

Table of Contents

Service Mesh Monitoring with Kiali Lab

- [1. Review Service Mesh Monitoring](#)
- [2. Explore Kiali](#)
- [3. Generate Sample Data](#)
- [4. View Service Graph](#)
- [5. View Animations in Service Graph](#)
- [6. Explore Service Listing](#)
- [7. Clean Up Environment](#)
- [8. References](#)

Service Mesh Monitoring with Kiali Lab

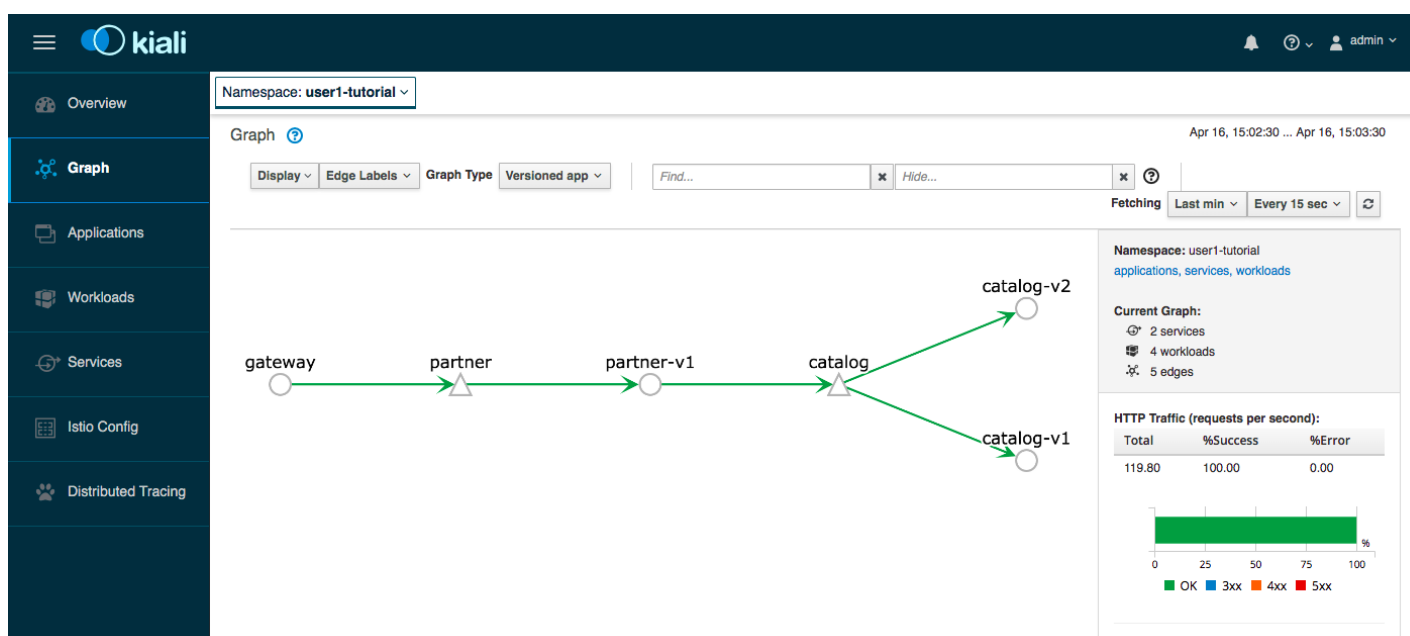
Goal

- Monitor your Istio service mesh with Kiali

1. Review Service Mesh Monitoring

At some point when you are developing your microservices architecture, you may want to visualize what is happening in your service mesh. You may have questions like “Which service is connected to which other service?” and “How much traffic goes to each microservice?” But because of the loosely tied nature of microservices architectures, these questions can be difficult to answer.

Those are the kinds of question that [Kiali](https://www.kiali.io/) (<https://www.kiali.io/>) has the ability to answer—by giving you a big picture of the mesh and showing the whole flow of your requests and data.



Kiali taps into the data provided by Istio and OpenShift Container Platform to generate its visualizations. It fetches ingress data (such as request tracing with Jaeger), the listing and data of the services, health indexes, and so on.

Kiali runs as a service together with Istio, and does not require any changes to Istio or OpenShift Container Platform configuration (besides the ones required to install Istio).

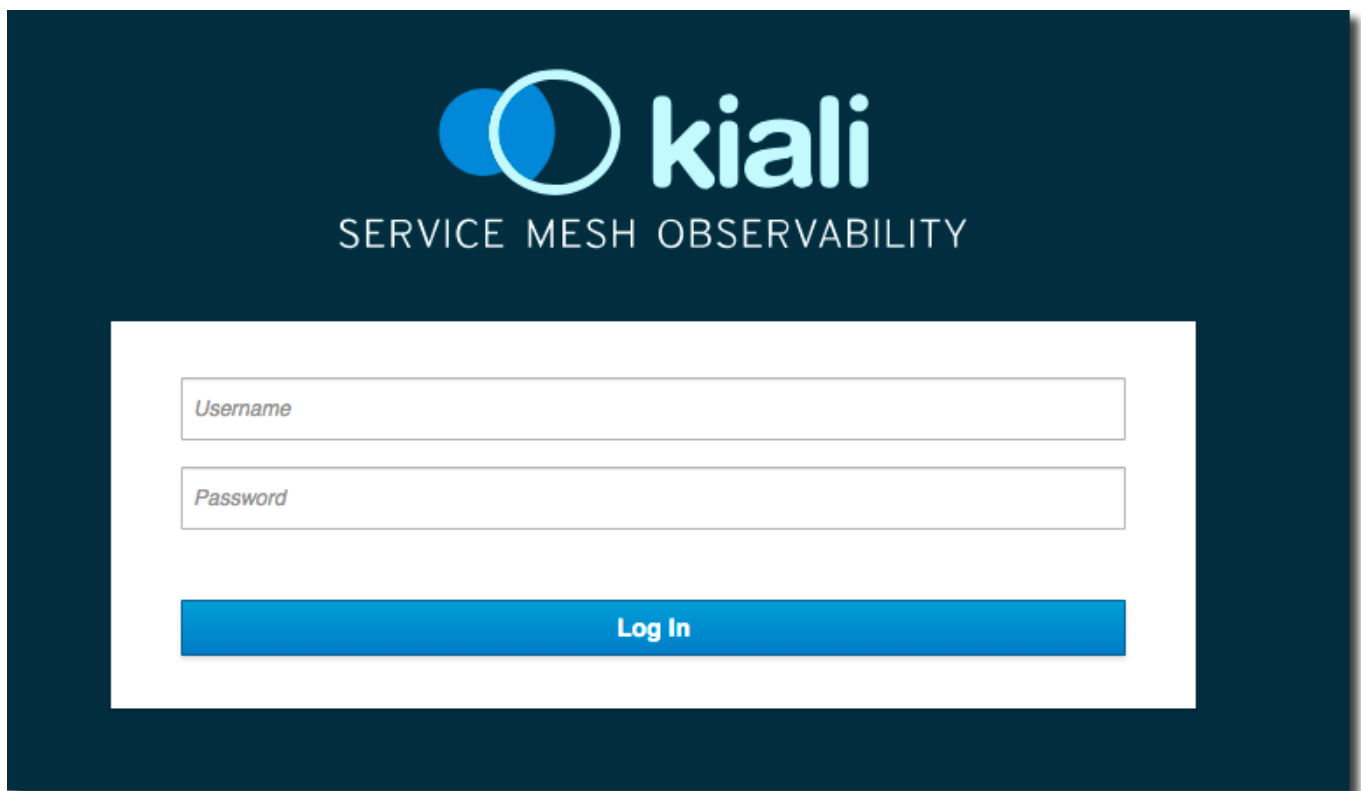
1. Get the URL of the Kiali web console and set as an environment variable:

```
export KIALI_URL=https://$(oc get route kiali -n istio-system -o template  
--template='{{.spec.host}}')
```

2. Display the **KIALI_URL** URL:

```
echo $KIALI_URL
```

3. Start a web browser on your computer and go to the URL for **\$KIALI_URL**:



4. At the login screen, enter the default credentials:

- **Username:** **admin**
- **Password:** **r3dh4t1!**

3. Generate Sample Data

To show the capabilities of Kiali, you need to generate some sample data. For this, you can use the

1. Move back to your terminal window:

```
cd ~/lab/ocp-service-mesh-foundations
```

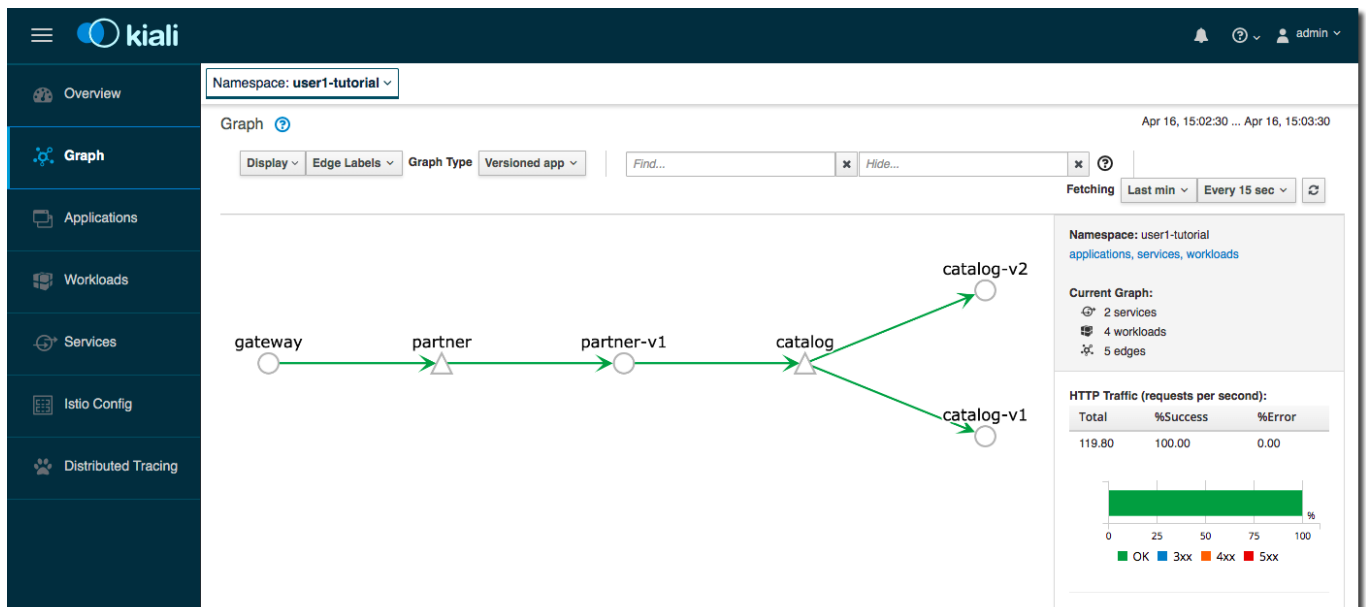
2. Generate data in the same manner you did in the previous lab:

```
$HOME/lab/ocp-service-mesh-foundations/scripts/run-all.sh
```

- o Let this script continue to run.

4. View Service Graph

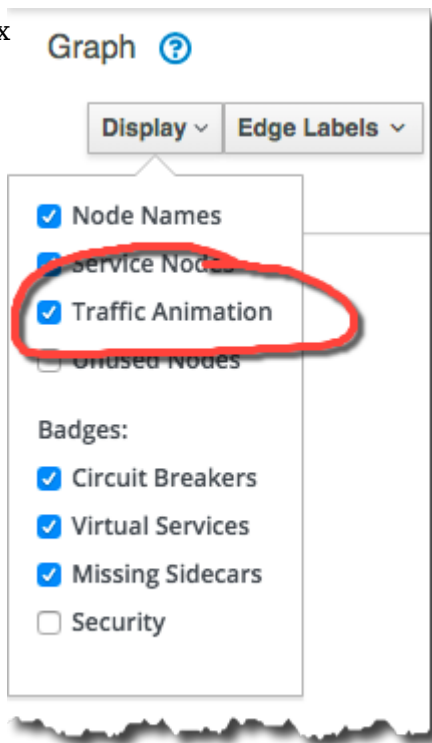
1. Move back to the Kiali web console.
2. In the left-hand panel, click **Graph**.
3. From the **Namespace** list, select **user1-tutorial**.



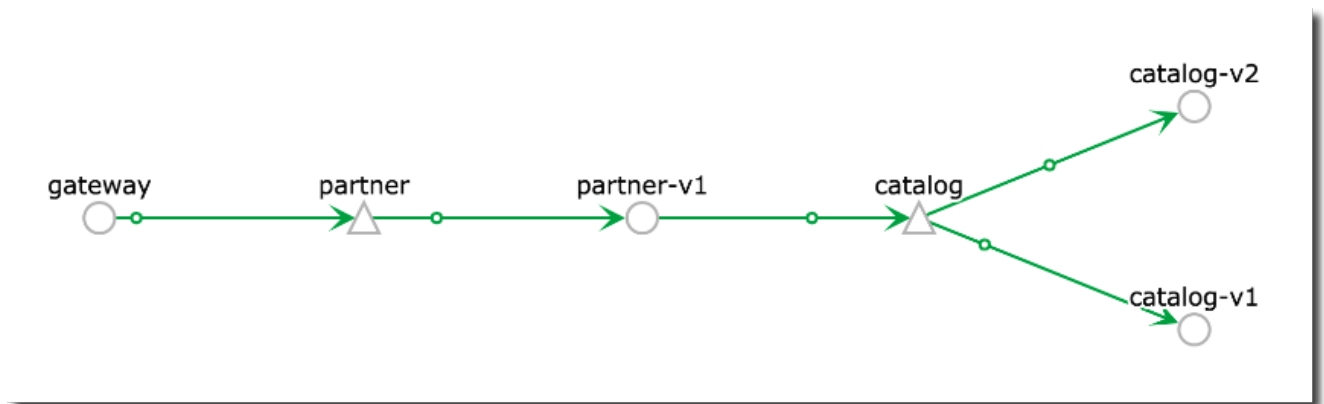
- o This page shows a graph of the microservices, connected by the requests going through them. On this page you can see how the services interact with each other, and you can zoom in or out.

5. View Animations in Service Graph

1. On the **Graph** screen, click the **Display** list and check the **Traffic Animation** option:



- Expect to see traffic animation on the graph based on traffic that is generated by the `scripts/run-all.sh` script you started earlier:



- Currently, load-balancing between the different versions of the catalog service is occurring via a Service Mesh *VirtualService*.

You can visualize the round-robin load-balancing of OpenShift by simply deleting the *VirtualService*.

- Delete the *catalog* virtualservice:

```
oc delete VirtualService catalog -n $OCP_TUTORIAL_PROJECT
```

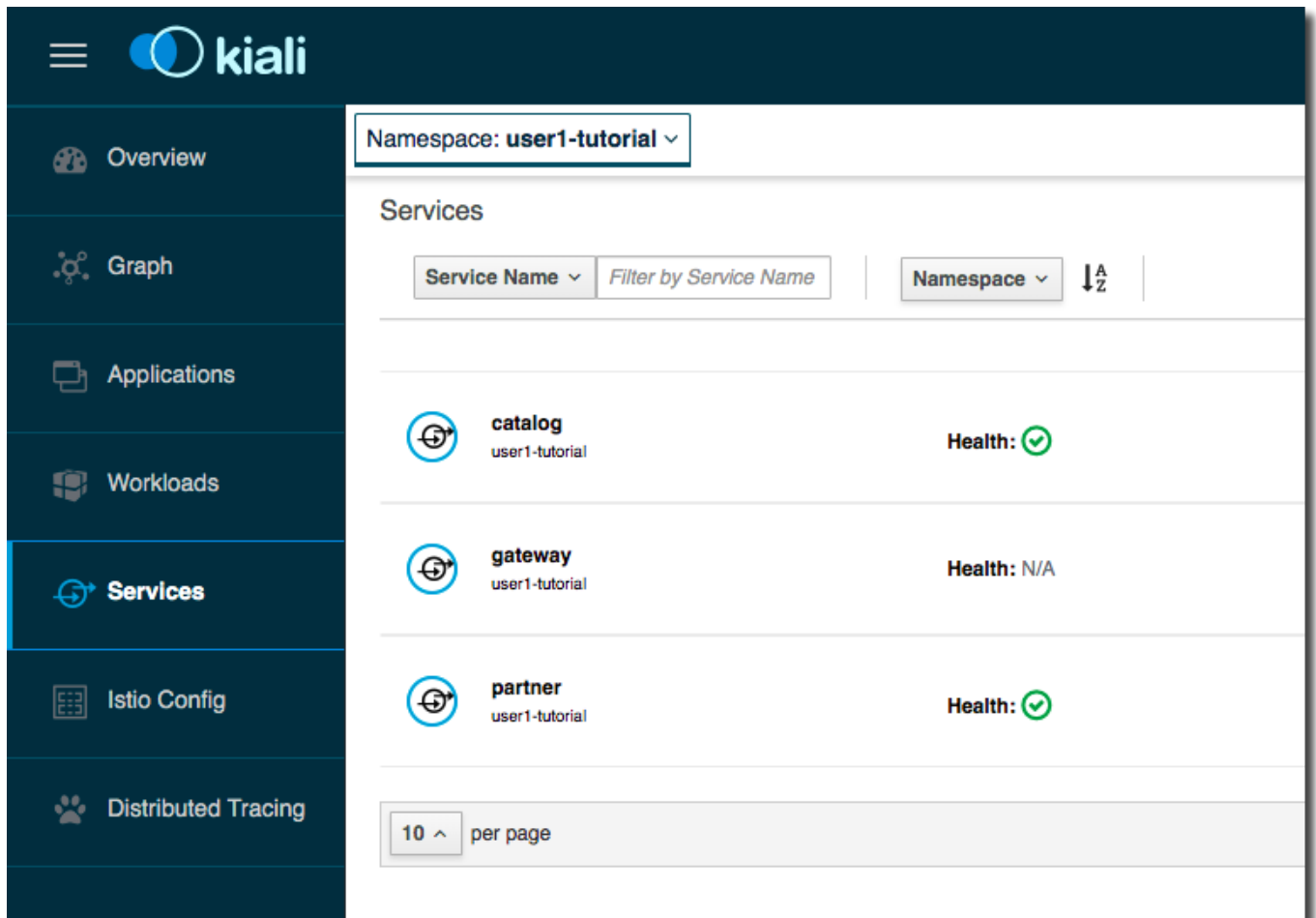
- Observe the animation in Kiali now displays the traffic being load-balanced in a round-robin manner between v1 and v2 of the catalog service.

6. Explore Service Listing

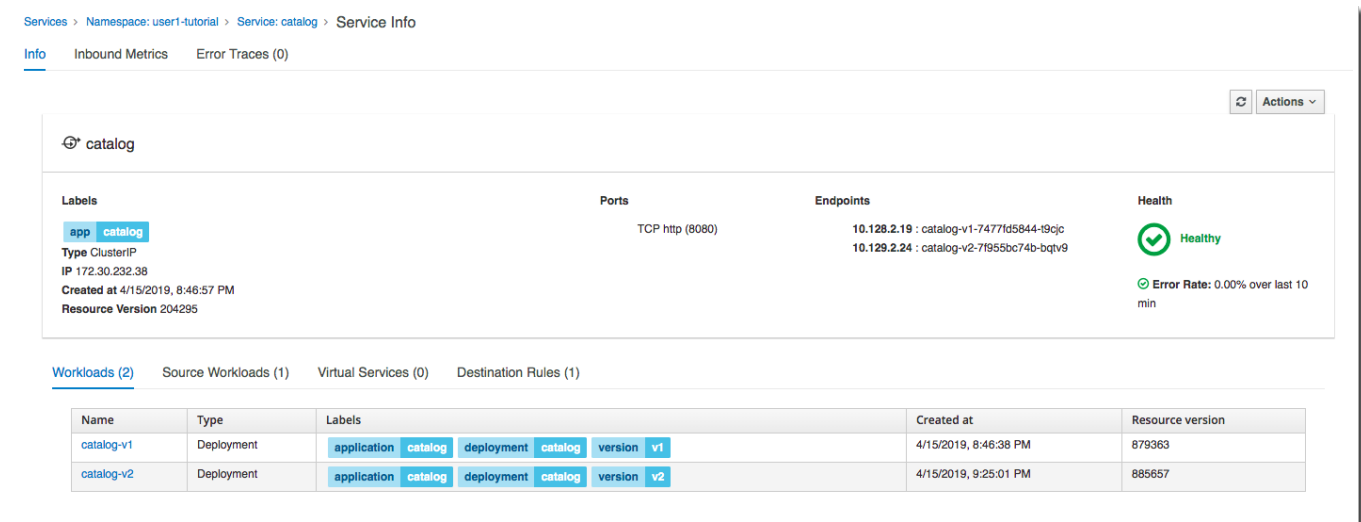
- In the left-hand panel, click **Services**.

Firefox ○ On the Services page you can view a listing of the services that are running in the cluster, and additional information about them such as health status.

2. Observe that the **Namespace** list is set to **user1-tutorial**. This filters the list of services to just those for this tutorial namespace:



3. Click the **catalog** service to see its details:



- At the bottom, you can see the catalog service's workloads, which display the pod(s).

7. Clean Up Environment

1. Move back to the terminal window running the script.

This completes the lab for using Kiali to monitor your Istio service mesh.

8. References

- [Kiali \(https://www.kiali.io/\)](https://www.kiali.io/)
- [Istio Homepage \(https://istio.io\)](https://istio.io)
- [Learn Istio on OpenShift \(https://learn.openshift.com/servicemesh\)](https://learn.openshift.com/servicemesh)

Last updated 2020-02-26 17:17:48 EST