Configuring add-ons in Istio:

There are many add-ons available for Istio. For example:

1. Prometheus
2. Grafana
3. Zipkin
4. Jaeger
5. Service Graph

Exposing services:

Istio add-on services are deployed by default as ***ClusterIP*** type services. We can expose the services outside the cluster by either changing the Kubernetes service type to ***NodePort*** or ***LoadBalancer*** or by port-forwarding or by configuring Kubernetes Ingress.

Here, we briefly explain the NodePort and port-forwarding ways of exposing services.

Option 1: Expose services with NodePort

To expose them using NodePort service type, we can edit the services and change the service type from ClusterIP to NodePort:

```
kubectl -n istio-system edit svc prometheus

kubectl -n istio-system edit svc grafana

kubectl -n istio-system edit svc servicegraph
```

For Jaeger, either of tracing or jaeger-query can be exposed.

```
kubectl -n istio-system edit svc tracing
```

Once this is done the services will be assigned dedicated ports on the hosts.

To find the assigned ports for Grafana:

```
kubectl -n istio-system get svc grafana
```

Once this is done the services will be assigned dedicated ports on the hosts.

To find the assigned ports for Grafana:

```
kubectl -n istio-system get svc grafana
```

To find the assigned ports for Prometheus:

```
kubectl -n istio-system get svc prometheus
```

To find the assigned ports for Servicegraph:

```
kubectl -n istio-system get svc servicegraph
```

To find the assigned ports for Jaeger:

```
kubectl -n istio-system get svc tracing
```

Option 2: Expose services with port-forwarding

To port-forward Grafana:

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

To port-forward Prometheus:

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

To port-forward 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 &
```

To port-forward Jaeger:

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

Port-forwarding runs in the foreground. We have appeneded & to the end of the above 2 commands to run them in the background. If you do not want this behavior, please remove the & from the end.

Accessing Exposed Services:

once a port is exposed, you can access the service from local host:

```
http://localhost:<port-number>
```