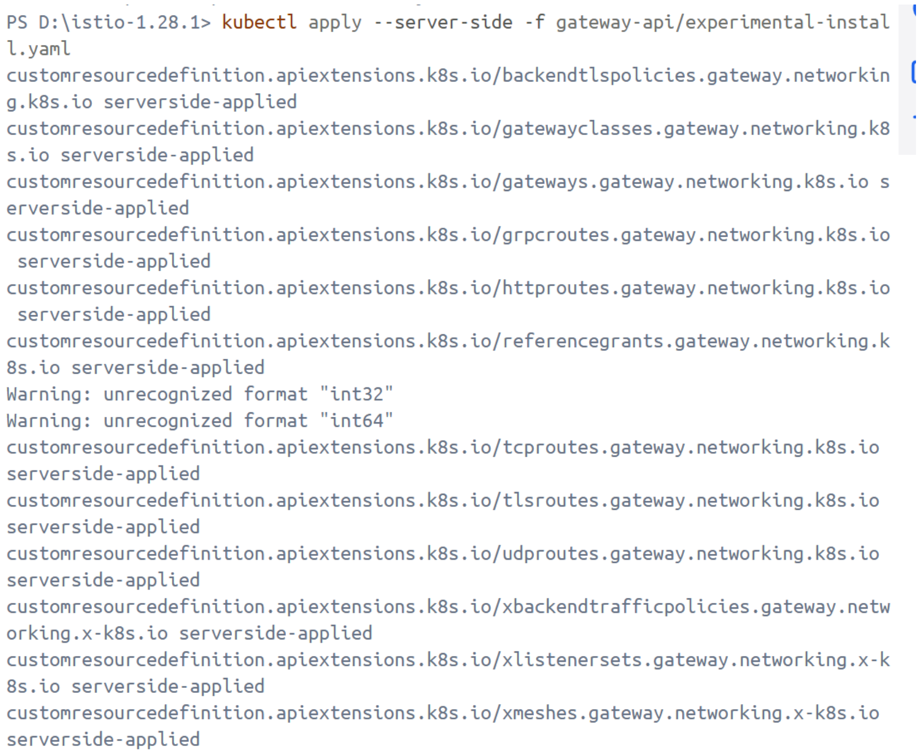
Content

Traffic Shifting

1. Preparation: install Kubernetes Gateway API CRDs

   kubectl apply --server-side -f https://github.com/kubernetes-sigs/gateway-api/releases/download/v1.4.0/experimental-install.yaml



```
PS D:\istio-1.28.1> kubectl apply --server-side -f gateway-api/experimental-instal
l.yaml
customresourcedefinition.apiextensions.k8s.io/backendtlspolicies.gateway.networkin
g.k8s.io serverside-applied
customresourcedefinition.apiextensions.k8s.io/gatewayclasses.gateway.networking.k8
s.io serverside-applied
customresourcedefinition.apiextensions.k8s.io/gateways.gateway.networking.k8s.io s
erverside-applied
customresourcedefinition.apiextensions.k8s.io/grpcroutes.gateway.networking.k8s.io
 serverside-applied
customresourcedefinition.apiextensions.k8s.io/httproutes.gateway.networking.k8s.io
 serverside-applied
customresourcedefinition.apiextensions.k8s.io/referencegrants.gateway.networking.k
8s.io serverside-applied
Warning: unrecognized format "int32"
Warning: unrecognized format "int64"
customresourcedefinition.apiextensions.k8s.io/tcproutes.gateway.networking.k8s.io
serverside-applied
customresourcedefinition.apiextensions.k8s.io/tlsroutes.gateway.networking.k8s.io
serverside-applied
customresourcedefinition.apiextensions.k8s.io/udproutes.gateway.networking.k8s.io
serverside-applied
customresourcedefinition.apiextensions.k8s.io/xbackendtrafficpolicies.gateway.netw
orking.x-k8s.io serverside-applied
customresourcedefinition.apiextensions.k8s.io/xlistenersets.gateway.networking.x-k
8s.io serverside-applied
customresourcedefinition.apiextensions.k8s.io/xmeshes.gateway.networking.x-k8s.io
serverside-applied
```

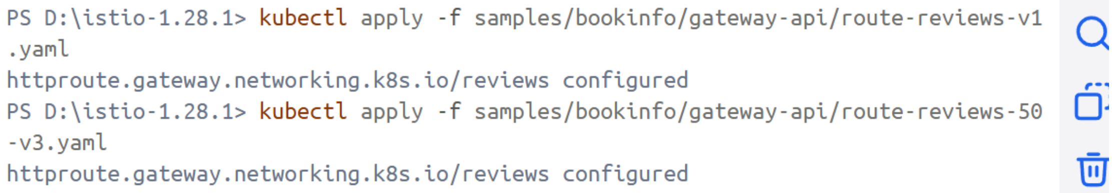
2. Apply weight-based routing

2.1. Route all traffic to the v1 version

kubectl apply -f samples/bookinfo/gateway-api/route-reviews-v1.yaml

2.2. Transfer 50% of the traffic from reviews:v1 to reviews:v3

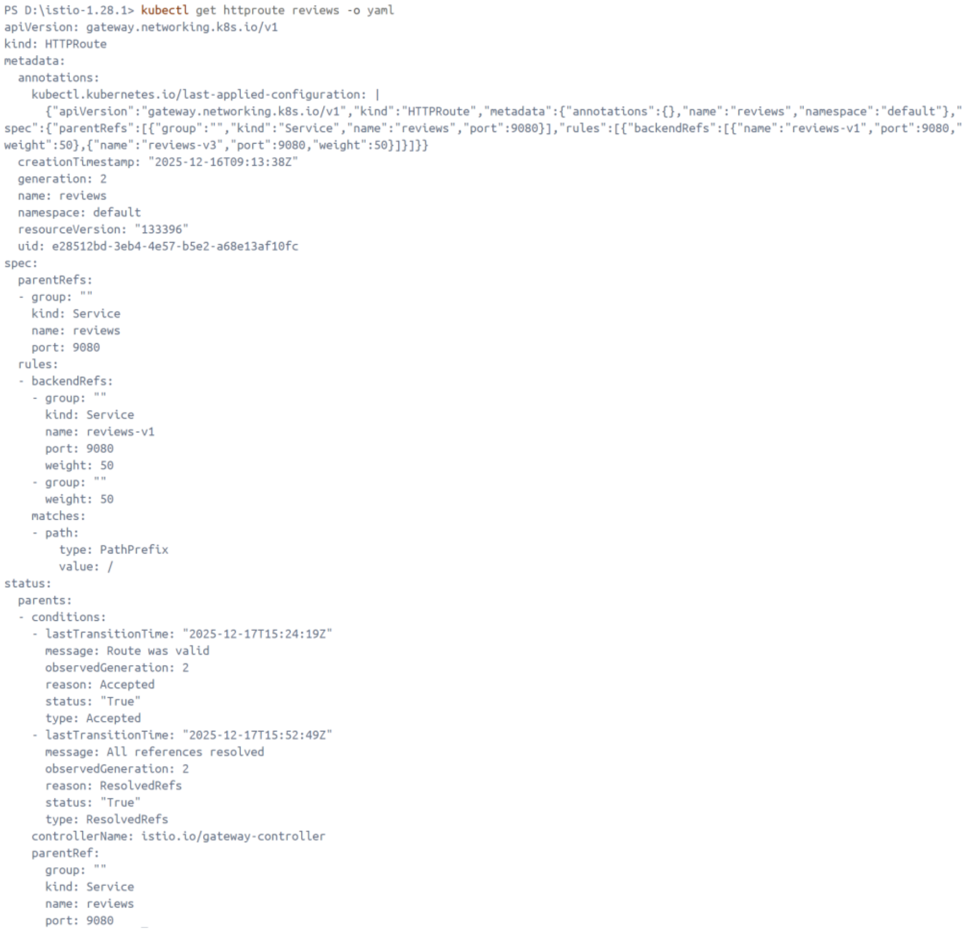
kubectl apply -f samples/bookinfo/gateway-api/route-reviews-50-v3.yaml



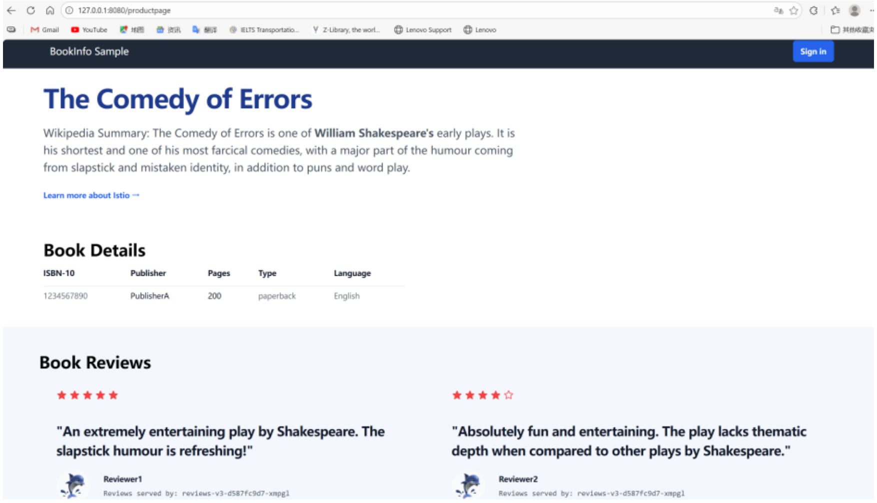
```
PS D:\istio-1.28.1> kubectl apply -f samples/bookinfo/gateway-api/route-reviews-v1
.yaml
httproute.gateway.networking.k8s.io/reviews configured
PS D:\istio-1.28.1> kubectl apply -f samples/bookinfo/gateway-api/route-reviews-50
-v3.yaml
httproute.gateway.networking.k8s.io/reviews configured
```

2.3. Wait a few seconds for the new rules to propagate and then confirm the rule was replaced

kubectl get httproute reviews -o yaml



```
PS D:\istio-1.28.1> kubectl get httproute reviews -o yaml
apiVersion: gateway.networking.k8s.io/v1
kind: HTTPRoute
metadata:
  annotations:
    kubectl.kubernetes.io/last-applied-configuration: |
      {"apiVersion":"gateway.networking.k8s.io/v1","kind":"HTTPRoute","metadata":{"annotations":{},"name":"reviews","namespace":"default"},"
spec":{"parentRefs":[{"group":"","kind":"Service","name":"reviews","port":9080}],"rules":[{"backendRefs":[{"name":"reviews-v1","port":9080,"
weight":50},{"name":"reviews-v3","port":9080,"weight":50}]}]}}
  creationTimestamp: "2025-12-16T09:13:38Z"
  generation: 2
  name: reviews
  namespace: default
  resourceVersion: "133396"
  uid: e28512bd-3eb4-4e57-b5e2-a68e13af10fc
spec:
  parentRefs:
  - group: ""
    kind: Service
    name: reviews
    port: 9080
  rules:
  - backendRefs:
    - group: ""
      kind: Service
      name: reviews-v1
      port: 9080
      weight: 50
    - group: ""
      weight: 50
    matches:
    - path:
        type: PathPrefix
        value: /
status:
  parents:
  - conditions:
    - lastTransitionTime: "2025-12-17T15:24:19Z"
      message: Route was valid
      observedGeneration: 2
      reason: Accepted
      status: "True"
      type: Accepted
    - lastTransitionTime: "2025-12-17T15:52:49Z"
      message: All references resolved
      observedGeneration: 2
      reason: ResolvedRefs
      status: "True"
      type: ResolvedRefs
    controllerName: istio.io/gateway-controller
    parentRef:
      group: ""
      kind: Service
      name: reviews
      port: 9080
```



The Comedy of Errors

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.

Learn more about Istio →

Book Details

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

Book Reviews

★★★★★

"An extremely entertaining play by Shakespeare. The slapstick humour is refreshing!"

Reviewer1
Reviews served by: reviews-v3-d587fc9d7-xmpgl

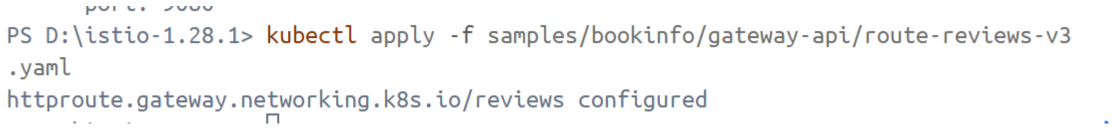
★★★★☆

"Absolutely fun and entertaining. The play lacks thematic depth when compared to other plays by Shakespeare."

Reviewer2
Reviews served by: reviews-v3-d587fc9d7-xmpgl

2.4. Route 100% of the traffic to reviews:v3

kubectl apply -f samples/bookinfo/gateway-api/route-reviews-v3.yaml



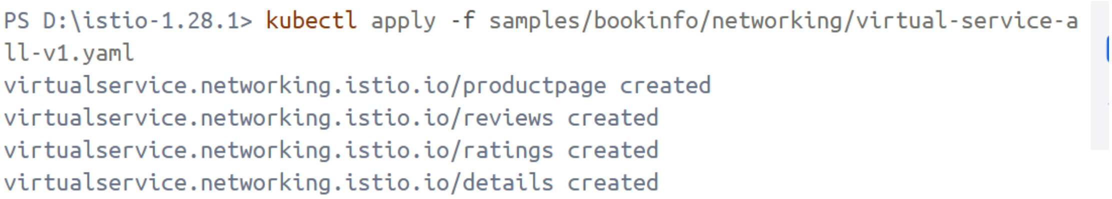
```
PS D:\istio-1.28.1> kubectl apply -f samples/bookinfo/gateway-api/route-reviews-v3
.yaml
httproute.gateway.networking.k8s.io/reviews configured
```

Request Routing

1. Route to version 1

1.1. Apply virtual services that will route all traffic to v1 of each microservice

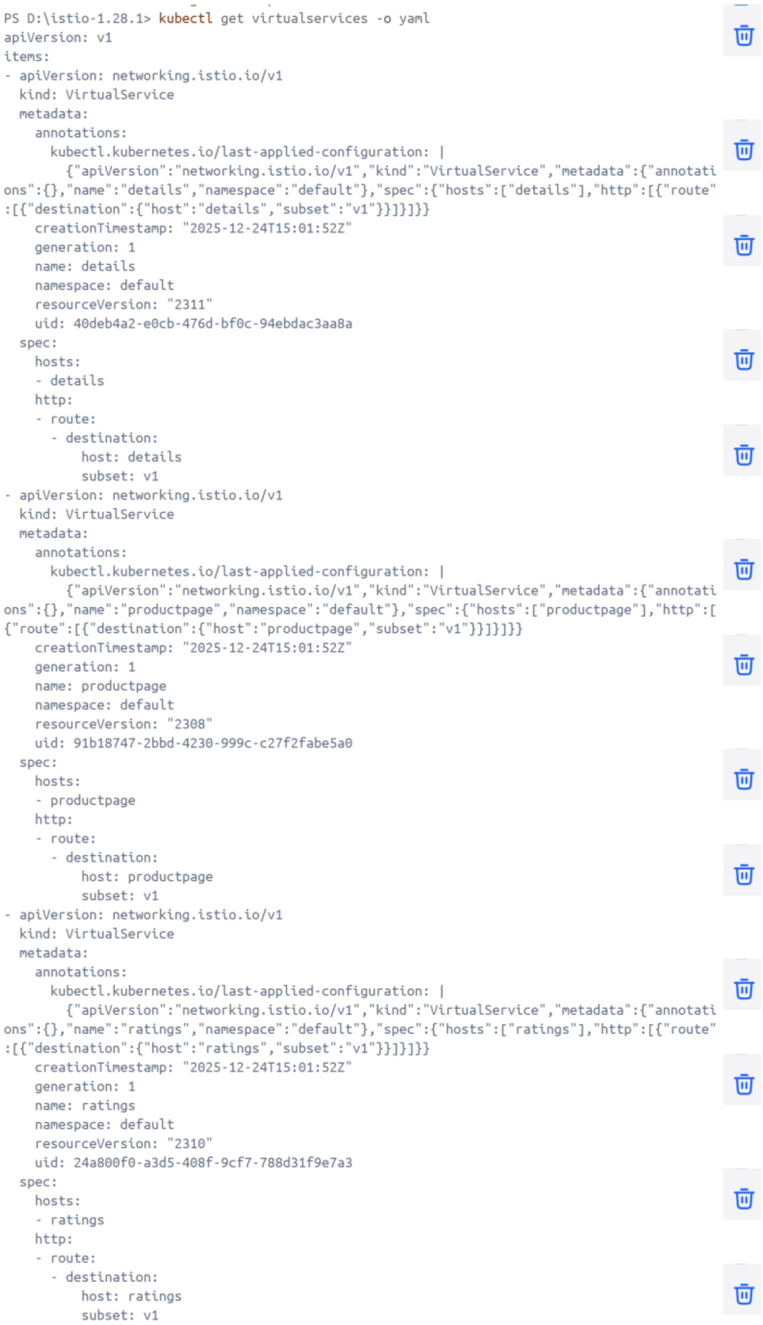
kubectl apply -f samples/bookinfo/networking/virtual-service-all-v1.yaml



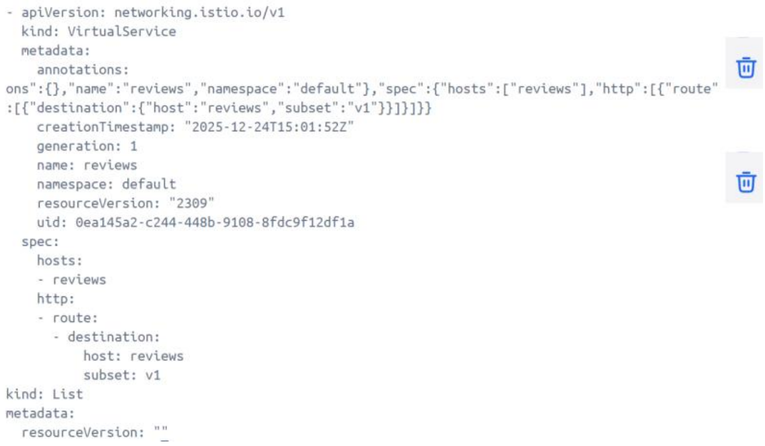
```
PS D:\istio-1.28.1> kubectl apply -f samples/bookinfo/networking/virtual-service-a
ll-v1.yaml
virtualservice.networking.istio.io/productpage created
virtualservice.networking.istio.io/reviews created
virtualservice.networking.istio.io/ratings created
virtualservice.networking.istio.io/details created
```

1.2. Display the defined routes

kubectl get virtualservices -o yaml



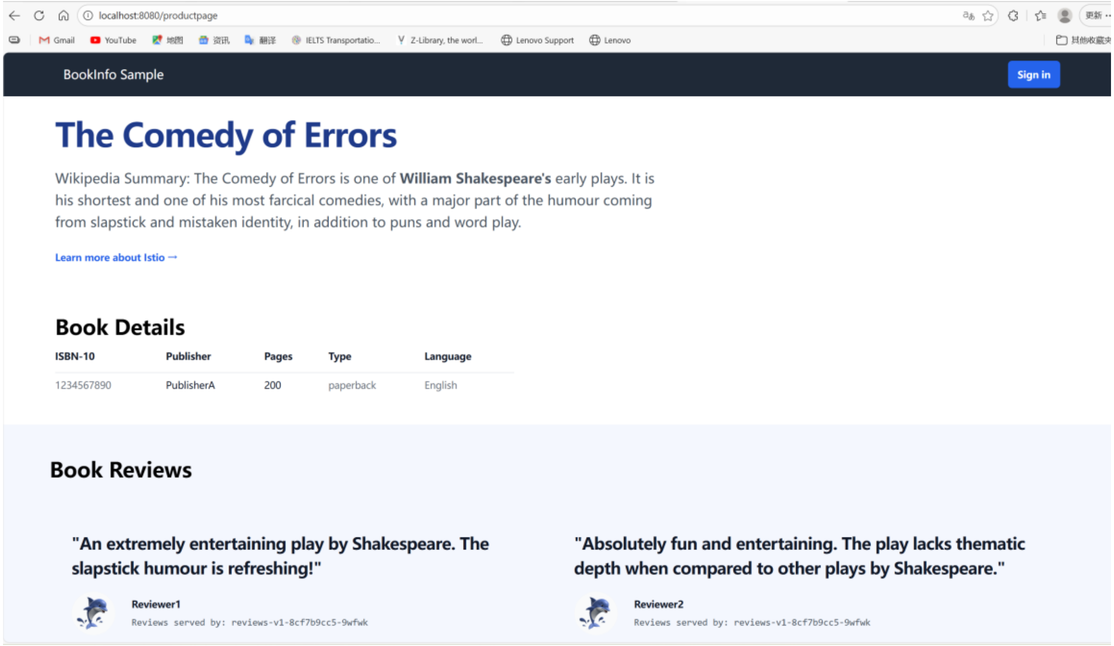
```
PS D:\istio-1.28.1> kubectl get virtualservices -o yaml
apiVersion: v1
items:
- apiVersion: networking.istio.io/v1
  kind: VirtualService
  metadata:
    annotations:
      kubectl.kubernetes.io/last-applied-configuration: |
        {"apiVersion":"networking.istio.io/v1","kind":"VirtualService","metadata":{"annotati
ons":{},"name":"details","namespace":"default"},"spec":{"hosts":["details"],"http":[{"route"
:[{"destination":{"host":"details","subset":"v1"}}]}]}}
    creationTimestamp: "2025-12-24T15:01:52Z"
    generation: 1
    name: details
    namespace: default
    resourceVersion: "2311"
    uid: 40deb4a2-e0cb-476d-bf0c-94ebdac3aa8a
  spec:
    hosts:
    - details
    http:
    - route:
      - destination:
          host: details
          subset: v1
- apiVersion: networking.istio.io/v1
  kind: VirtualService
  metadata:
    annotations:
      kubectl.kubernetes.io/last-applied-configuration: |
        {"apiVersion":"networking.istio.io/v1","kind":"VirtualService","metadata":{"annotati
ons":{},"name":"productpage","namespace":"default"},"spec":{"hosts":["productpage"],"http":[
{"route":[{"destination":{"host":"productpage","subset":"v1"}}]}]}}
    creationTimestamp: "2025-12-24T15:01:52Z"
    generation: 1
    name: productpage
    namespace: default
    resourceVersion: "2308"
    uid: 91b18747-2bbd-4230-999c-c27f2fabe5a0
  spec:
    hosts:
    - productpage
    http:
    - route:
      - destination:
          host: productpage
          subset: v1
- apiVersion: networking.istio.io/v1
  kind: VirtualService
  metadata:
    annotations:
      kubectl.kubernetes.io/last-applied-configuration: |
        {"apiVersion":"networking.istio.io/v1","kind":"VirtualService","metadata":{"annotati
ons":{},"name":"ratings","namespace":"default"},"spec":{"hosts":["ratings"],"http":[{"route"
:[{"destination":{"host":"ratings","subset":"v1"}}]}]}}
    creationTimestamp: "2025-12-24T15:01:52Z"
    generation: 1
    name: ratings
    namespace: default
    resourceVersion: "2310"
    uid: 24a800f0-a3d5-408f-9cf7-788d31f9e7a3
  spec:
    hosts:
    - ratings
    http:
    - route:
      - destination:
          host: ratings
          subset: v1
```



```
- apiVersion: networking.istio.io/v1
  kind: VirtualService
  metadata:
    annotations:
ons":{},"name":"reviews","namespace":"default"},"spec":{"hosts":["reviews"],"http":[{"route"
:[{"destination":{"host":"reviews","subset":"v1"}}]}]}}
    creationTimestamp: "2025-12-24T15:01:52Z"
    generation: 1
    name: reviews
    namespace: default
    resourceVersion: "2309"
    uid: 0ea145a2-c244-448b-9108-8fdc9f12df1a
  spec:
    hosts:
    - reviews
    http:
    - route:
      - destination:
          host: reviews
          subset: v1
kind: List
metadata:
  resourceVersion: ""
```

2. Test the new routing configuration

Notice that the reviews part of the page displays with no rating stars, no matter how many times we refresh. This is because we configured Istio to route all traffic for the reviews service to the version reviews:v1 and this version of the service does not access the star ratings service.



The Comedy of Errors

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.

Learn more about Istio →

Book Details

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

Book Reviews

"An extremely entertaining play by Shakespeare. The slapstick humour is refreshing!"

Reviewer1
Reviews served by: reviews-v1-8cf7b9cc5-9wfwk

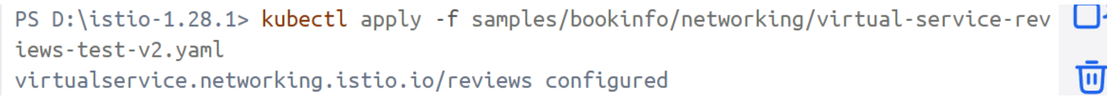
"Absolutely fun and entertaining. The play lacks thematic depth when compared to other plays by Shakespeare."

Reviewer2
Reviews served by: reviews-v1-8cf7b9cc5-9wfwk

3. Route based on user identity

3.1. Enable user-based routing

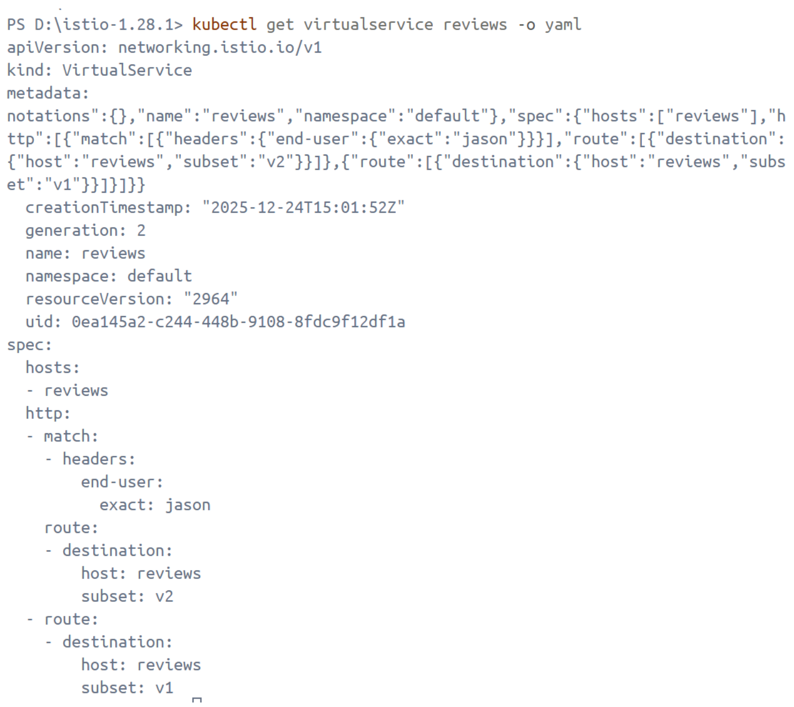
kubectl apply -f samples/bookinfo/networking/virtual-service-reviews-test-v2.yaml



```
PS D:\istio-1.28.1> kubectl apply -f samples/bookinfo/networking/virtual-service-rev
iews-test-v2.yaml
virtualservice.networking.istio.io/reviews configured
```

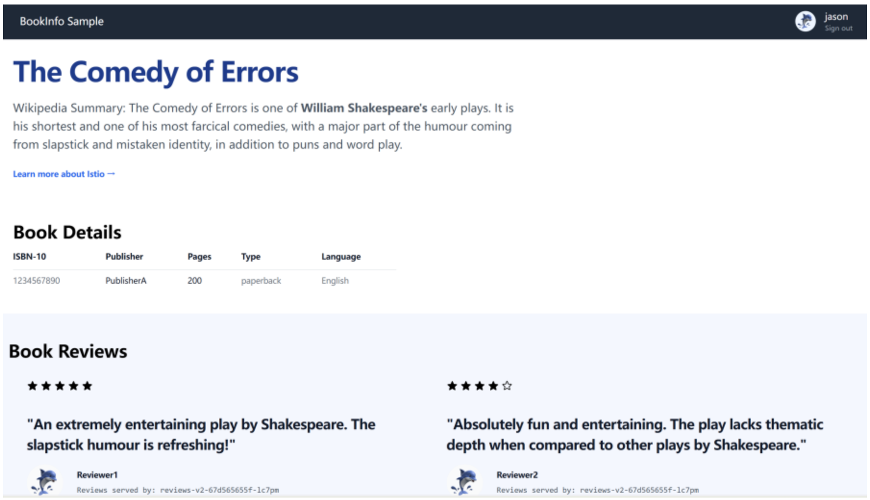
3.2. Confirm the rule is created

kubectl get virtualservice reviews -o yaml



```
PS D:\istio-1.28.1> kubectl get virtualservice reviews -o yaml
apiVersion: networking.istio.io/v1
kind: VirtualService
metadata:
notations":{},"name":"reviews","namespace":"default"},"spec":{"hosts":["reviews"],"h
ttp":[{"match":[{"headers":{"end-user":{"exact":"jason"}}}],"route":[{"destination":
{"host":"reviews","subset":"v2"}}]},{"route":[{"destination":{"host":"reviews","subs
et":"v1"}}]}]}}
  creationTimestamp: "2025-12-24T15:01:52Z"
  generation: 2
  name: reviews
  namespace: default
  resourceVersion: "2964"
  uid: 0ea145a2-c244-448b-9108-8fdc9f12df1a
spec:
  hosts:
  - reviews
  http:
  - match:
    - headers:
        end-user:
          exact: jason
    route:
    - destination:
        host: reviews
        subset: v2
  - route:
    - destination:
        host: reviews
        subset: v1
```

3.3. On the /productpage of the Bookinfo app, log in as user jason. The star ratings appear next to each review.



The Comedy of Errors

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.

Learn more about Istio →

Book Details

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

Book Reviews

★★★★★

"An extremely entertaining play by Shakespeare. The slapstick humour is refreshing!"

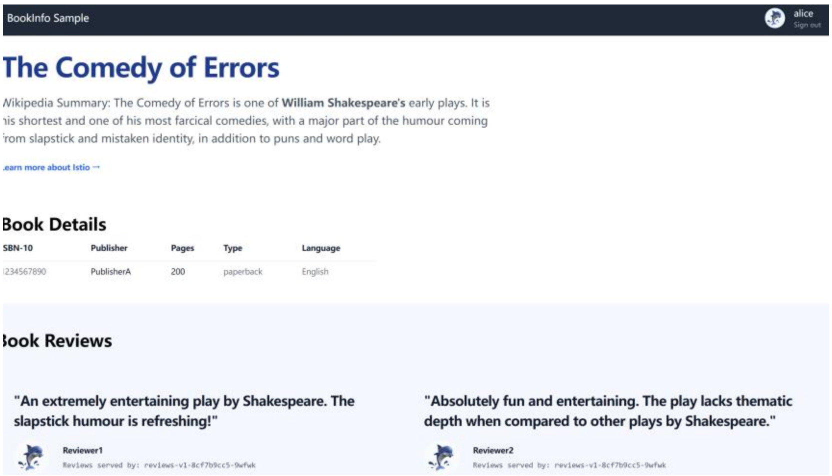
Reviewer1
Reviews served by: reviews-v2-67d565655f-lc7pm

★★★★☆

"Absolutely fun and entertaining. The play lacks thematic depth when compared to other plays by Shakespeare."

Reviewer2
Reviews served by: reviews-v2-67d565655f-lc7pm

3.4. Log in as another user and the stars are gone. This is because traffic is routed to reviews:v1 for all users except Jason.



The Comedy of Errors

Vikipedia Summary: The Comedy of Errors is one of **William Shakespeare's** early plays. It is nis shortest and one of his most farcical comedies, with a major part of the humour coming rom slapstick and mistaken identity, in addition to puns and word play.

earn more about Istio →

Book Details

SBN-10	Publisher	Pages	Type	Language
234567890	PublisherA	200	paperback	English

3ook Reviews

"An extremely entertaining play by Shakespeare. The slapstick humour is refreshing!"

Reviewer1
Reviews served by: reviews-v1-8cf7b9cc5-9wfwk

"Absolutely fun and entertaining. The play lacks thematic depth when compared to other plays by Shakespeare."

Reviewer2
Reviews served by: reviews-v1-8cf7b9cc5-9wfwk

Fault Injection

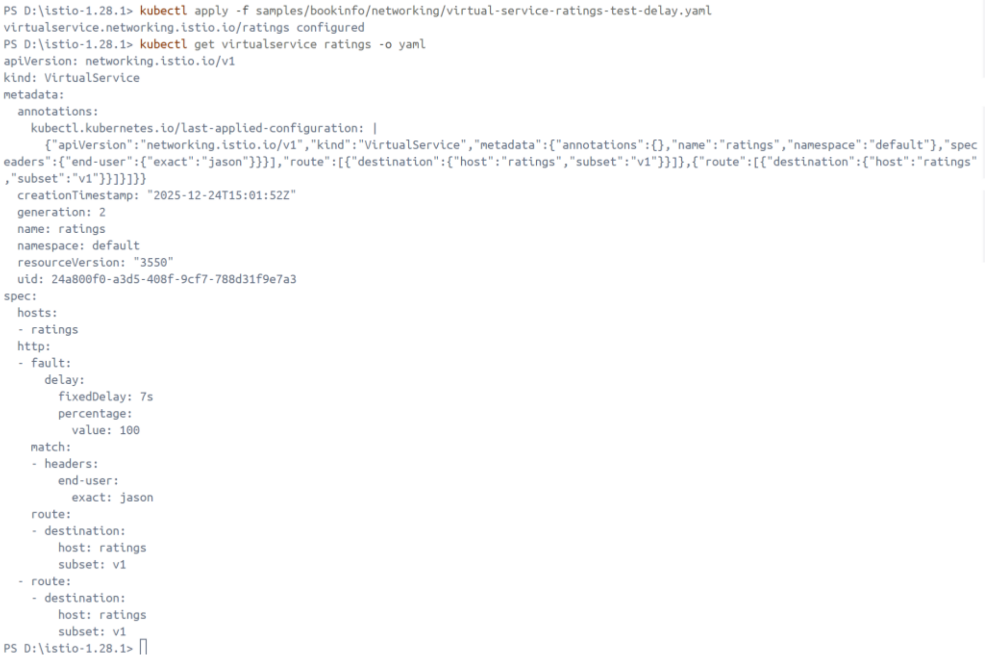
1. Injecting an HTTP delay fault

1.1. Create a fault injection rule to delay traffic coming from the test user jason

kubectl apply -f samples/bookinfo/networking/virtual-service-ratings-test-delay.yaml

1.2. Confirm the rule was created

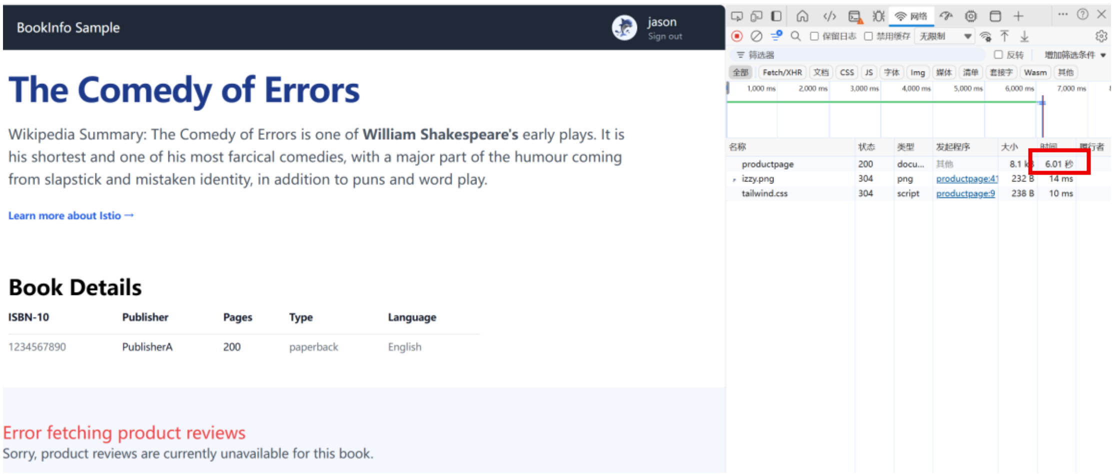
kubectl get virtualservice ratings -o yaml



```
PS D:\istio-1.28.1> kubectl apply -f samples/bookinfo/networking/virtual-service-ratings-test-delay.yaml
virtualservice.networking.istio.io/ratings configured
PS D:\istio-1.28.1> kubectl get virtualservice ratings -o yaml
apiVersion: networking.istio.io/v1
kind: VirtualService
metadata:
  annotations:
    kubectl.kubernetes.io/last-applied-configuration: |
      {"apiVersion":"networking.istio.io/v1","kind":"VirtualService","metadata":{"annotations":{},"name":"ratings","namespace":"default"},"spec
eaders":{"end-user":{"exact":"jason"}}}],"route":[{"destination":{"host":"ratings","subset":"v1"}}]},{"route":[{"destination":{"host":"ratings"
,"subset":"v1"}}]}]}}
  creationTimestamp: "2025-12-24T15:01:52Z"
  generation: 2
  name: ratings
  namespace: default
  resourceVersion: "3550"
  uid: 24a800f0-a3d5-408f-9cf7-788d31f9e7a3
spec:
  hosts:
  - ratings
  http:
  - fault:
      delay:
        fixedDelay: 7s
        percentage:
          value: 100
    match:
    - headers:
        end-user:
          exact: jason
    route:
    - destination:
        host: ratings
        subset: v1
  - route:
    - destination:
        host: ratings
        subset: v1
PS D:\istio-1.28.1> 
```

2. Testing the delay configuration

On the /productpage web page, log in as user Jason. There is a problem that the Reviews section displays an error message: “Sorry, product reviews are currently unavailable for this book.” Open the *Developer Tools* menu and open the Network tab. It can be seen that the page actually loads in about 6 seconds.



The Comedy of Errors

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.

Learn more about Istio →

Book Details

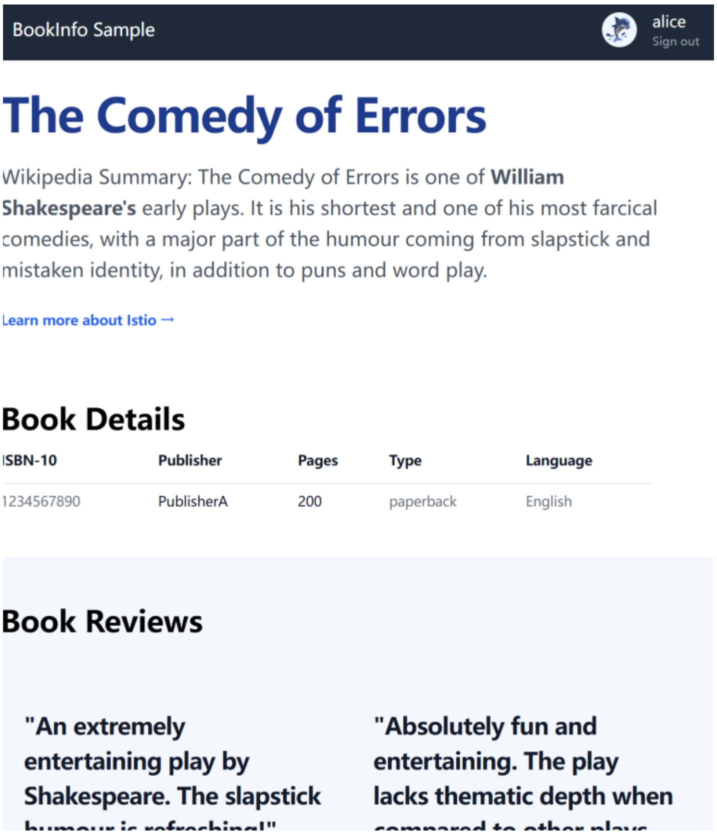
ISBN-10	Publisher	Pages	Type	Language
1234567890	PublisherA	200	paperback	English

Error fetching product reviews

Sorry, product reviews are currently unavailable for this book.

名称	状态	类型	发起程序	大小	时间	履行者
productpage	200	docu...	其他	8.1 kB	6.01 秒	
izzy.png	304	png	productpage:41	232 B	14 ms	
tailwind.css	304	script	productpage:9	238 B	10 ms	

While logging in as any other user, there is no problem.



The Comedy of Errors

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.

Book Details

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

Book Reviews

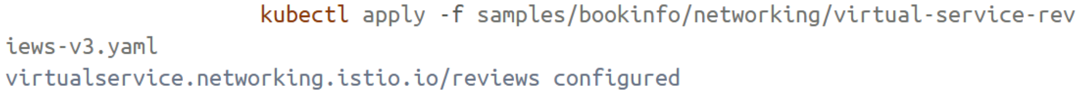
"An extremely entertaining play by Shakespeare. The slapstick humour is refreshing!"

"Absolutely fun and entertaining. The play lacks thematic depth when compared to other plays

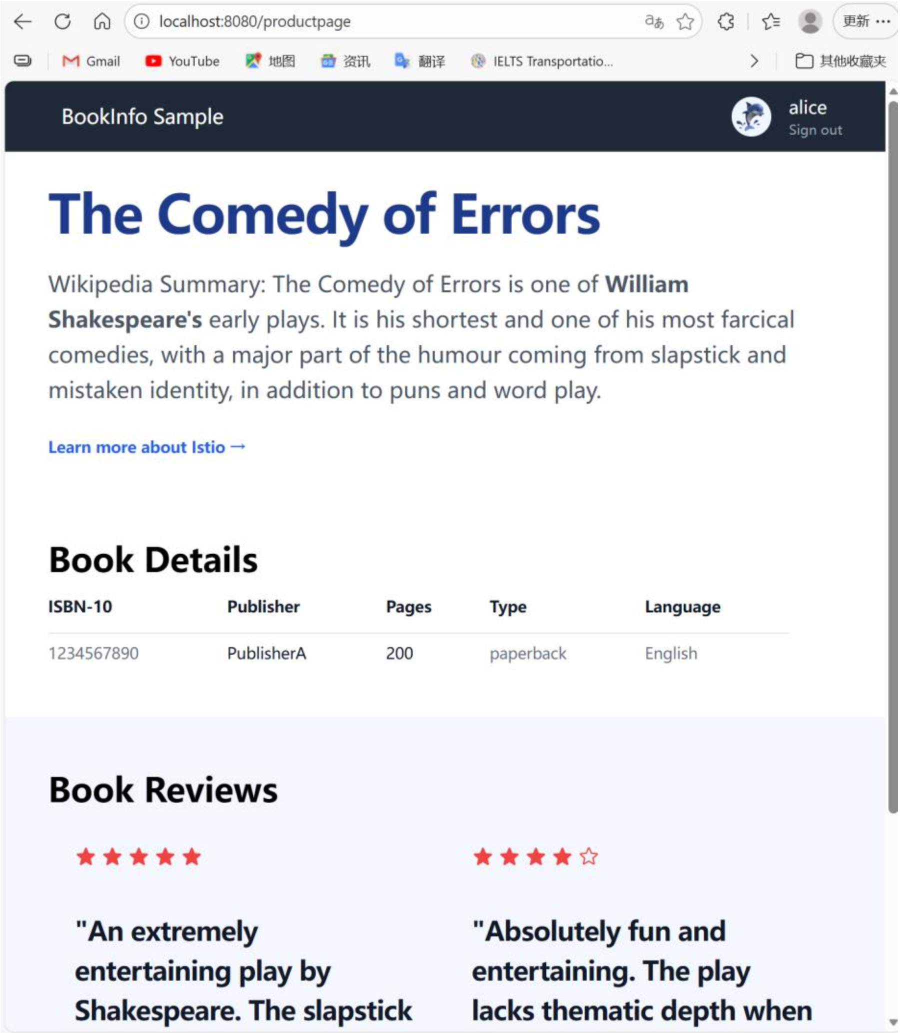
Reason: As expected, the 7s delay I introduced doesn't affect the reviews service because the timeout between the reviews and ratings service is hard-coded at 10s. However, there is also a hard-coded timeout between the productpage and the reviews service, coded as 3s + 1 retry for 6s total. As a result, the productpage call to reviews times out prematurely and throws an error after 6s.

Fix bug: transfer all traffics to reviews:v3

kubectl apply -f samples/bookinfo/networking/virtual-service-reviews-v3.yaml



```
                          kubectl apply -f samples/bookinfo/networking/virtual-service-reviews-v3.yaml
virtualservice.networking.istio.io/reviews configured
```



The Comedy of Errors

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.

Book Details

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

Book Reviews

★★★★★

"An extremely entertaining play by Shakespeare. The slapstick

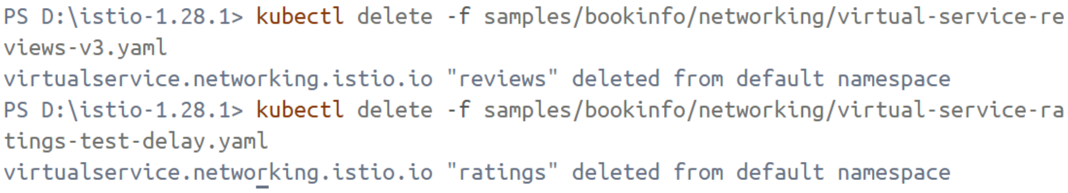
★★★★☆

"Absolutely fun and entertaining. The play lacks thematic depth when

Then delete the following rules and add necessary rules to make preparation for another way.

kubectl delete -f samples/bookinfo/networking/virtual-service-reviews-v3.yaml

kubectl delete -f samples/bookinfo/networking/virtual-service-ratings=test=delay.yaml



```
PS D:\istio-1.28.1> kubectl delete -f samples/bookinfo/networking/virtual-service-reviews-v3.yaml
virtualservice.networking.istio.io "reviews" deleted from default namespace
PS D:\istio-1.28.1> kubectl delete -f samples/bookinfo/networking/virtual-service-ratings-test-delay.yaml
virtualservice.networking.istio.io "ratings" deleted from default namespace
```

kubectl apply -f samples/bookinfo/networking/virtual-service-all-v1.yaml

kubectl apply -f samples/bookinfo/networking/virtual-service-reviews-test-v2.yaml

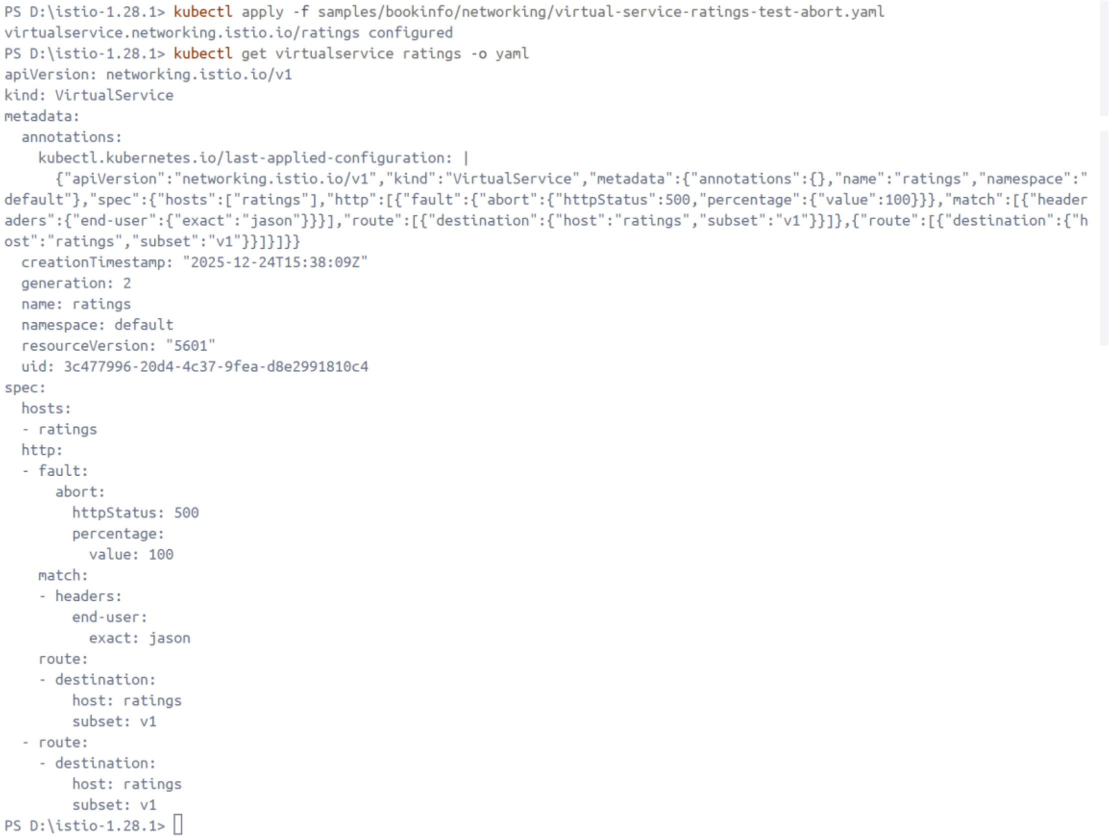
Injecting an HTTP abort fault

1. Create a fault injection rule to send an HTTP abort for user jason

   kubectl apply -f samples/bookinfo/networking/virtual-service-ratings-test-abort.yaml

2. Confirm the rule was created

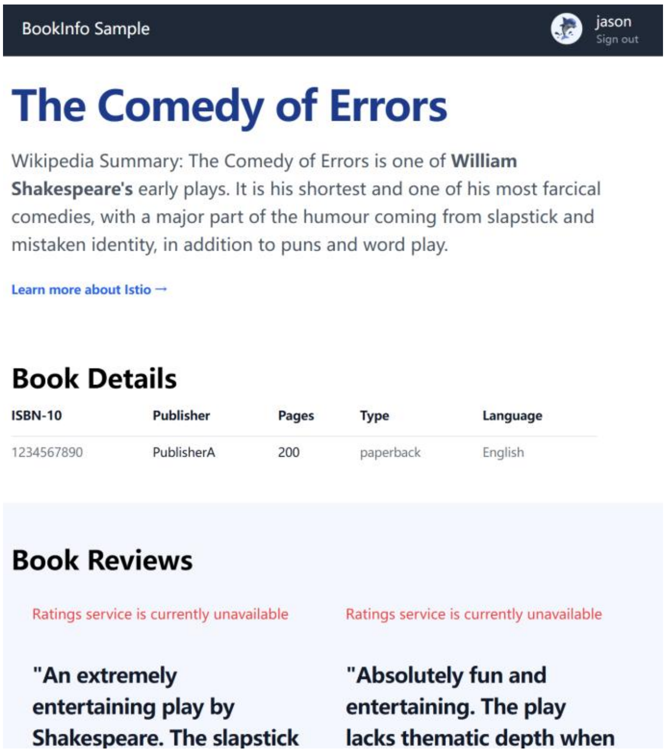
   kubectl get virtualservice ratings -o yaml



```
PS D:\istio-1.28.1> kubectl apply -f samples/bookinfo/networking/virtual-service-ratings-test-abort.yaml
virtualservice.networking.istio.io/ratings configured
PS D:\istio-1.28.1> kubectl get virtualservice ratings -o yaml
apiVersion: networking.istio.io/v1
kind: VirtualService
metadata:
  annotations:
    kubectl.kubernetes.io/last-applied-configuration: |
      {"apiVersion":"networking.istio.io/v1","kind":"VirtualService","metadata":{"annotations":{},"name":"ratings","namespace":"
default"},"spec":{"hosts":["ratings"],"http":[{"fault":{"abort":{"httpStatus":500,"percentage":{"value":100}},"match":[{"header
aders":{"end-user":{"exact":"jason"}}}],"route":[{"destination":{"host":"ratings","subset":"v1"}}]},{"route":[{"destination":{"h
ost":"ratings","subset":"v1"}}]}]}}
  creationTimestamp: "2025-12-24T15:38:09Z"
  generation: 2
  name: ratings
  namespace: default
  resourceVersion: "5601"
  uid: 3c477996-20d4-4c37-9fea-d8e2991810c4
spec:
  hosts:
  - ratings
  http:
  - fault:
      abort:
        httpStatus: 500
        percentage:
          value: 100
    match:
    - headers:
        end-user:
          exact: jason
    route:
    - destination:
        host: ratings
        subset: v1
  - route:
    - destination:
        host: ratings
        subset: v1
PS D:\istio-1.28.1> 
```

3. Test the abort configuration

   On the /productpage, log in as user jason. The page loads immediately and the "Ratings service is currently unavailable" message appears. However, while logging in another user, there is no problem.



BookInfo Sample

jason
Sign out

The Comedy of Errors

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.

Learn more about Istio →

Book Details

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

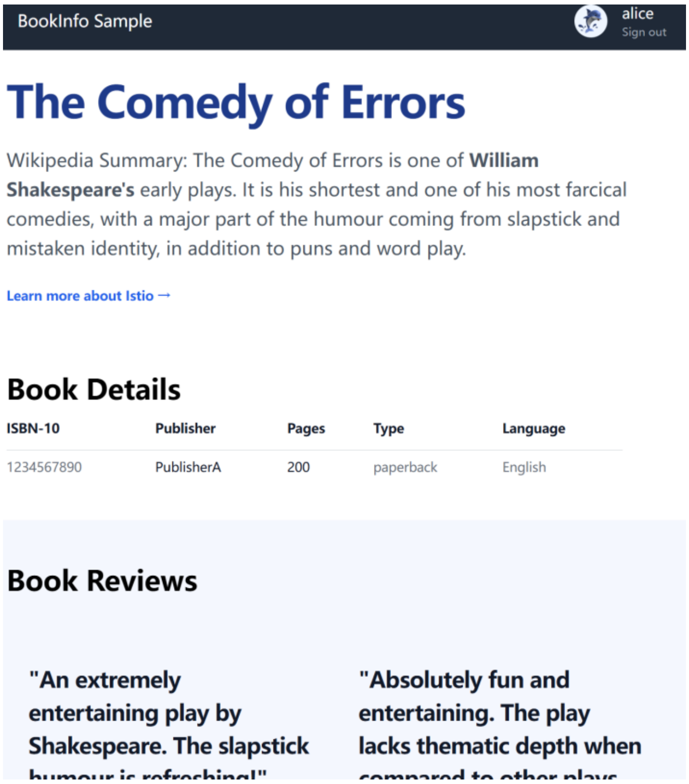
Book Reviews

Ratings service is currently unavailable

"An extremely entertaining play by Shakespeare. The slapstick

Ratings service is currently unavailable

"Absolutely fun and entertaining. The play lacks thematic depth when



BookInfo Sample

alice
Sign out

The Comedy of Errors

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.

Learn more about Istio →

Book Details

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

Book Reviews

"An extremely entertaining play by Shakespeare. The slapstick humour is refreshing!"

"Absolutely fun and entertaining. The play lacks thematic depth when compared to other plays

Clean up

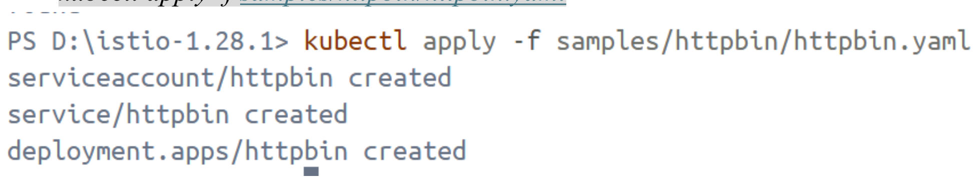
Remove the application routing rules

```
kubectl delete -f samples/bookinfo/networking/virtual-service-ratings-test-abort.yaml
kubectl delete -f samples/bookinfo/networking/virtual-service-all-v1.yaml
kubectl delete -f samples/bookinfo/networking/virtual-service-reviews-test-v2.yaml
```

Circuit Breaking

1. Preparation: deploy the httpbin service

```
kubectl apply -f samples/httpbin/httpbin.yaml
```

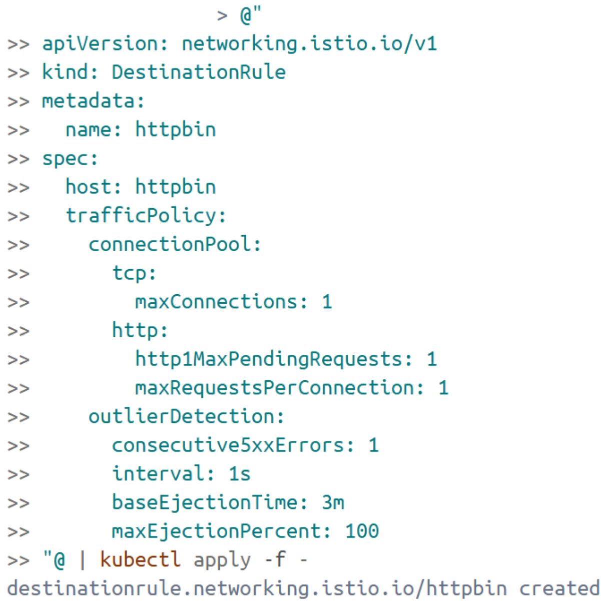


```
PS D:\istio-1.28.1> kubectl apply -f samples/httpbin/httpbin.yaml
serviceaccount/httpbin created
service/httpbin created
deployment.apps/httpbin created
```

2. Configurating the circuit breaker

2.1. Create a destination rule to apply circuit breaking settings when calling the httpbin service

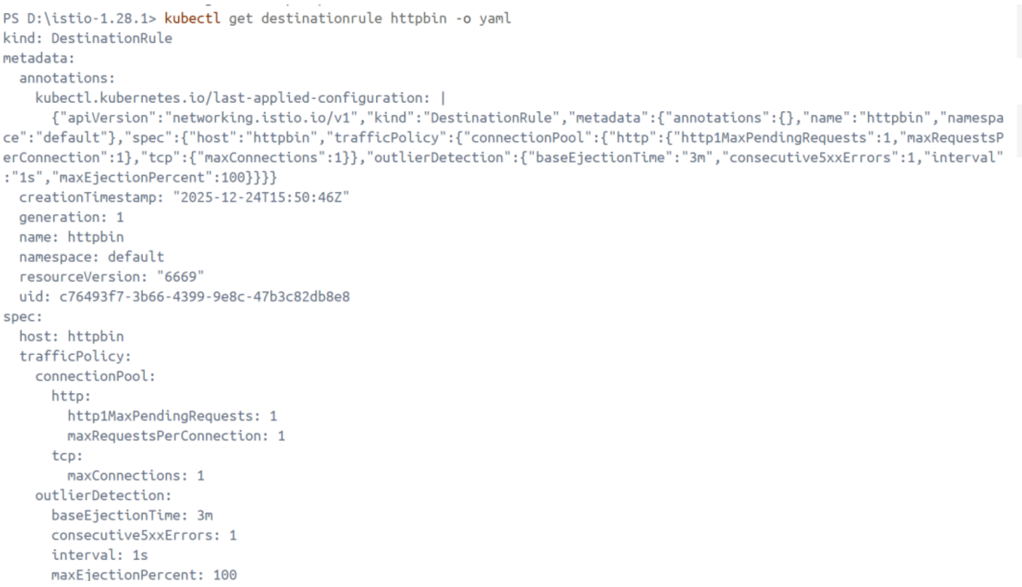
```
@"
apiVersion: networking.istio.io/v1
kind: DestinationRule
metadata:
  name: httpbin
spec:
  host: httpbin
  trafficPolicy:
    connectionPool:
      tcp:
        maxConnections: 1
      http:
        http1MaxPendingRequests: 1
        maxRequestsPerConnection: 1
    outlierDetection:
      consecutive5xxErrors: 1
      interval: 1s
      baseEjectionTime: 3m
      maxEjectionPercent: 100
"@ | kubectl apply -f –
```



```
                > @"
>> apiVersion: networking.istio.io/v1
>> kind: DestinationRule
>> metadata:
>>   name: httpbin
>> spec:
>>   host: httpbin
>>   trafficPolicy:
>>     connectionPool:
>>       tcp:
>>         maxConnections: 1
>>       http:
>>         http1MaxPendingRequests: 1
>>         maxRequestsPerConnection: 1
>>     outlierDetection:
>>       consecutive5xxErrors: 1
>>       interval: 1s
>>       baseEjectionTime: 3m
>>       maxEjectionPercent: 100
>> "@ | kubectl apply -f -
destinationrule.networking.istio.io/httpbin created
```

2.2. Verify the destination rule was created correctly

kubectl get destinationrule httpbin -o yaml

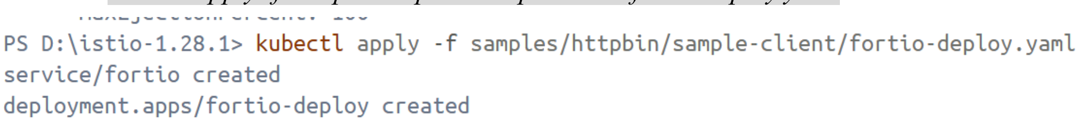


```
PS D:\istio-1.28.1> kubectl get destinationrule httpbin -o yaml
kind: DestinationRule
metadata:
  annotations:
    kubectl.kubernetes.io/last-applied-configuration: |
      {"apiVersion":"networking.istio.io/v1","kind":"DestinationRule","metadata":{"annotations":{},"name":"httpbin","namespa
ce":"default"},"spec":{"host":"httpbin","trafficPolicy":{"connectionPool":{"http":{"http1MaxPendingRequests":1,"maxRequestsP
erConnection":1},"tcp":{"maxConnections":1}},"outlierDetection":{"baseEjectionTime":"3m","consecutive5xxErrors":1,"interval"
:"1s","maxEjectionPercent":100}}}}
  creationTimestamp: "2025-12-24T15:50:46Z"
  generation: 1
  name: httpbin
  namespace: default
  resourceVersion: "6669"
  uid: c76493f7-3b66-4399-9e8c-47b3c82db8e8
spec:
  host: httpbin
  trafficPolicy:
    connectionPool:
      http:
        http1MaxPendingRequests: 1
        maxRequestsPerConnection: 1
      tcp:
        maxConnections: 1
    outlierDetection:
      baseEjectionTime: 3m
      consecutive5xxErrors: 1
      interval: 1s
      maxEjectionPercent: 100
```

3. Add a client

3.1. Deploy the fortio service

kubectl apply -f samples/httpbin/sample-client/fortio-deploy.yaml



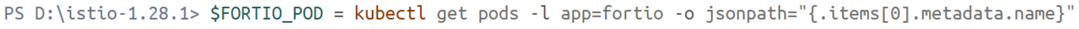
```
PS D:\istio-1.28.1> kubectl apply -f samples/httpbin/sample-client/fortio-deploy.yaml
service/fortio created
deployment.apps/fortio-deploy created
```

3.2. Log in to the client pod and use the fortio tool to call httpbin. Pass in curl to indicate that you just want to make one call

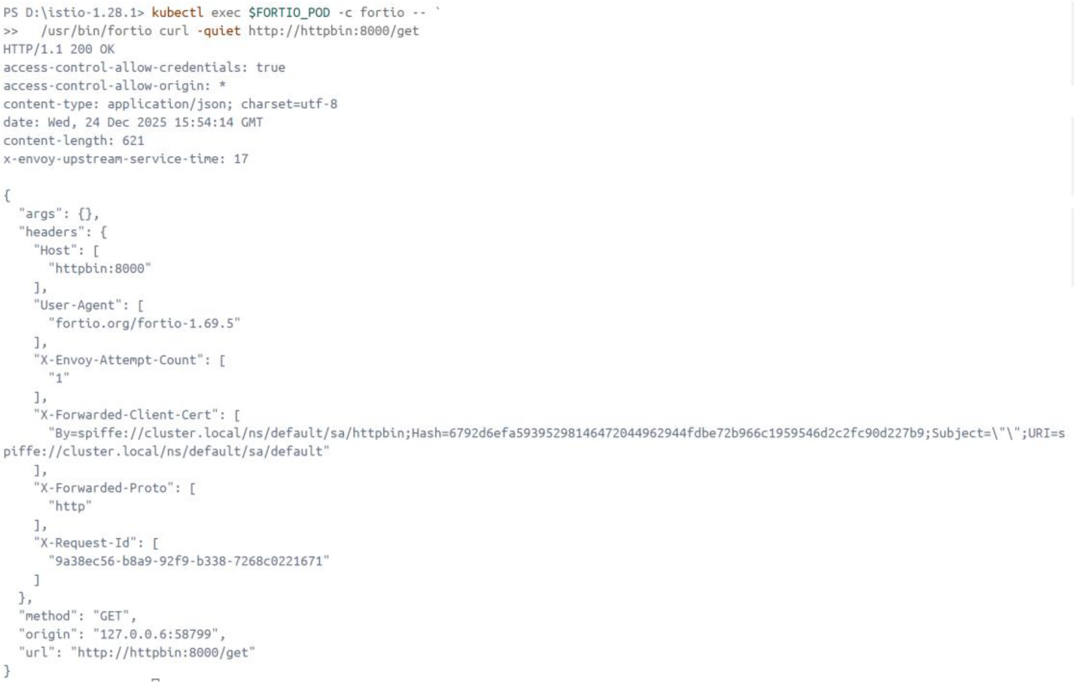
$FORTIO_POD = kubectl get pods -l app=fortio -o jsonpath="{.items[0].metadata.name}"

kubectl exec $FORTIO_POD -c fortio -- `

/usr/bin/fortio curl -quiet http://httpbin:8000/get



```
PS D:\istio-1.28.1> $FORTIO_POD = kubectl get pods -l app=fortio -o jsonpath="{.items[0].metadata.name}"
```



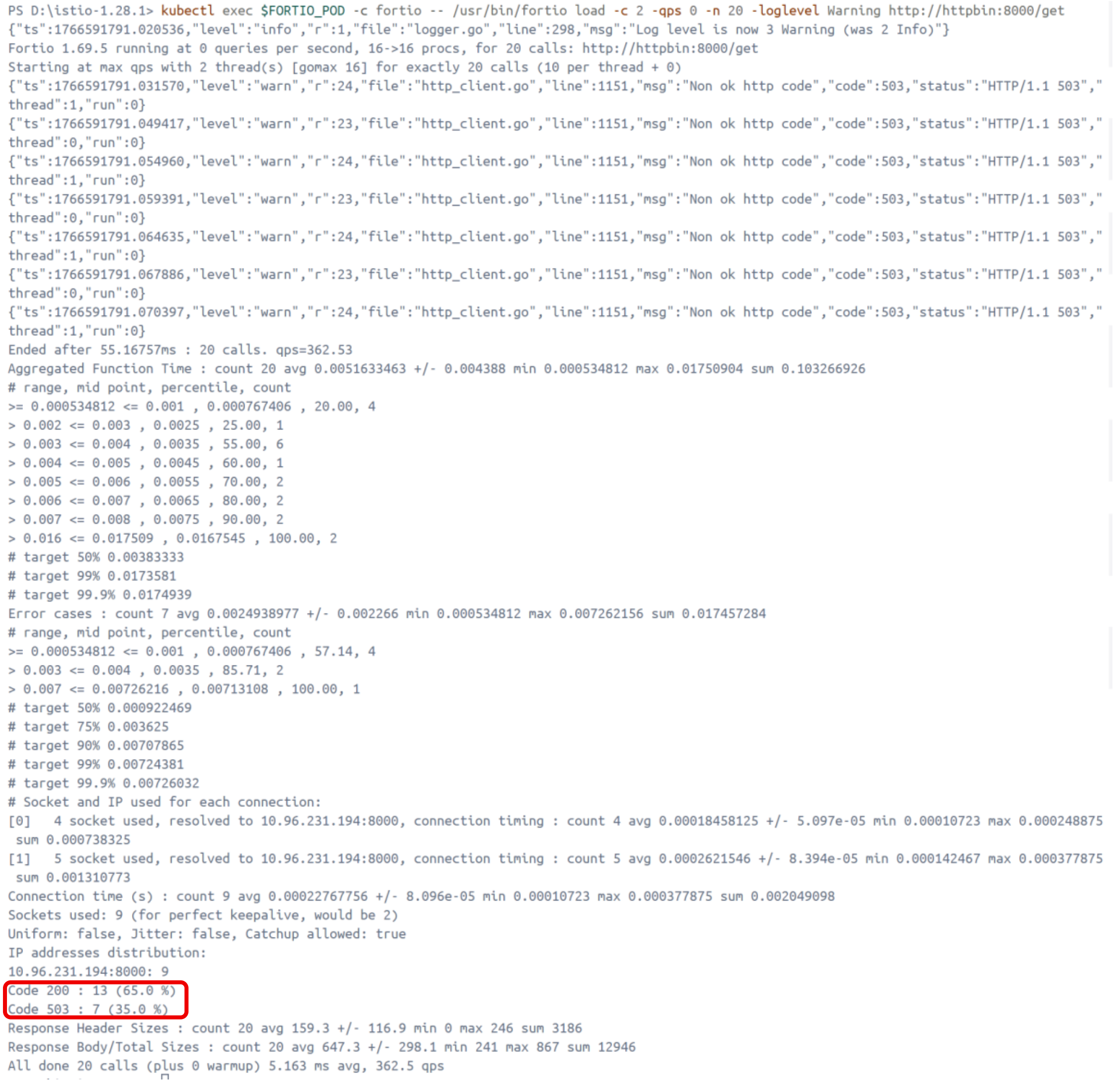
```
PS D:\istio-1.28.1> kubectl exec $FORTIO_POD -c fortio -- `
>>   /usr/bin/fortio curl -quiet http://httpbin:8000/get
HTTP/1.1 200 OK
access-control-allow-credentials: true
access-control-allow-origin: *
content-type: application/json; charset=utf-8
date: Wed, 24 Dec 2025 15:54:14 GMT
content-length: 621
x-envoy-upstream-service-time: 17

{
  "args": {},
  "headers": {
    "Host": [
      "httpbin:8000"
    ],
    "User-Agent": [
      "fortio.org/fortio-1.69.5"
    ],
    "X-Envoy-Attempt-Count": [
      "1"
    ],
    "X-Forwarded-Client-Cert": [
      "By=spiffe://cluster.local/ns/default/sa/httpbin;Hash=6792d6efa593952981464720449629 44fdbe72b966c1959546d2c2fc90d227b9;Subject=\"\";URI=s
piffe://cluster.local/ns/default/sa/default"
    ],
    "X-Forwarded-Proto": [
      "http"
    ],
    "X-Request-Id": [
      "9a38ec56-b8a9-92f9-b338-7268c0221671"
    ]
  },
  "method": "GET",
  "origin": "127.0.0.6:58799",
  "url": "http://httpbin:8000/get"
}
```

4. Tripping the circuit breaker

4.1. Call the service with two concurrent connections (-c 2) and send 20 requests (-n 20)

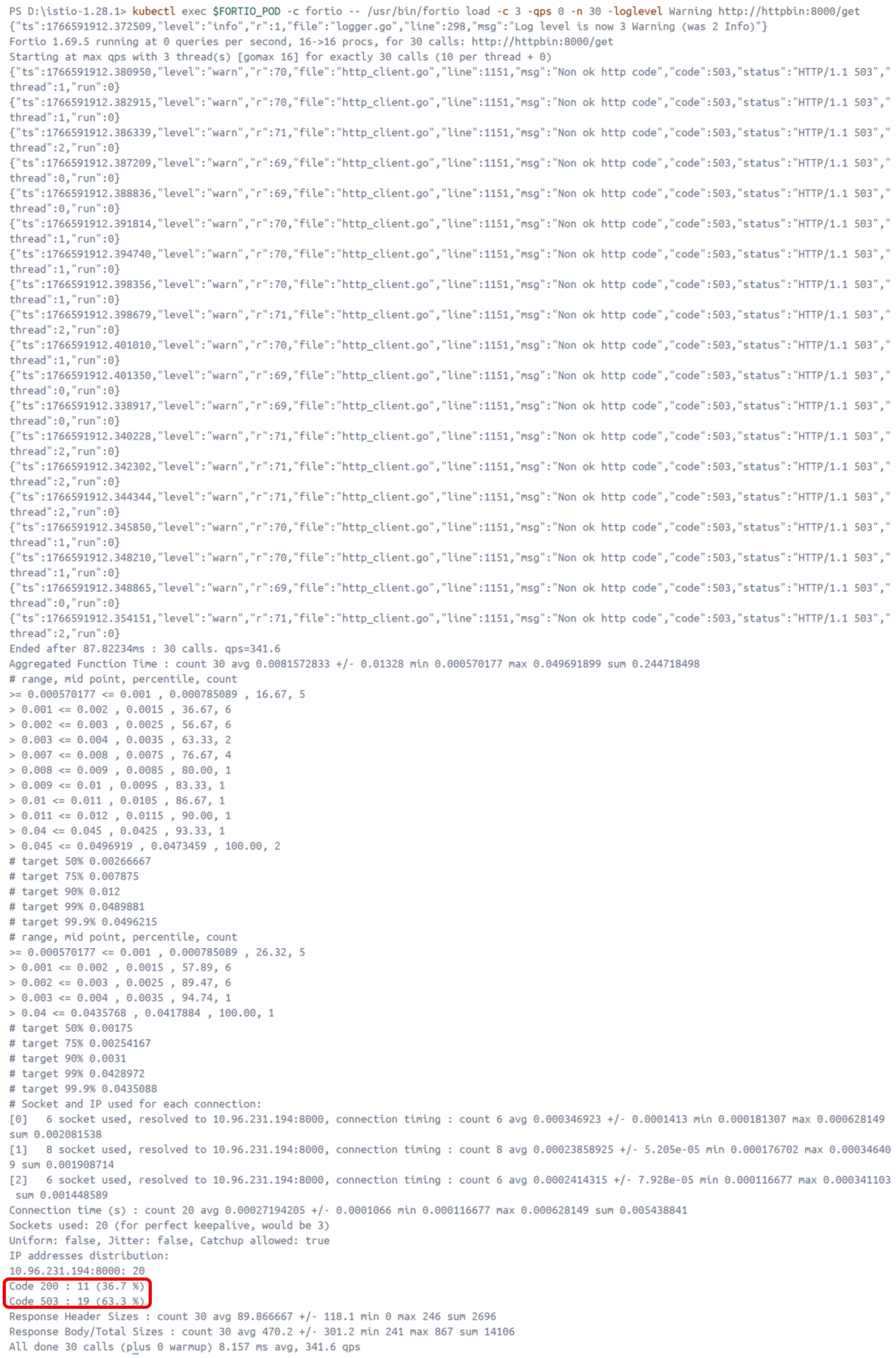
kubectl exec $FORTIO_POD -c fortio -- /usr/bin/fortio load -c 2 -qps 0 -n 20 -loglevel Warning http://httpbin:8000/get



```
PS D:\istio-1.28.1> kubectl exec $FORTIO_POD -c fortio -- /usr/bin/fortio load -c 2 -qps 0 -n 20 -loglevel Warning http://httpbin:8000/get
{"ts":1766591791.020536,"level":"info","r":1,"file":"logger.go","line":298,"msg":"Log level is now 3 Warning (was 2 Info)"}
Fortio 1.69.5 running at 0 queries per second, 16->16 procs, for 20 calls: http://httpbin:8000/get
Starting at max qps with 2 thread(s) [gomax 16] for exactly 20 calls (10 per thread + 0)
{"ts":1766591791.031570,"level":"warn","r":24,"file":"http_client.go","line":1151,"msg":"Non ok http code","code":503,"status":"HTTP/1.1 503","
thread":1,"run":0}
{"ts":1766591791.049417,"level":"warn","r":23,"file":"http_client.go","line":1151,"msg":"Non ok http code","code":503,"status":"HTTP/1.1 503","
thread":0,"run":0}
{"ts":1766591791.054960,"level":"warn","r":24,"file":"http_client.go","line":1151,"msg":"Non ok http code","code":503,"status":"HTTP/1.1 503","
thread":1,"run":0}
{"ts":1766591791.059391,"level":"warn","r":23,"file":"http_client.go","line":1151,"msg":"Non ok http code","code":503,"status":"HTTP/1.1 503","
thread":0,"run":0}
{"ts":1766591791.064635,"level":"warn","r":24,"file":"http_client.go","line":1151,"msg":"Non ok http code","code":503,"status":"HTTP/1.1 503","
thread":1,"run":0}
{"ts":1766591791.067886,"level":"warn","r":23,"file":"http_client.go","line":1151,"msg":"Non ok http code","code":503,"status":"HTTP/1.1 503","
thread":0,"run":0}
{"ts":1766591791.070397,"level":"warn","r":24,"file":"http_client.go","line":1151,"msg":"Non ok http code","code":503,"status":"HTTP/1.1 503","
thread":1,"run":0}
Ended after 55.16757ms : 20 calls. qps=362.53
Aggregated Function Time : count 20 avg 0.0051633463 +/- 0.004388 min 0.000534812 max 0.01750904 sum 0.103266926
# range, mid point, percentile, count
>= 0.000534812 <= 0.001 , 0.000767406 , 20.00, 4
> 0.002 <= 0.003 , 0.0025 , 25.00, 1
> 0.003 <= 0.004 , 0.0035 , 55.00, 6
> 0.004 <= 0.005 , 0.0045 , 60.00, 1
> 0.005 <= 0.006 , 0.0055 , 70.00, 2
> 0.006 <= 0.007 , 0.0065 , 80.00, 2
> 0.007 <= 0.008 , 0.0075 , 90.00, 2
> 0.016 <= 0.017509 , 0.0167545 , 100.00, 2
# target 50% 0.00383333
# target 99% 0.0173581
# target 99.9% 0.0174939
Error cases : count 7 avg 0.0024938977 +/- 0.002266 min 0.000534812 max 0.007262156 sum 0.017457284
# range, mid point, percentile, count
>= 0.000534812 <= 0.001 , 0.000767406 , 57.14, 4
> 0.003 <= 0.004 , 0.0035 , 85.71, 2
> 0.007 <= 0.00726216 , 0.00713108 , 100.00, 1
# target 50% 0.000922469
# target 75% 0.003625
# target 90% 0.00707865
# target 99% 0.00724381
# target 99.9% 0.00726032
# Socket and IP used for each connection:
[0]   4 socket used, resolved to 10.96.231.194:8000, connection timing : count 4 avg 0.00018458125 +/- 5.097e-05 min 0.00010723 max 0.000248875
 sum 0.000738325
[1]   5 socket used, resolved to 10.96.231.194:8000, connection timing : count 5 avg 0.0002621546 +/- 8.394e-05 min 0.000142467 max 0.000377875
 sum 0.001310773
Connection time (s) : count 9 avg 0.00022767756 +/- 8.096e-05 min 0.00010723 max 0.000377875 sum 0.002049098
Sockets used: 9 (for perfect keepalive, would be 2)
Uniform: false, Jitter: false, Catchup allowed: true
IP addresses distribution:
10.96.231.194:8000: 9
Code 200 : 13 (65.0 %)
Code 503 : 7 (35.0 %)
Response Header Sizes : count 20 avg 159.3 +/- 116.9 min 0 max 246 sum 3186
Response Body/Total Sizes : count 20 avg 647.3 +/- 298.1 min 241 max 867 sum 12946
All done 20 calls (plus 0 warmup) 5.163 ms avg, 362.5 qps
```

4.2. Bring the number of concurrent connections up to 3

kubectl exec $FORTIO_POD -c fortio -- /usr/bin/fortio load -c 3 -qps 0 -n 30 -loglevel Warning http://httpbin:8000/get



```
PS D:\istio-1.28.1> kubectl exec $FORTIO_POD -c fortio -- /usr/bin/fortio load -c 3 -qps 0 -n 30 -loglevel Warning http://httpbin:8000/get
{"ts":1766591912.372509,"level":"info","r":1,"file":"logger.go","line":298,"msg":"Log level is now 3 Warning (was 2 Info)"}
Fortio 1.69.5 running at 0 queries per second, 16->16 procs, for 30 calls: http://httpbin:8000/get
Starting at max qps with 3 thread(s) [gomax 16] for exactly 30 calls (10 per thread + 0)
{"ts":1766591912.380950,"level":"warn","r":70,"file":"http_client.go","line":1151,"msg":"Non ok http code","code":503,"status":"HTTP/1.1 503","
thread":1,"run":0}
{"ts":1766591912.382915,"level":"warn","r":70,"file":"http_client.go","line":1151,"msg":"Non ok http code","code":503,"status":"HTTP/1.1 503","
thread":1,"run":0}
{"ts":1766591912.386339,"level":"warn","r":71,"file":"http_client.go","line":1151,"msg":"Non ok http code","code":503,"status":"HTTP/1.1 503","
thread":2,"run":0}
{"ts":1766591912.387209,"level":"warn","r":69,"file":"http_client.go","line":1151,"msg":"Non ok http code","code":503,"status":"HTTP/1.1 503","
thread":0,"run":0}
{"ts":1766591912.388836,"level":"warn","r":69,"file":"http_client.go","line":1151,"msg":"Non ok http code","code":503,"status":"HTTP/1.1 503","
thread":0,"run":0}
{"ts":1766591912.391814,"level":"warn","r":70,"file":"http_client.go","line":1151,"msg":"Non ok http code","code":503,"status":"HTTP/1.1 503","
thread":1,"run":0}
{"ts":1766591912.394740,"level":"warn","r":70,"file":"http_client.go","line":1151,"msg":"Non ok http code","code":503,"status":"HTTP/1.1 503","
thread":1,"run":0}
{"ts":1766591912.398356,"level":"warn","r":70,"file":"http_client.go","line":1151,"msg":"Non ok http code","code":503,"status":"HTTP/1.1 503","
thread":1,"run":0}
{"ts":1766591912.398679,"level":"warn","r":71,"file":"http_client.go","line":1151,"msg":"Non ok http code","code":503,"status":"HTTP/1.1 503","
thread":2,"run":0}
{"ts":1766591912.401010,"level":"warn","r":70,"file":"http_client.go","line":1151,"msg":"Non ok http code","code":503,"status":"HTTP/1.1 503","
thread":1,"run":0}
{"ts":1766591912.401350,"level":"warn","r":69,"file":"http_client.go","line":1151,"msg":"Non ok http code","code":503,"status":"HTTP/1.1 503","
thread":0,"run":0}
{"ts":1766591912.338917,"level":"warn","r":69,"file":"http_client.go","line":1151,"msg":"Non ok http code","code":503,"status":"HTTP/1.1 503","
thread":0,"run":0}
{"ts":1766591912.340228,"level":"warn","r":71,"file":"http_client.go","line":1151,"msg":"Non ok http code","code":503,"status":"HTTP/1.1 503","
thread":2,"run":0}
{"ts":1766591912.342302,"level":"warn","r":71,"file":"http_client.go","line":1151,"msg":"Non ok http code","code":503,"status":"HTTP/1.1 503","
thread":2,"run":0}
{"ts":1766591912.344344,"level":"warn","r":71,"file":"http_client.go","line":1151,"msg":"Non ok http code","code":503,"status":"HTTP/1.1 503","
thread":2,"run":0}
{"ts":1766591912.345850,"level":"warn","r":70,"file":"http_client.go","line":1151,"msg":"Non ok http code","code":503,"status":"HTTP/1.1 503","
thread":1,"run":0}
{"ts":1766591912.348210,"level":"warn","r":70,"file":"http_client.go","line":1151,"msg":"Non ok http code","code":503,"status":"HTTP/1.1 503","
thread":1,"run":0}
{"ts":1766591912.348865,"level":"warn","r":69,"file":"http_client.go","line":1151,"msg":"Non ok http code","code":503,"status":"HTTP/1.1 503","
thread":0,"run":0}
{"ts":1766591912.354151,"level":"warn","r":71,"file":"http_client.go","line":1151,"msg":"Non ok http code","code":503,"status":"HTTP/1.1 503","
thread":2,"run":0}
Ended after 87.82234ms : 30 calls. qps=341.6
Aggregated Function Time : count 30 avg 0.0081572833 +/- 0.01328 min 0.000570177 max 0.049691899 sum 0.244718498
# range, mid point, percentile, count
>= 0.000570177 <= 0.001 , 0.000785089 , 16.67, 5
> 0.001 <= 0.002 , 0.0015 , 36.67, 6
> 0.002 <= 0.003 , 0.0025 , 56.67, 6
> 0.003 <= 0.004 , 0.0035 , 63.33, 2
> 0.007 <= 0.008 , 0.0075 , 76.67, 4
> 0.008 <= 0.009 , 0.0085 , 80.00, 1
> 0.009 <= 0.01 , 0.0095 , 83.33, 1
> 0.01 <= 0.011 , 0.0105 , 86.67, 1
> 0.011 <= 0.012 , 0.0115 , 90.00, 1
> 0.04 <= 0.045 , 0.0425 , 93.33, 1
> 0.045 <= 0.0496919 , 0.0473459 , 100.00, 2
# target 50% 0.00266667
# target 75% 0.007875
# target 90% 0.012
# target 99% 0.0489881
# target 99.9% 0.0496215
# range, mid point, percentile, count
>= 0.000570177 <= 0.001 , 0.000785089 , 26.32, 5
> 0.001 <= 0.002 , 0.0015 , 57.89, 6
> 0.002 <= 0.003 , 0.0025 , 89.47, 6
> 0.003 <= 0.004 , 0.0035 , 94.74, 1
> 0.04 <= 0.0435768 , 0.0417884 , 100.00, 1
# target 50% 0.00175
# target 75% 0.00254167
# target 90% 0.0031
# target 99% 0.0428972
# target 99.9% 0.0435088
# Socket and IP used for each connection:
[0]   6 socket used, resolved to 10.96.231.194:8000, connection timing : count 6 avg 0.000346923 +/- 0.0001413 min 0.000181307 max 0.000628149
sum 0.002081538
[1]   8 socket used, resolved to 10.96.231.194:8000, connection timing : count 8 avg 0.00023858925 +/- 5.205e-05 min 0.000176702 max 0.00034640
9 sum 0.001908714
[2]   6 socket used, resolved to 10.96.231.194:8000, connection timing : count 6 avg 0.0002414315 +/- 7.928e-05 min 0.000116677 max 0.000341103
 sum 0.001448589
Connection time (s) : count 20 avg 0.00027194205 +/- 0.0001066 min 0.000116677 max 0.000628149 sum 0.005438841
Sockets used: 20 (for perfect keepalive, would be 3)
Uniform: false, Jitter: false, Catchup allowed: true
IP addresses distribution:
10.96.231.194:8000: 20
Code 200 : 11 (36.7 %)
Code 503 : 19 (63.3 %)
Response Header Sizes : count 30 avg 89.866667 +/- 118.1 min 0 max 246 sum 2696
Response Body/Total Sizes : count 30 avg 470.2 +/- 301.2 min 241 max 867 sum 14106
All done 30 calls (plus 0 warmup) 8.157 ms avg, 341.6 qps
```

Only 36.7% of the requests succeeded and the rest were trapped by circuit breaking:

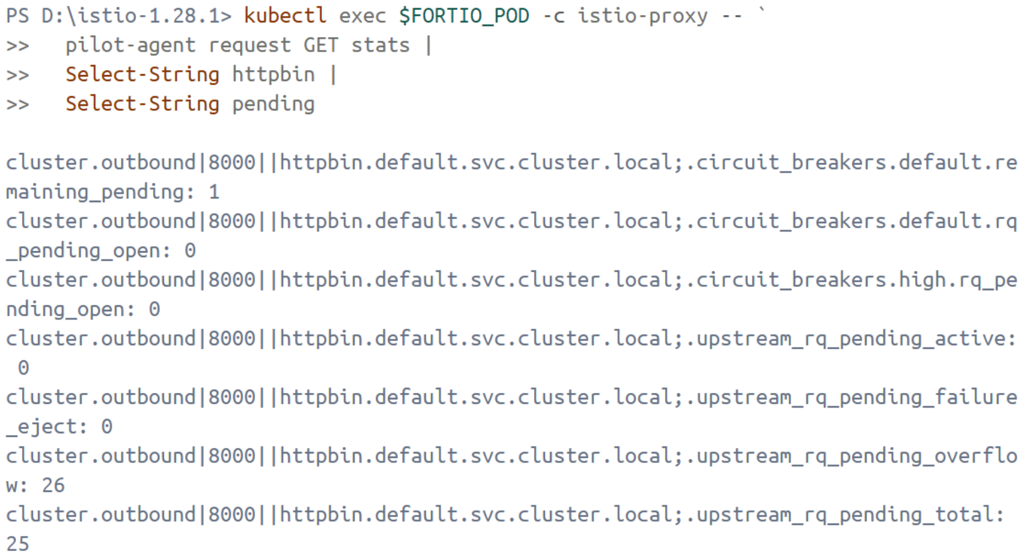
4.3. Query the istio-proxy stats to see more

```
kubectl exec $FORTIO_POD -c istio-proxy -- `
```

pilot-agent request GET stats |

Select-String httpbin |

Select-String pending



```
PS D:\istio-1.28.1> kubectl exec $FORTIO_POD -c istio-proxy -- `
>>   pilot-agent request GET stats |
>>   Select-String httpbin |
>>   Select-String pending

cluster.outbound|8000||httpbin.default.svc.cluster.local;.circuit_breakers.default.re
maining_pending: 1
cluster.outbound|8000||httpbin.default.svc.cluster.local;.circuit_breakers.default.rq
_pending_open: 0
cluster.outbound|8000||httpbin.default.svc.cluster.local;.circuit_breakers.high.rq_pe
nding_open: 0
cluster.outbound|8000||httpbin.default.svc.cluster.local;.upstream_rq_pending_active:
 0
cluster.outbound|8000||httpbin.default.svc.cluster.local;.upstream_rq_pending_failure
_eject: 0
cluster.outbound|8000||httpbin.default.svc.cluster.local;.upstream_rq_pending_overflo
w: 26
cluster.outbound|8000||httpbin.default.svc.cluster.local;.upstream_rq_pending_total:
25
```

It shows 26 for the upstream_rq_pending_overflow value which means 26 calls so far have been flagged for circuit breaking.

5. Clean up

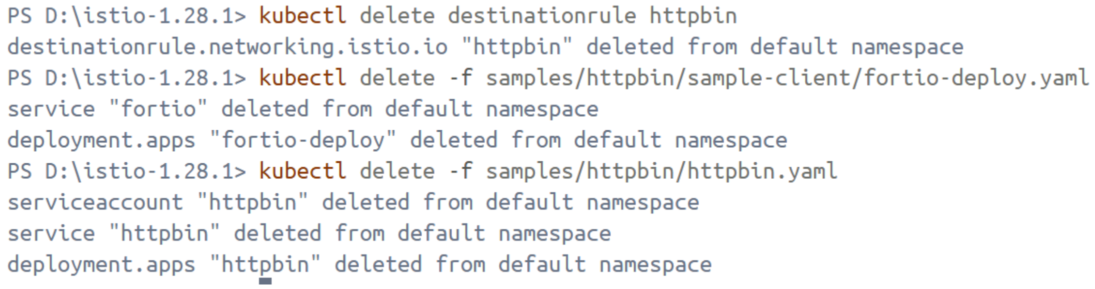
5.1. Remove the rules:

kubectl delete destinationrule httpbin

5.2. Shutdown the httpbin service and client:

kubectl delete -f samples/httpbin/sample-client/fortio-deploy.yaml

kubectl delete -f samples/httpbin/httpbin.yaml



```
PS D:\istio-1.28.1> kubectl delete destinationrule httpbin
destinationrule.networking.istio.io "httpbin" deleted from default namespace
PS D:\istio-1.28.1> kubectl delete -f samples/httpbin/sample-client/fortio-deploy.yaml
service "fortio" deleted from default namespace
deployment.apps "fortio-deploy" deleted from default namespace
PS D:\istio-1.28.1> kubectl delete -f samples/httpbin/httpbin.yaml
serviceaccount "httpbin" deleted from default namespace
service "httpbin" deleted from default namespace
deployment.apps "httpbin" deleted from default namespace
```