'Cluster basics', 'NODE POOLS' (with 'default-pool' selected), and 'CLUSTER' sections containing 'Automation', 'Networking', 'Security', 'Metadata', and 'Features'. The main panel has several configuration sections:

- Specify default node locations**: A checkbox with a note about increasing availability by selecting more than one zone, current default: us-central1-c.
- Control plane version**: A section where users can choose whether to upgrade manually or let GKE do it automatically. It includes 'Static version' (described as manually managing upgrades) and 'Release channel' (selected, described as letting GKE automatically manage). Below these are dropdown menus for 'Release channel' (set to 'Regular channel (default)') and 'Version' (set to '1.23.12-gke.100 (default)'). A note at the bottom states that these versions have passed internal validation and are considered production-quality.
- Buttons at the bottom**: 'CREATE' (highlighted in blue), 'CANCEL', 'Equivalent REST or COMMAND LINE'.

- Click on default-pool. In Size, Enter **Number of nodes** to 1.

The screenshot shows the 'Create a Kubernetes cluster' interface with the 'default-pool' node pool selected in the sidebar. The main panel displays 'Node pool details' for the 'default-pool':

- Name**: A text input field containing 'default-pool'.
- Control plane version**: Set to '1.23.12-gke.100'.
- Size**: A text input field containing '1'.
- Other settings**: Includes 'Compact placement' (unchecked), 'Beta' (radio button selected), 'Enable cluster autoscaler' (unchecked), and 'Specify node locations' (checkbox unchecked).
- Buttons at the bottom**: 'CREATE' (highlighted in blue), 'CANCEL', 'Equivalent REST or COMMAND LINE'.

- In the default-pool sub section, Click on **Nodes**. Select the machine type for your cluster.

The screenshot shows the 'Create a Kubernetes cluster' interface. On the left, a sidebar lists 'Cluster basics', 'NODE POOLS' (with 'default-pool' selected), and 'CLUSTER' sections. The main area is titled 'Configure node settings' and contains the following fields:

- Image type:** Container-Optimized OS with containerd (cos_containerd) (default)
- Machine configuration:**
 - Machine family:** GENERAL-PURPOSE
 - Series:** E2
 - CPU platform selection based on availability**
- Machine type:** e2-standard-4 (4 vCPU, 16 GB memory)

	vCPU 4	Memory 16 GB
--	-----------	-----------------

At the bottom right are 'CREATE', 'CANCEL', 'Equivalent REST or COMMAND LINE' buttons.

- Select **Series** as **E2** machine type as **e2-standard-4**. In Boot disk size enter **30 GB**. Boot disk size can be decided based on application requirements.
- Select the checkbox **Enable nodes on spot VMs**, spot VM's are available for only 24 hours post that they are terminated. By enabling the spot vm option, the total cost of cluster is reduced.

The screenshot shows the 'Create a Kubernetes cluster' interface with more detailed node configuration options. The sidebar and basic settings are similar to the previous screenshot. The main area includes:

- Series:** E2
- Machine type:** e2-standard-4 (4 vCPU, 16 GB memory)
- CPU PLATFORM AND GPU:**
 - Boot disk type:** Standard persistent disk
 - Boot disk size (GB):** 100
- Encryption:** Options for Google-managed encryption key (selected) and Customer-managed encryption key (CMEK).
- Local SSD disks:** A field with a question mark icon.
- Enable nodes on spot VMs:** A checked checkbox with a question mark icon.

At the bottom right are 'CREATE', 'CANCEL', 'Equivalent REST or COMMAND LINE' buttons.

- In **Networking**, leave with default values.

Node networking

These node networking settings will be used when new nodes are created using this node pool.

The cluster settings specify a maximum of 110 Pods per node, but you can override that setting at the node pool level.

Maximum Pods per node: 110

Mask for Pod address range per node: /24

Network tags

Node Pool Pod Address Range

The cluster settings specify a default cluster level pod address range, but you can override that setting at the node pool level.

Automatically create secondary ranges

Pod secondary CIDR range

- In **Security**, In Identity defaults select the Service Account created previously from the dropdown i.e. **labs-gke-sa**.

Node security

These node security settings will be used when new nodes are created using this node pool.

Identity defaults

Specify the default identity for new auto-provisioned node pools using either a service account or one or more scopes. [Learn more](#)

Service account: labs-gke-sa

The service account is used to call Google Cloud APIs. Use the default service account if available.

Access scopes

Access scopes are permanent. Select the type and level of API access to grant the VM. [Learn more](#)

Use IAM roles with service accounts to control VM access. [Learn more](#)

Enable sandbox with gVisor

Shielded options

Enable integrity monitoring

Enable secure boot

- In **Metadata**, leave with default values.

[← Create a Kubernetes cluster](#) [+ ADD NODE POOL](#) [REMOVE NODE POOL](#) [USE A SETUP GUIDE ▾](#)

<ul style="list-style-type: none"> ● Cluster basics <p>NODE POOLS</p> <ul style="list-style-type: none"> ● default-pool <ul style="list-style-type: none"> ● Nodes ● Networking ● Security <ul style="list-style-type: none"> ● Metadata <p>CLUSTER</p> <ul style="list-style-type: none"> ● Automation ● Networking ● Security ● Metadata ● Features 	<h3>Node metadata</h3> <p>These node metadata settings will be used when new nodes are created using this node pool.</p> <h4>Kubernetes labels</h4> <p>Use Kubernetes labels to control how workloads are scheduled to your nodes. Labels are applied to all nodes in this node pool and cannot be changed once the cluster is created.</p> <p>+ ADD KUBERNETES LABEL</p> <h4>Node taints</h4> <p>A node taint lets you mark a node so that the scheduler avoids or prevents using it for certain Pods. Node taints can be used with tolerations to ensure that Pods aren't scheduled onto inappropriate nodes. Learn more</p> <p>+ ADD TAINT</p> <h4>GCE instance metadata</h4> <p>Use Kubernetes labels to control how workloads are scheduled to your nodes. Labels are applied to all nodes in this node pool and cannot be changed once the cluster is created. Learn more</p> <p>Key 1 * disable-legacy-endpoints Value 1 * true</p> <p>This key disables legacy Compute Engine metadata endpoints and is added by default to all new clusters. Learn more</p>
--	--

[CREATE](#) [CANCEL](#) Equivalent [REST](#) or [COMMAND LINE](#)

- In **Networking**, Select checkbox networks in this project. In **Network**, select the network created previously **labs-vpc** from the dropdown. In the **Node subnet**, select the subnet created previously **labs-subnet**. In **Network access**, select the checkbox **Private Cluster**.

[← Create a Kubernetes cluster](#) [+ ADD NODE POOL](#) [REMOVE NODE POOL](#) [USE A SETUP GUIDE ▾](#)

<ul style="list-style-type: none"> ● Cluster basics <p>NODE POOLS</p> <ul style="list-style-type: none"> ● default-pool <ul style="list-style-type: none"> ● Nodes ● Networking ● Security ● Metadata <p>CLUSTER</p> <ul style="list-style-type: none"> ● Automation 	<h3>Networking</h3> <p>Define how applications in this cluster communicate with each other and with the Kubernetes control plane, and how clients can reach them.</p> <p><input checked="" type="radio"/> Networks in this project <input type="radio"/> Networks shared with me (from host project: searce-playground-host-project)</p> <p>Network * labs-vpc</p> <p>Node subnet * labs-subnet</p> <p>⚠ Choose a network that has subnetworks in the us-central1 region. To use this network, choose a different region.</p> <h3>Network access</h3> <p>Choose the type of network you want to allow to access your cluster's workloads. Learn more</p> <p><input type="radio"/> Public cluster Choose a public cluster to configure access from public networks to the cluster's workloads. Routes aren't created automatically. You cannot change this setting after the cluster is created.</p> <p><input checked="" type="radio"/> Private cluster Choose a private cluster to assign internal IP addresses to Pods and nodes. This isolates the cluster's workloads from public networks. You cannot change this setting after the cluster is created.</p> <p><input checked="" type="checkbox"/> Access control plane using its external IP address ?</p>
---	---

[CREATE](#) [CANCEL](#) Equivalent [REST](#) or [COMMAND LINE](#)

- Enable the checkbox, **Access control plane using its external IP address** and **Enable Control plane global access**. In control plane IP range enter **172.16.0.0/28**. Disable checkbox **Automatically create Secondary ranges** & select pod and service.

Create a Kubernetes cluster

CLUSTER

- Automation
- Networking**
- Security
- Metadata
- Features

NODE POOLS

- default-pool
 - Nodes
 - Networking
 - Security
 - Metadata

Private cluster

Choose a private cluster to assign internal IP addresses to Pods and nodes. This isolates the cluster's workloads from public networks. You cannot change this setting after the cluster is created.

Access control plane using its external IP address [?](#)

Enable Control plane global access [?](#)

Control plane IP range *
172.16.0.0/28
Example: 172.16.0.0/28

Disable Default SNAT [?](#)

Advanced networking options

Enable VPC-native traffic routing (uses alias IP) [?](#)

Automatically create secondary ranges [?](#)

Pod secondary CIDR range *
pod (10.91.0.0/22)

Maximum Pods per node
110

Mask for Pod address range per node: /24

Services secondary CIDR range *
svc (10.90.0.0/22)

CREATE CANCEL Equivalent REST or COMMAND LINE

- Select the checkbox, enable **control plane authorized network** to connect to your cluster from local machine or vm. In Authorized Networks, Enter **Name** and In Network enter your **public-ip/32**. Click on **DONE**.

Create a Kubernetes cluster

CLUSTER

- Automation
- Networking**
- Security
- Metadata
- Features

NODE POOLS

- default-pool
 - Nodes
 - Networking
 - Security
 - Metadata

Enable Dataplane V2 [?](#)

⚠ GKE Dataplane V2 has been certified to run up to 500 nodes per cluster, including node autoscaling and surge upgrades. You may request a cluster size of up to 1000 nodes by filing a Support ticket with GCP. For more information, please see [this page](#).

Enable Kubernetes Network Policy [?](#)

Enable Intranode visibility [?](#)
Reveals your intranode traffic to Google's networking fabric. To get logs, you need to enable VPC flow logs in the [selected subnetwork](#).

Enable NodeLocal DNSCache [?](#)

Enable HTTP load balancing [?](#)

Enable subsetting for L4 internal load balancers [?](#)

Enable control plane authorized networks [?](#)

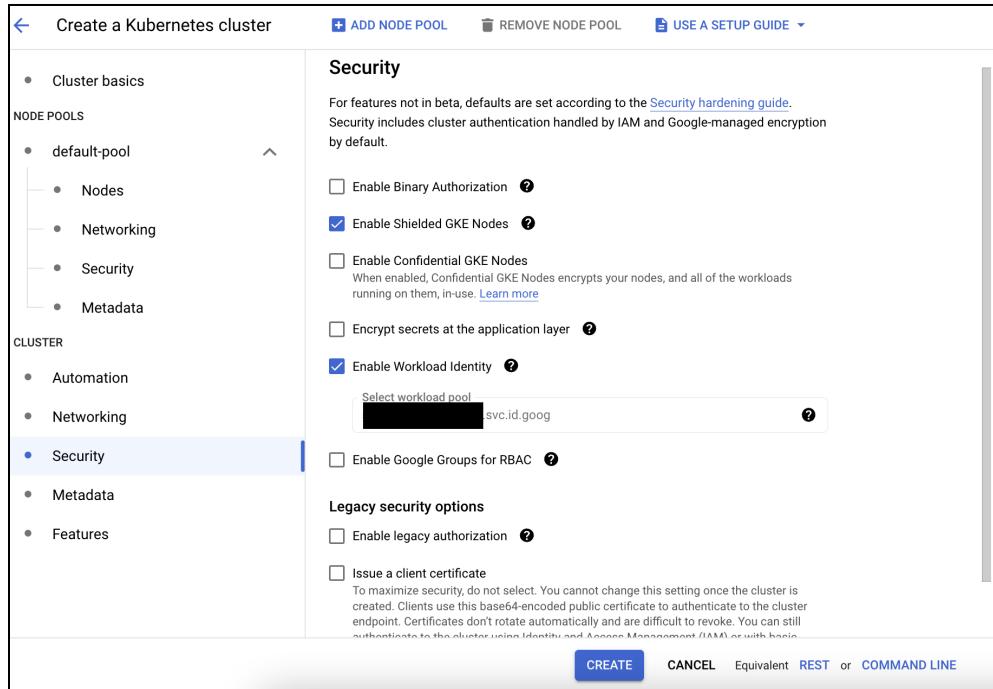
⚠ If you plan to access your cluster using kubectl (or Kubernetes API) from outside Google Cloud Platform, you must add at least one authorized network

Authorized networks

ADD AUTHORIZED NETWORK

CREATE CANCEL Equivalent REST or COMMAND LINE

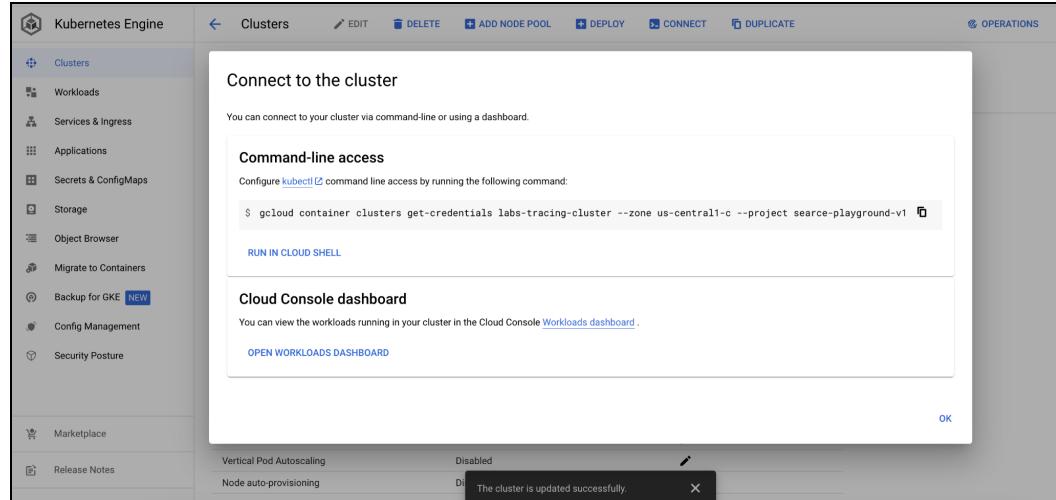
- In Security, Select the checkbox **Enable Workload Identity**.



- Metadata and Features, leave with default values.
- Cluster creation may take from 5-10 mins. Cluster named labs-demo-cluster is created successfully.

Status	Name	Location	Number of nodes	Total vCPUs	Total memory	Notifications
<input type="checkbox"/>	<input checked="" type="checkbox"/> labs-tracing-cluster	us-central1-c	1	0.2	0.2 GiB	

- To connect to the cluster, click on cluster name, Click on **CONNECT** and copy the command-line access command. Run on your terminal



Follow the documentation - [kubectl](#). Post installation of kubectl run command **kubectl get nodes** to verify nodes details.

```
~ % gcloud container clusters get-credentials labs-tracing-cluster --zone us-central1-c --project searce-playground-v1
Fetching cluster endpoint and auth data.
CRITICAL: ACTION REQUIRED: gke-gcloud-auth-plugin, which is needed for continued use of kubectl, was not found or is not executable. Install gke-gcloud-auth-plugin for use with kubectl by following https://cloud.google.com/blog/products/containers-kubernetes/kubectl-auth-changes-in-gke
kubecfg entry generated for labs-tracing-cluster.
~ % kubectl get nodes
W1128 20:34:41.391103    6291 gcp.go:119] WARNING: the gcp auth plugin is deprecated in v1.22+, unavailable in v1.26+; use gcloud instead.
To learn more, consult https://cloud.google.com/blog/products/containers-kubernetes/kubectl-auth-changes-in-gke
NAME           STATUS   ROLES      AGE   VERSION
gke-labs-tracing-cluster-default-pool-eb5c4eff-g7w3   Ready    <none>    10m   v1.23.12-gke.100
```

- **Install Istio**

Now, Install Istio CLI command in your system.

- To Download [Istioctl](#) command run this Command.

```
$ curl -L https://istio.io/downloadIstio | sh -
```

```
~ % curl -L https://istio.io/downloadIstio | sh -
          % Total    % Received % Xferd  Average Speed   Time   Time     Time Current
                                         Dload  Upload Total   Spent   Left Speed
100  101  100  101    0     0   149      0  --:--:-- --:--:-- --:--:-- 150
100 4856 100 4856    0     0   4140      0  0:00:01  0:00:01 --:--:-- 4140

Downloading istio-1.16.0 from https://github.com/istio/istio/releases/download/1.16.0/istio-1.16.0-osx-arm64.tar.gz ...
Istio 1.16.0 Download Complete!

Istio has been successfully downloaded into the istio-1.16.0 folder on your system.

Next Steps:
See https://istio.io/latest/docs/setup/install/ to add Istio to your Kubernetes cluster.

To configure the istioctl client tool for your workstation,
add the /Users/tanmaydoke/istio-1.16.0/bin directory to your environment path variable with:
export PATH="$PATH:/Users/tanmaydoke/istio-1.16.0/bin"

Begin the Istio pre-installation check by running:
    istioctl x precheck

Need more information? Visit https://istio.io/latest/docs/setup/install/
~ %
```

- Set the ENV variable for `istioctl` command.

```
$ export PATH=$PATH:/path/of/your/istio/bin/dir"
```

```
~ % export PATH="/Users/          /istio-1.16.0/bin"
~ % echo $PATH
/Users/~ 'Downloads/google-cloud-sdk/bin:/opt/homebrew/bin:/opt/homebrew/sbin:/usr/local/bin:/usr/bin:/bin:/usr
/sbin:/sbin:/Users/~ /istio-1.16.0/bin
```

- Allow require port number for this step in Kubernetes Cluster firewalls. Follow the given command.

Note: Set `Cluster_name` environment variable.

```
export $CLUSTER_NAME=<your-cluster-name>
```

```
#list the master firewall rule
gcloud compute firewall-rules list
--filter="name~gke-${CLUSTER_NAME}-[0-9a-z]*-master"
```

```
#update the firewall rule with additional ports
$gcloud compute firewall-rules update
<firewall-rule-name>
--allow
tcp:10250,tcp:443,tcp:15017,tcp:8080,tcp:15000
```

```
microservices-demo % gcloud compute firewall-rules list --filter="name~gke-${CLUSTER_NAME}-[0-9a-z]*-master"
NAME           NETWORK   DIRECTION  PRIORITY ALLOW  DENY  DISABLED
gke-labs-tracing-cluster-e790fc48-master  labs-vpc  INGRESS  1000    tcp:10250,tcp:443,tcp:15017  False
```

```
microservices-demo % gcloud compute firewall-rules update gke-labs-tracing-cluster-e790fc48-master --allow tcp:10250,tcp:443,tcp:15017,
tcp:8080,tcp:15000
Updated [https://www.googleapis.com/compute/v1/projects/searce-playground-v1/global/firewalls/gke-labs-tracing-cluster-e790fc48-master].
microservices-demo % gcloud compute firewall-rules list --filter="name~gke-${CLUSTER_NAME}-[0-9a-z]*-master"
NAME           NETWORK   DIRECTION  PRIORITY ALLOW  DENY  DISABLED
gke-labs-tracing-cluster-e790fc48-master  labs-vpc  INGRESS  1000    tcp:10250,tcp:443,tcp:15017,tcp:8080,tcp:15000  False
```

- Move to the Istio package directory. For example, if the package is `istio-1.16.0`:

```
$ cd istio-1.16.0
```

- The installation directory contains: Sample applications in `samples`/ The `istioctl` client binary in the `bin/` directory.
- Install the istio using the below command.

```
$ istioctl manifest apply --set profile=demo -y
```

```
Istio-1.16.0 % istioctl manifest apply --set profile=demo
This will install the Istio 1.16.0 demo profile with ["Istio core" "Istiod" "Ingress gateways" "Egress gateways"] components into the cluster. Proceed? (y/N) y
✓ Istio core installed
✓ Istiod installed
✓ Egress gateways installed
✓ Ingress gateways installed
✓ Installation complete
Making this installation the default for injection and validation.

Thank you for installing Istio 1.16. Please take a few minutes to tell us about your install/upgrade experience! http
s://forms.gle/99uiMML96AmsXY5d6
```

- Enable istio sidecar injection on namespace default to attach sidecar to running workloads.

```
$ istioctl install --set profile=demo -y
$ kubectl label namespace default istio-injection=enabled
```

```
~ % istioctl install --set profile=demo -y
✓ Istio core installed
✓ Istiod installed
✓ Egress gateways installed
✓ Ingress gateways installed
✓ Installation complete
Making this installation the default for injection and validation.

Thank you for installing Istio 1.16. Please take a few minutes to tell us about your install/upgrade experience! http://forms.gle/99uiMML96AmsXY5d6
~ % kubectl label namespace default istio-injection=enabled
W1128 20:44:11.803286    6404 gcp.go:119] WARNING: the gcp auth plugin is deprecated in v1.22+, unavailable in v1.26+; use gcloud instead.
To learn more, consult https://cloud.google.com/blog/products/containers-kubernetes/kubectl-auth-changes-in-gke
namespace/default labeled
~ %
```

- To verify, execute the following command.

```
$ kubectl get ns --show-labels
```

```
~ % kubectl get ns --show-labels
W1128 20:46:14.505288    6483 gcp.go:119] WARNING: the gcp auth plugin is deprecated in v1.22+, unavailable in v1.26+; use gcloud instead.
To learn more, consult https://cloud.google.com/blog/products/containers-kubernetes/kubectl-auth-changes-in-gke
^[[A
NAME           STATUS   AGE   LABELS
default        Active   23m   istio-injection=enabled,kubernetes.io/metadata.name=default
istio-system   Active   3m41s  kubernetes.io/metadata.name=istio-system
kube-node-lease Active   23m   kubernetes.io/metadata.name=kube-node-lease
kube-public    Active   23m   kubernetes.io/metadata.name=kube-public
kube-system    Active   23m   kubernetes.io/metadata.name=kube-system
```

- Deploy the sample application

- Google uses this application to demonstrate use of technologies like Kubernetes/GKE, Istio, Stackdriver, and gRPC. This application works on any Kubernetes cluster, as well as Google Kubernetes Engine. It's easy to deploy with little to no configuration. Clone this repository.

```
$ git clone
https://github.com/GoogleCloudPlatform/microservices-demo.git

$ cd microservices-demo
```

```

~ % git clone https://github.com/GoogleCloudPlatform/microservices-demo.git
Cloning into 'microservices-demo'...
remote: Enumerating objects: 8818, done.
remote: Counting objects: 100% (273/273), done.
remote: Compressing objects: 100% (136/136), done.
remote: Total 8818 (delta 160), reused 217 (delta 136), pack-reused 8545
Receiving objects: 100% (8818/8818), 30.86 MiB / 5.15 MiB/s, done.
Resolving deltas: 100% (6290/6290), done.
~ % cd microservices-demo
microservices-demo %

```

- Deploy the sample app to the cluster.

\$ kubectl apply -f ./release/kubernetes-manifests.yaml

```

microservices-demo % kubectl apply -f ./release/kubernetes-manifests.yaml
W1129 20:50:55.410070  6570 gcp.go:119] WARNING: the gcp auth plugin is deprecated in v1.22+, unavailable in v1.26+; use gcloud instead.
To learn more, consult https://cloud.google.com/blog/products/containers-kubernetes/kubectl-auth-changes-in-gke
deployment.apps/emailservice created
service/emailservice created
deployment.apps/checkoutservice created
service/checkoutservice created
deployment.apps/recommendationservice created
service/recommendationservice created
deployment.apps/frontend created
service/frontend created
service/frontend-external created
deployment.apps/paymentservice created
service/paymentservice created
deployment.apps/productcatalogservice created
service/productcatalogservice created
deployment.apps/cartservice created
service/cartservice created
deployment.apps/loadgenerator created
deployment.apps/currencyservice created
service/currencyservice created
deployment.apps/shippingservice created
service/shippingservice created
deployment.apps/redis-cart created
service/redis-cart created
deployment.apps/adservice created
service/adservice created
error: resource mapping not found for name: "" namespace: "" from "./release/kubernetes-manifests.yaml": no matches for
kind "Kustomization" in version "kustomize.config.k8s.io/v1beta1"
ensure CRDs are installed first
microservices-demo %

```

- To verify, execute the following command it will show 2/2 pods means istio side-car running in each pods.

\$ kubectl get pod

```

microservices-demo % kubectl get pods
W1129 13:01:44.714022 19469 gcp.go:119] WARNING: the gcp auth plugin is deprecated in v1.22+, unavailable in v1.26+; use gcloud instead.
To learn more, consult https://cloud.google.com/blog/products/containers-kubernetes/kubectl-auth-changes-in-gke
NAME          READY   STATUS    RESTARTS   AGE
adservice-5c47dc7fbf-stjd4   2/2     Running   0          11h
cartservice-8499d78bd4-bzfqp  2/2     Running   0          11h
checkoutservice-6ccf5c787f-82k9q  2/2     Running   0          11h
currencyservice-549dc64c55-46p8t  2/2     Running   0          11h
emailservice-56ddbd949-rqfgr   2/2     Running   0          11h
frontend-6df7744567-wks6      2/2     Running   0          11h
loadgenerator-d4f74c9b6-7m4zm  2/2     Running   0          11h
paymentservice-b4b7fc5dc-jlvdn  2/2     Running   0          11h
productcatalogservice-546759f798-h2z49  2/2     Running   0          11h
recommendationservice-66f9fdbb7c-h95s9  2/2     Running   0          11h
redis-cart-58648d854-99xkv    2/2     Running   0          11h
shippingservice-7bc77d59ff-cfckh   2/2     Running   0          11h

```

- View the dashboard

Use the following instructions to deploy the [Kiali](#) dashboard, along with [Prometheus](#), [Grafana](#), and [Jaeger](#).

- Install [Kiali and the other addons](#) and wait for them to be deployed.

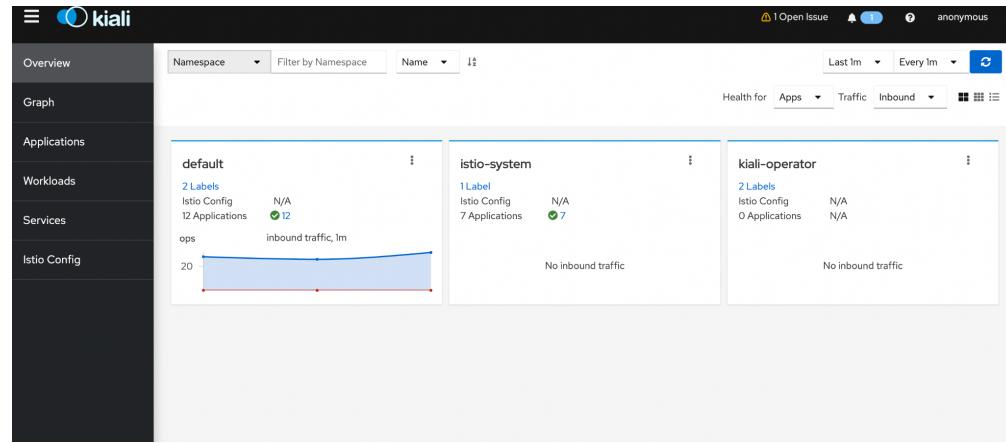
```
$ cd istio-1.16.0  
$ kubectl apply -f samples/addons
```

```
istio-1.16.0 % kubectl apply -f samples/addons  
W1129 13:38:12.126619 19813 gcp.go:119] WARNING: the gcp auth plugin is deprecated in v1.22+, unavailable in v1.26+; use gcloud instead.  
To learn more, consult https://cloud.google.com/blog/products/container-kubernetes/kubectl-auth-changes-in-gke  
serviceaccount/grafana unchanged  
configmap/grafana unchanged  
service/grafana unchanged  
deployment.apps/grafana configured  
configmap/istio-grafana-dashboards configured  
configmap/istio-services-grafana-dashboards configured  
deployment.apps/jaeger created  
service/tracing created  
service/zipkin created  
service/jaeger-collector created  
serviceaccount/kiali unchanged  
configmap/kiali unchanged  
clusterrole.rbac.authorization.k8s.io/kiali-viewer unchanged  
clusterrole.rbac.authorization.k8s.io/kiali unchanged  
clusterrolebinding.rbac.authorization.k8s.io/kiali unchanged  
role.rbac.authorization.k8s.io/kiali-controlplane unchanged  
rolebinding.rbac.authorization.k8s.io/kiali-controlplane unchanged  
service/kiali unchanged  
deployment.apps/kiali configured  
serviceaccount/prometheus created  
configmap/prometheus created  
clusterrole.rbac.authorization.k8s.io/prometheus created  
clusterrolebinding.rbac.authorization.k8s.io/prometheus created  
service/prometheus created  
deployment.apps/prometheus created  
istio-1.16.0 % kubectl rollout status deployment/kiali -n istio-system  
W1129 13:39:17.276163 19814 gcp.go:119] WARNING: the gcp auth plugin is deprecated in v1.22+, unavailable in v1.26+; use gcloud instead.  
To learn more, consult https://cloud.google.com/blog/products/container-kubernetes/kubectl-auth-changes-in-gke  
deployment "kiali" successfully rolled out  
istio-1.16.0 %
```

- Access the Kiali dashboard. It will show the below dashboard.

```
$ istioctl dashboard kiali
```

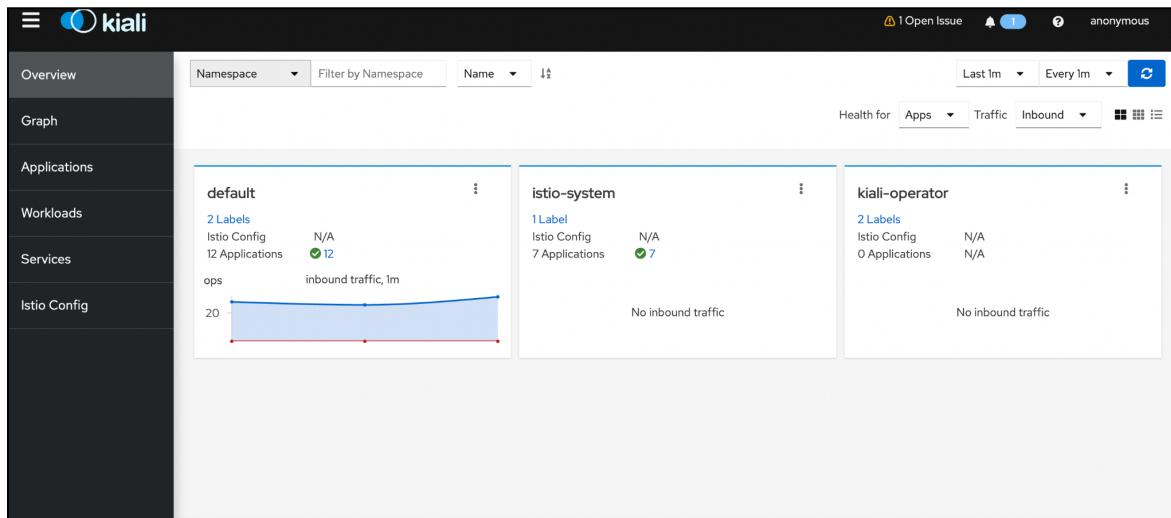
```
istio-1.16.0 % istioctl dashboard kiali  
http://localhost:20001/kiali
```



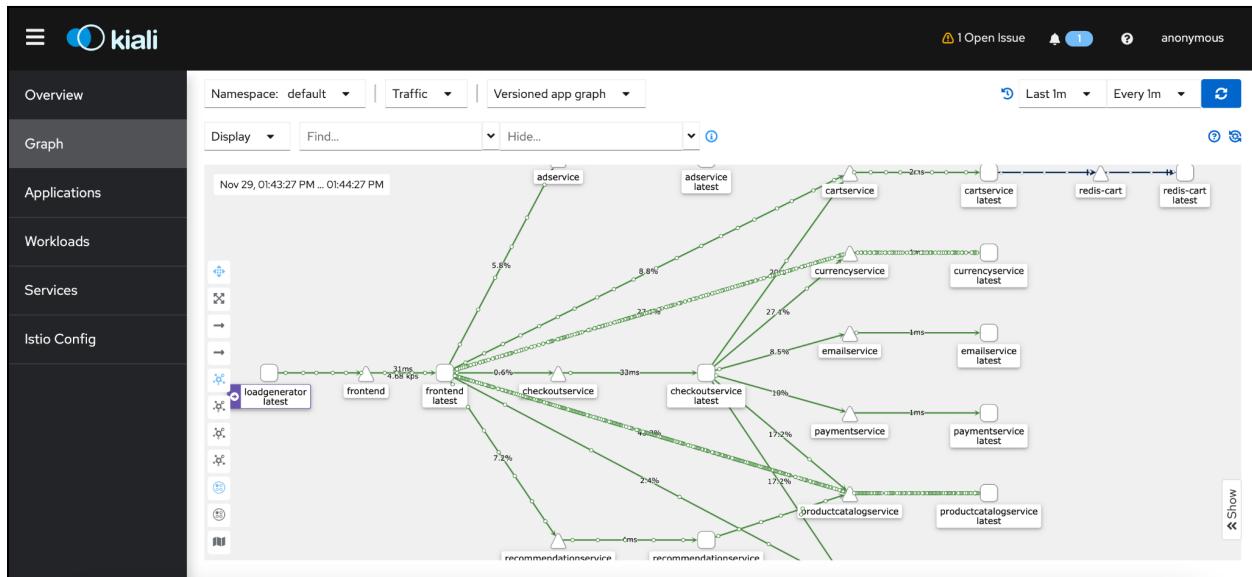
Kiali Dashboard Outputs

In the left navigation menu, select Graph and in the Namespace drop down, select default. The Kiali dashboard shows an overview of your mesh with the relationships between the services in the sample application. It also provides filters to visualize the traffic flow.

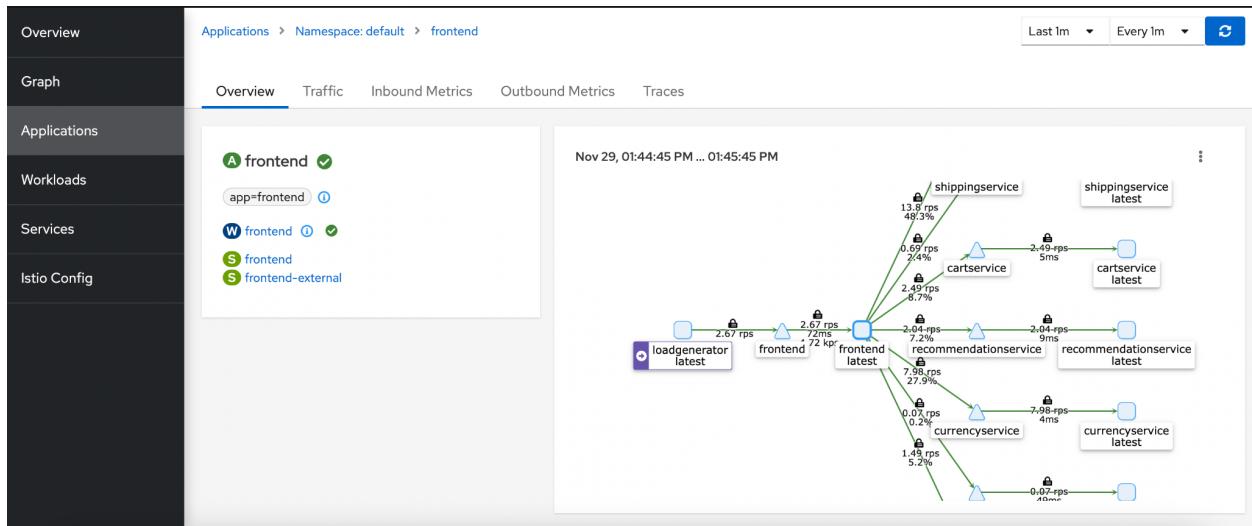
Overview Panel



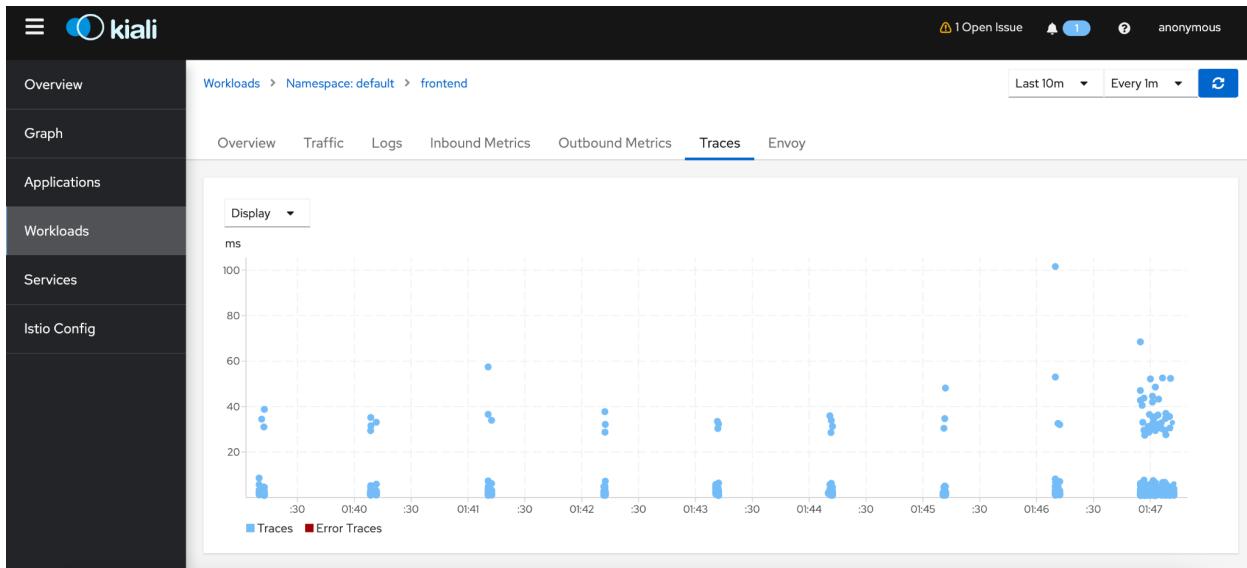
Graph panel



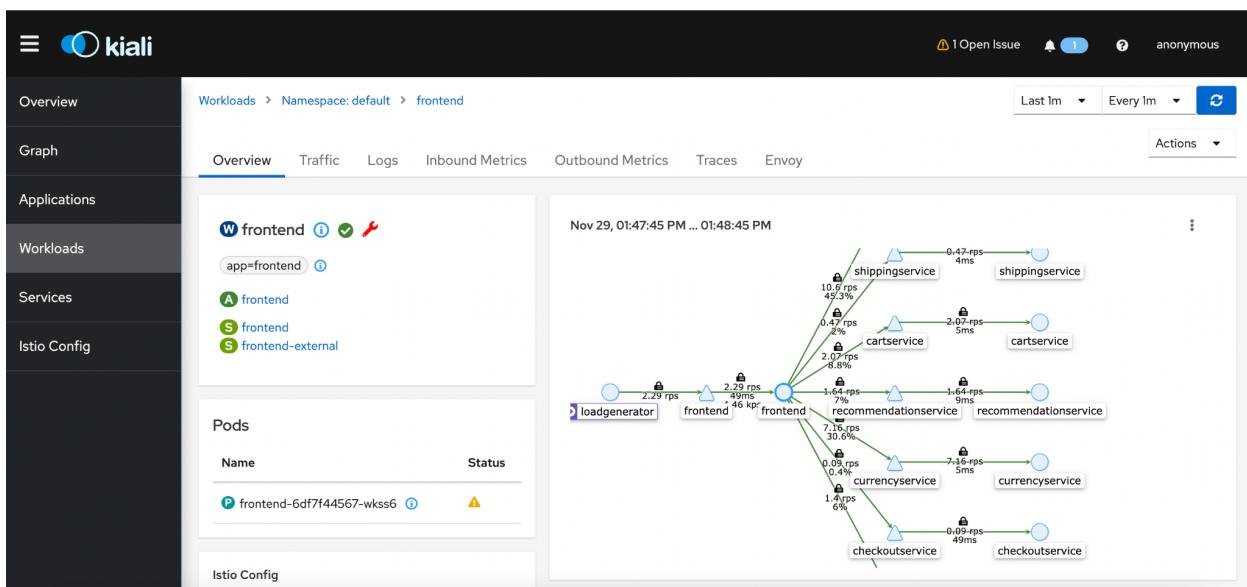
Application panel



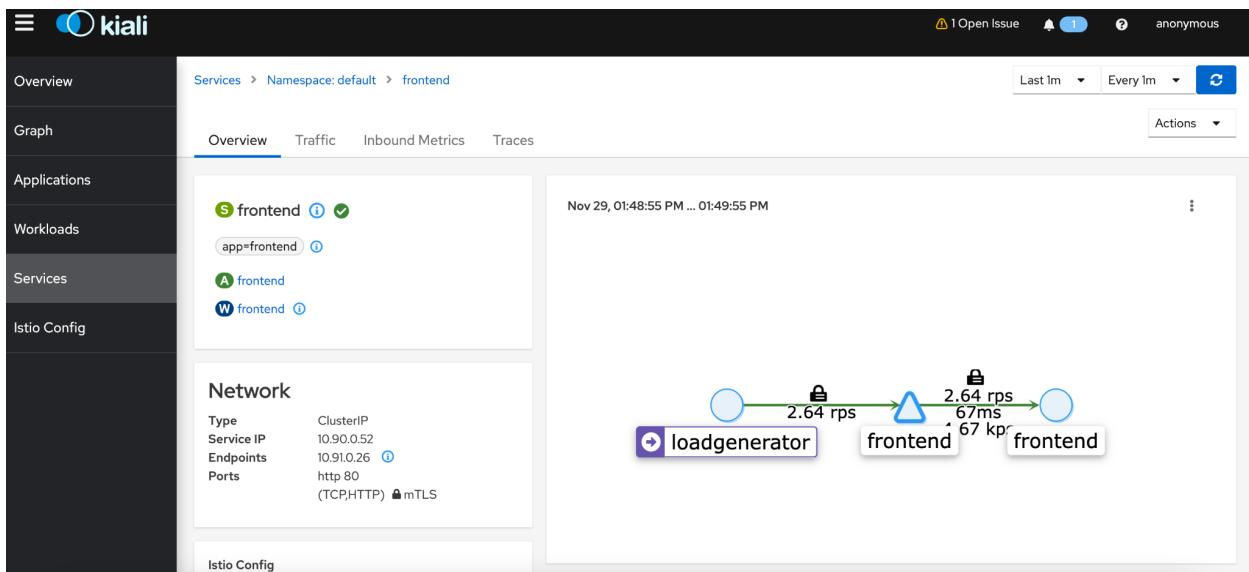
Workloads panel - Traces



Workloads Panel - Overview



Services Panel



Conclusion

In this lab, we deployed a boutique application which consisted of multiple microservices intercommunication. By using kiali dashboard we traced the latency and traffic distribution between different microservices.

References

- Download and install Istio : [istio-downloads](#)
- Deploy the sample application : [link](#)
- Visualizing Your Mesh : [kiali](#)
- Istio concept : [istio](#)