

1. Install and unzip istio-1.28.1 from the official website

2. Move to the Istio package directory

```
cd istio-1.28-1
```

3. Add the istioctl client to my path

```
$env:PATH = "$PWD\bin;$env:PATH"
```

4. Install Istio using the demo profile, without any gateways

```
istioctl install -f samples\bookinfo\demo-profile-no-gateways.yaml -y
```

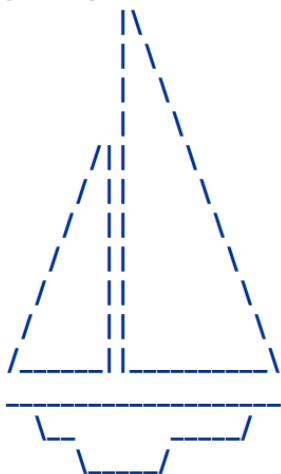
5. Install Istio using the demo profile, without any gateways

```
kubectl label namespace default istio-injection=enabled
```

```
PS D:\> cd istio-1.28.1
```

```
PS D:\istio-1.28.1> $env:PATH = "$PWD\bin;$env:PATH"
```

```
PS D:\istio-1.28.1> istioctl install -f samples\bookinfo\demo-profile-no-gateways.yaml -y
```



```
✓ Istio core installed 🎉  
✓ Istiod installed 💡  
✓ Installation complete
```

```
PS D:\istio-1.28.1> kubectl label namespace default istio-injection=enabled  
namespace/default labeled
```

6. Install the Gateway API CRDs

```
kubectl kustomize "github.com/kubernetes-sigs/gateway-api/config/crd?ref=v1.4.0" |  
kubectl apply -f -
```

Problem that I met:

```
PS D:\istio-1.28.1> kubectl kustomize "github.com/kubernetes-sigs/gateway-api/config/crd?ref=v1.4.0" | kubectl apply -f -  
error: failed to run 'D:\Git\cmd\git.exe fetch --depth=1 https://github.com/kubernetes-sigs/gateway-api v1.4.0': fatal: unable to access 'https://github.com/kubernetes-sigs/gateway-api/': Failed to connect to github.com port 443 after 21103 ms: Could not connect to server  
: exit status 128  
error: no objects passed to apply
```

Solution: pull manually

```
kubectl apply --server-side -f D:\istio-1.28.1\gateway-api\standard-install.yaml
```

```
PS D:\istio-1.28.1> kubectl apply -f D:\istio-1.28.1\gateway-api\standard-install.yaml
Warning: unrecognized format "int64"
Warning: unrecognized format "int32"
customresourcedefinition.apiextensions.k8s.io/backendtlspolicies.gateway.networking.k8s.io created
customresourcedefinition.apiextensions.k8s.io/gatewayclasses.gateway.networking.k8s.io created
customresourcedefinition.apiextensions.k8s.io/gateways.gateway.networking.k8s.io created
customresourcedefinition.apiextensions.k8s.io/grpcroutes.gateway.networking.k8s.io created
customresourcedefinition.apiextensions.k8s.io/httproutes.gateway.networking.k8s.io created
customresourcedefinition.apiextensions.k8s.io/referencegrants.gateway.networking.k8s.io created
PS D:\istio-1.28.1> kubectl get crd | findstr gateway.networking.k8s.io
backendtlspolicies.gateway.networking.k8s.io    2025-12-15T09:00:38Z
gatewayclasses.gateway.networking.k8s.io        2025-12-15T09:00:38Z
gateways.gateway.networking.k8s.io              2025-12-15T09:00:38Z
grpcroutes.gateway.networking.k8s.io            2025-12-15T09:00:39Z
httproutes.gateway.networking.k8s.io             2025-12-15T09:00:39Z
referencegrants.gateway.networking.k8s.io        2025-12-15T09:00:39Z
```

7. Deploy the Bookinfo sample application

```
kubectl apply -f samples/bookinfo/platform/kube/bookinfo.yaml
```

```
PS D:\istio-1.28.1> kubectl apply -f samples/bookinfo/platform/kube/bookinfo.yaml
service/details created
serviceaccount/bookinfo-details created
deployment.apps/details-v1 created
service/ratings created
serviceaccount/bookinfo-ratings created
deployment.apps/ratings-v1 created
service/reviews created
serviceaccount/bookinfo-reviews created
deployment.apps/reviews-v1 created
deployment.apps/reviews-v2 created
deployment.apps/reviews-v3 created
service/productpage created
serviceaccount/bookinfo-productpage created
deployment.apps/productpage-v1 created
```

8. Check if each pod becomes ready

```
-----  
PS D:\istio-1.28.1> kubectl get services
NAME      TYPE      CLUSTER-IP      EXTERNAL-IP      PORT(S)      AGE
details   ClusterIP  10.96.125.79  <none>          9080/TCP    16s
kubernetes  ClusterIP  10.96.0.1    <none>          443/TCP     59d
productpage ClusterIP  10.99.180.196  <none>          9080/TCP    16s
ratings   ClusterIP  10.106.87.186  <none>          9080/TCP    16s
reviews   ClusterIP  10.111.60.241  <none>          9080/TCP    16s
```

```

PS D:\istio-1.28.1> kubectl get pods
NAME                  READY   STATUS    RESTARTS   AGE
details-v1-77d6bd5675-7jh2h   2/2     Running   0          4m49s
productpage-v1-bb87ff47b-6hjrj 2/2     Running   0          4m48s
ratings-v1-8589f64b4c-4ztrj   2/2     Running   0          4m49s
reviews-v1-8cf7b9cc5-wh2v8    2/2     Running   0          4m49s
reviews-v2-67d565655f-t7kc5   2/2     Running   0          4m49s
reviews-v3-d587fc9d7-v6m4w    2/2     Running   0          4m49s

```

Note that the pods show READY 2/2, confirming they have their application container and the Istio sidecar container.

9. Note that the pods show READY 2/2, confirming they have their application container and the Istio sidecar container.

```

$RATINGS_POD = kubectl get pod -l app=ratings -o jsonpath="{.items[0].metadata.name}"
kubectl exec $RATINGS_POD -c ratings -- `

curl -sS productpage:9080/productpage |
Select-String "<title>.*</title>" |
ForEach-Object { $_.Matches[0].Value }

```

```

PS D:\istio-1.28.1> $RATINGS_POD = kubectl get pod -l app=ratings -o jsonpath=".items[0].metadata.name"
PS D:\istio-1.28.1> kubectl exec $RATINGS_POD -c ratings -- `

>> curl -sS productpage:9080/productpage |
>> Select-String -Pattern "<title>.*</title>" |
>> ForEach-Object { $_.Matches[0].Value }

<title>Simple Bookstore App</title>

```

10. Create a Kubernetes Gateway for the Bookinfo application

```
kubectl apply -f samples/bookinfo/gateway-api/bookinfo-gateway.yaml
```

11. Change the service type to ClusterIP by annotating the gateway

```

kubectl annotate gateway bookinfo-gateway networking.istio.io/service-type=ClusterIP
--namespace=default

PS D:\istio-1.28.1> kubectl apply -f samples/bookinfo/gateway-api/bookinfo-gateway.yaml
gateway.gateway.networking.k8s.io/bookinfo-gateway created
httproute.gateway.networking.k8s.io/bookinfo created
PS D:\istio-1.28.1> kubectl annotate gateway bookinfo-gateway networking.istio.io/
service-type=ClusterIP --namespace=default
gateway.gateway.networking.k8s.io/bookinfo-gateway annotated
PS D:\istio-1.28.1> kubectl get gateway
NAME      CLASS      ADDRESS                                     PROG
RAMMED    AGE
bookinfo-gateway  istio  bookinfo-gateway-istio.default.svc.cluster.local  True
21s

```

12. kubectl port-forward svc/bookinfo-gateway-istio 8080:80

```

PS D:\istio-1.28.1> kubectl port-forward svc/bookinfo-gateway-istio 8080:80
Forwarding from 127.0.0.1:8080 -> 80
Forwarding from [::1]:8080 -> 80
Handling connection for 8080
Handling connection for 8080

```

13. access to the Bookinfo productpage

The screenshot shows a web browser window with the title bar "Simple Bookstore App". The address bar displays "localhost:8080/productpage". Below the address bar, there are several icons for various services like Gmail, YouTube, and IELTS. The main content area has a dark header with "BookInfo Sample" on the left and a "Sign in" button on the right. The main title "The Comedy of Errors" is displayed in large blue text. A summary text follows: "Wikipedia Summary: The Comedy of Errors is one of William Shakespeare's early plays. It is his shortest and one of his most farcical comedies, with a major part of the humour coming from slapstick and mistaken identity, in addition to puns and word play." Below the summary is a link "Learn more about Istio →". The section "Book Details" is titled "Book Details" and contains a table with the following data:

ISBN-10	Publisher	Pages	Type	Language
1234567890	PublisherA	200	paperback	English

The section "Book Reviews" is titled "Book Reviews" and shows two five-star reviews: "An extremely" and "Absolutely fun and".

14. Install Kiali and the other addons

```
kubectl apply -f samples/addons/kiali.yaml  
kubectl rollout status deployment/kiali -n istio-system
```

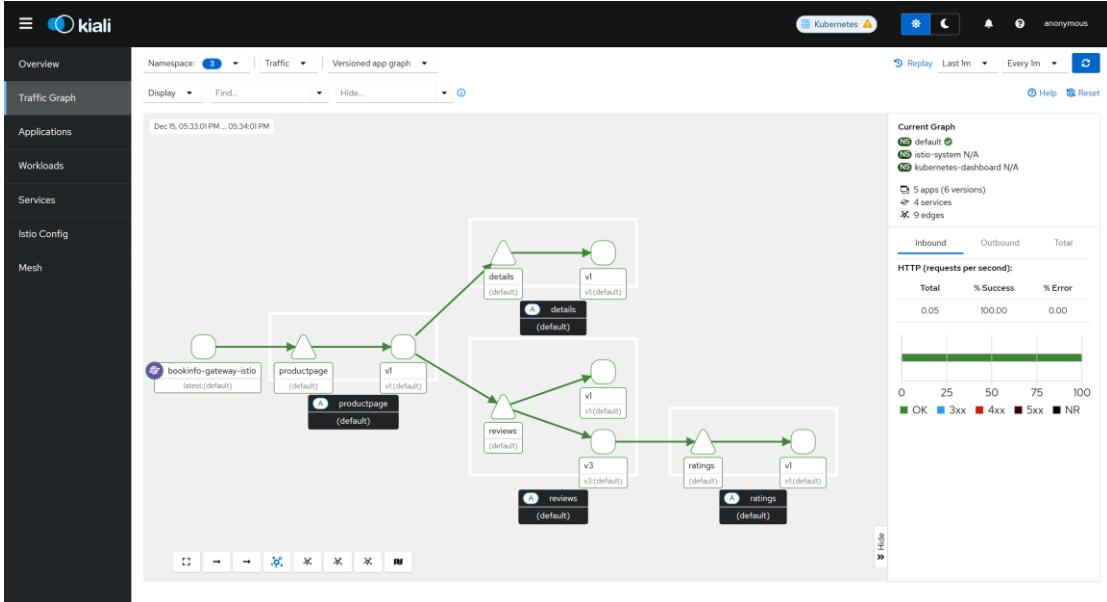
15. Access the Kiali dashboard

```
istioctl dashboard kiali  
  
PS D:\istio-1.28.1> kubectl apply -f samples/addons/kiali.yaml  
serviceaccount/kiali created  
configmap/kiali created  
clusterrole.rbac.authorization.k8s.io/kiali created  
clusterrolebinding.rbac.authorization.k8s.io/kiali created  
service/kiali created  
deployment.apps/kiali created  
PS D:\istio-1.28.1> kubectl rollout status deployment/kiali -n istio-system  
Waiting for deployment "kiali" rollout to finish: 0 of 1 updated replicas are available...  
deployment "kiali" successfully rolled out  
PS D:\istio-1.28.1> istioctl dashboard kiali  
http://localhost:20001/kiali
```

16. In the left navigation menu, select *Graph* and in the *Namespace* drop down, select *default*.

Then send 100 requests to the productpage service

```
PS D:\istio-1.28.1> 1..100 | ForEach-Object { curl.exe -s http://localhost:8080/productpage | Out-Null}
```



17. Delete the Bookinfo sample application and its configuration

```
kubectl delete -f samples/addons  
istioctl uninstall -y --purge  
kubectl delete namespace istio-system  
kubectl label namespace default istio-injection-
```

```
PS D:\istio-1.28.1> kubectl delete -f samples/addons  
serviceaccount "kiali" deleted from istio-system namespace  
configmap "kiali" deleted from istio-system namespace  
clusterrole.rbac.authorization.k8s.io "kiali" deleted  
clusterrolebinding.rbac.authorization.k8s.io "kiali" deleted  
service "kiali" deleted from istio-system namespace  
deployment.apps "kiali" deleted from istio-system namespace  
serviceaccount "prometheus" deleted from istio-system namespace  
configmap "prometheus" deleted from istio-system namespace  
clusterrole.rbac.authorization.k8s.io "prometheus" deleted  
clusterrolebinding.rbac.authorization.k8s.io "prometheus" deleted
```

```
PS D:\istio-1.28.1> istioctl uninstall -y --purge
All Istio resources will be pruned from the cluster

Removed apps/v1, Kind=Deployment/istiod.istio-system.
Removed /v1, Kind=Service/istiod.istio-system.
Removed /v1, Kind=Service/istiod-revision-tag-default.istio-system.
Removed /v1, Kind=ConfigMap/istio.istio-system.
Removed /v1, Kind=ConfigMap/istio-sidecar-injector.istio-system.
Removed /v1, Kind=ConfigMap/values.istio-system.
Removed /v1, Kind=Pod/istiod-7865c9494-pb52h.istio-system.
Removed /v1, Kind=ServiceAccount/istio-reader-service-account.istio-system.
Removed /v1, Kind=ServiceAccount/istiod.istio-system.
Removed rbac.authorization.k8s.io/v1, Kind=RoleBinding/istiod.istio-system.
Removed rbac.authorization.k8s.io/v1, Kind=Role/istiod.istio-system.
Removed admissionregistration.k8s.io/v1, Kind=MutatingWebhookConfiguration/istio-revision-tag-default..
Removed admissionregistration.k8s.io/v1, Kind=MutatingWebhookConfiguration/istio-sidecar-injector..
Removed admissionregistration.k8s.io/v1, Kind=ValidatingWebhookConfiguration/istio-validator-istio-system..
Removed admissionregistration.k8s.io/v1, Kind=ValidatingWebhookConfiguration/istiod-default-validator..

PS D:\istio-1.28.1> kubectl delete namespace istio-system
namespace "istio-system" deleted
PS D:\istio-1.28.1> kubectl label namespace default istio-injection-
namespace/default unlabeled
```