

# How Istio DestinationRule related to Kubernetes Service?

Asked 2 years, 2 months ago   Modified 2 years, 2 months ago   Viewed 4k times



I trying to understand how to work load balancing in Istio.

5

Istio DestinationRule define rules for traffic balancing between pods. K8s Service similar manages traffic load balancing between pods.



DestinationRule define host and k8s service define host.



But without k8s service, request failed with http code 503.

How k8s Service relates to DestinationRule?

[load-balancing](#)   [istio](#)

Share   Improve this question   Follow

asked Oct 22, 2020 at 8:41



[Pavel Serebryakov](#)

53   1   3

- 1   The k8s service still has *some* utility with Istio, but it's like Istio is reinterpreting it in its own way. I don't know exactly what happens under the cover, but the service name and ports are still used by istio to identify to some extent the destination(s) of the requests. Then, DestinationRules / VirtualServices will extend / refine the routing ; and load balancing takes place in the Envoy sidecars. – [Joel](#) Oct 22, 2020 at 14:29

Sorted by:

1 Answer

Highest score (default)  



## Kubernetes service

5

Kubernetes service type `ClusterIP` uses **kube-proxy's iptables rules** to distribute the requests.



The documentation says:



By default, kube-proxy in userspace mode chooses a backend via a round-robin algorithm.



More about it [here](#).

## Destination rule

As mentioned [here](#)

You can think of virtual services as how you route your traffic to a given destination, and then you use destination rules to configure what happens to traffic for that destination. Destination rules are applied after virtual service routing rules are evaluated, so they apply to the traffic's "real" destination.

Every HTTP route must have a target: a route, or a redirect. A route is a forwarding target, and it can point to one of several versions of a service described in DestinationRules. Weights associated with the service version determine the proportion of traffic it receives.

DestinationRule defines policies that apply to traffic intended for a service after routing has occurred.

And [here](#)

While a virtual service matches on a rule and evaluates a destination to route the traffic to, destination rules define available subsets of the service to send the traffic.

For example, if you have a service that has multiple versions running at a time, you can create destination rules to define routes to those versions. Then use virtual services to map to a specific subset defined by the destination rules or split a percentage of the traffic to particular versions.

## 503 without kubernetes service

But without k8s service, request failed with http code 503.

It's failing because there is no host, which was specified in virtual service, and destination rule.

For example, take a look at this [virtual service and destination rule](#).

```
apiVersion: networking.istio.io/v1alpha3
kind: VirtualService
metadata:
  name: reviews-route
spec:
  hosts:
  - reviews.prod.svc.cluster.local
  http:
  - name: "reviews-v2-routes"
    match:
    - uri:
        prefix: "/wpcatalog"
    route:
    - destination:
        host: reviews.prod.svc.cluster.local <---
        subset: v2
```

```

- name: "reviews-v1-route"
  route:
  - destination:
      host: reviews.prod.svc.cluster.local <---
      subset: v1

apiVersion: networking.istio.io/v1alpha3
kind: DestinationRule
metadata:
  name: reviews-destination
spec:
  host: reviews.prod.svc.cluster.local <---
  subsets:
  - name: v1
    labels:
      version: v1
  - name: v2
    labels:
      version: v2

```

If you check the `host` then you see there is your kubernetes service specified, and it won't work without it.

Additionally when setting route rules to direct traffic to specific versions (subsets) of a service, care must be taken to ensure that the subsets are available before they are used in the routes. Otherwise, calls to the service may return 503 errors during a reconfiguration period.

More about it [here](#).

## DestinationRule define host and k8s service define host.

Destination rule host is your kubernetes service. Kubernetes service hosts are your pods, you may be wondering, but why do I need a service?

As mentioned [here](#).

A Kubernetes Service is an abstraction which defines a logical set of Pods running somewhere in your cluster, that all provide the same functionality. When created, each Service is assigned a unique IP address (also called clusterIP). This address is tied to the lifespan of the Service, and will not change while the Service is alive. Pods can be configured to talk to the Service, and know that communication to the Service will be automatically load-balanced out to some pod that is a member of the Service.

## Kubernetes Service relates to DestinationRule

I couldn't find the exact information how it works, so I will explain how I understand it.

You need kubernetes service so virtual service and destination rule could actually work.

As kubernetes service uses kube-proxy's iptables rules to distribute the requests, I assume that istio destination rule can override it with his own rules, and apply them through envoy

sidecar, because all traffic that your mesh services send and receive (data plane traffic) is proxied through Envoy, making it easy to direct and control traffic around your mesh without making any changes to your services.

More about it [here](#).

Additional resources:

- <https://istio.io/latest/docs/reference/config/networking/destination-rule/#Subset>
- <https://istio.io/latest/docs/examples/bookinfo/#apply-default-destination-rules>
- <https://istio.io/latest/docs/concepts/traffic-management/#load-balancing-options>

Let me know if you have any more questions.

Share Improve this answer Follow

answered Oct 23, 2020 at 6:16



Jakub

7,779

1

15

30

---

Hi Jakub! I attentively studied your answer, it very useful. I dug Istio a little deeper and how I understand, when istio define subset in destinationrule, it use endpoint information from kubernetes services. Istio envoy load balancing perform endpoints from k8s service. I have a question: what is virtualservice host name ? VirtualService host is not part kubernetes DNS. I route traffic to k8s service, and envoy define existing VirtualService by host name ? – [Pavel Serebryakov](#) Oct 29, 2020 at 21:00

---

Hi @PavelSerebryakov, I'm not sure if you're asking about host or destination field, but they're both well described [here](#) and [here](#). Virtual Service uses istio registry for that, For example, if you've installed Istio on a Kubernetes cluster, then Istio automatically detects the services and endpoints in that cluster. More about it [here](#). – [Jakub](#) Oct 30, 2020 at 6:38

---

If this answer or any other one was helpful or solved your issue, please mark it as accepted or upvote it as per [stackoverflow rules](#). – [Jakub](#) Oct 30, 2020 at 6:40

---