

## Istio routing mechanism

This solution can be used to make canary deployments and also make user experience more resilient - “user resilience”. For example, new version of service can be made available only to one group of users (test group). It can be as much as only 1% of of the hole traffic. Users can be filtered by headers in http request. If something goes wrong with new version of service it is very easy to rollback and switch all the traffic back to production version.

```
$ make deploy-app-default
```

```
$ make deploy-istio-default
```

```
$ make health
```

```
curl http://192.168.99.113:31221/status
CPanel v1 : Online
curl http://192.168.99.113:31221/cameras/1/state
{"streaming":false,"cycle":88,"fps":0,"section":"1","destination":"http://collector.default.svc.cluster.local:8080","event":"exit"}
curl http://192.168.99.113:31221/cameras/2/state
{"streaming":false,"cycle":92,"fps":0,"section":"1","destination":"http://collector.default.svc.cluster.local:8080","event":"entry"}
curl http://192.168.99.113:31221/collector/status
Collector v1 : Online
curl http://192.168.99.113:31221/alerts/status
Alerts v1 : Online
curl http://192.168.99.113:31221/sections/1/status
Section 1 v1 : Online
```

```
$ make start-cameras
```

```
$ make health
```

```
curl http://192.168.99.113:31221/status
CPanel v1 : Online
curl http://192.168.99.113:31221/cameras/1/state
{"streaming":true,"cycle":7,"fps":0,"section":"1","destination":"http://collector.default.svc.cluster.local:8080","event":"exit"}
curl http://192.168.99.113:31221/cameras/2/state
{"streaming":true,"cycle":5,"fps":0,"section":"1","destination":"http://collector.default.svc.cluster.local:8080","event":"entry"}
curl http://192.168.99.113:31221/collector/status
Collector v1 : Online
curl http://192.168.99.113:31221/alerts/status
Alerts v1 : Online
curl http://192.168.99.113:31221/sections/1/status
Section 1 v1 : Online
```

```
curl http://192.168.99.113:31221/status
```

```
CPanel v1 : Online
```

```
curl http://192.168.99.113:31221/cameras/1/state
```

```
{"streaming":true,"cycle":7,"fps":0,"section":"1","destination":"http://collector.default.svc.cluster.local:8080","event":"exit"}
```

```
curl http://192.168.99.113:31221/cameras/2/state
```

```
{"streaming":true,"cycle":5,"fps":0,"section":"1","destination":"http://collector.default.svc.cluster.local:8080","event":"entry"}
```

```
curl http://192.168.99.113:31221/collector/status
```

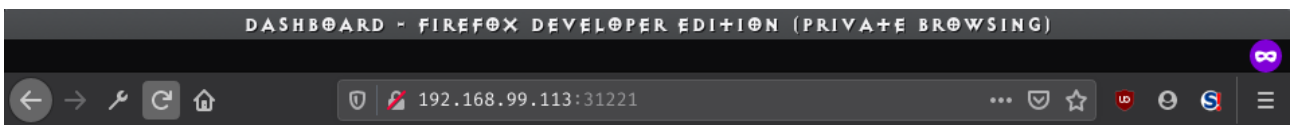
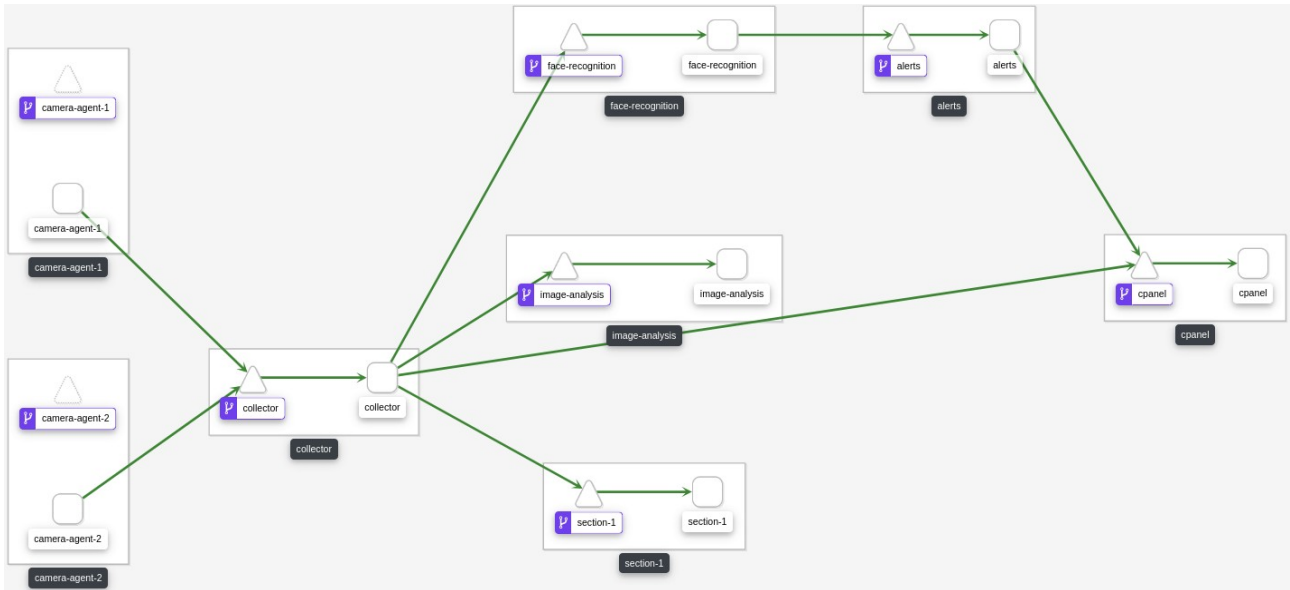
```
Collector v1 : Online
```

```
curl http://192.168.99.113:31221/alerts/status
```

```
Alerts v1 : Online
```

```
curl http://192.168.99.113:31221/sections/1/status
```

```
Section 1 v1 : Online
```



# Dashboard V1

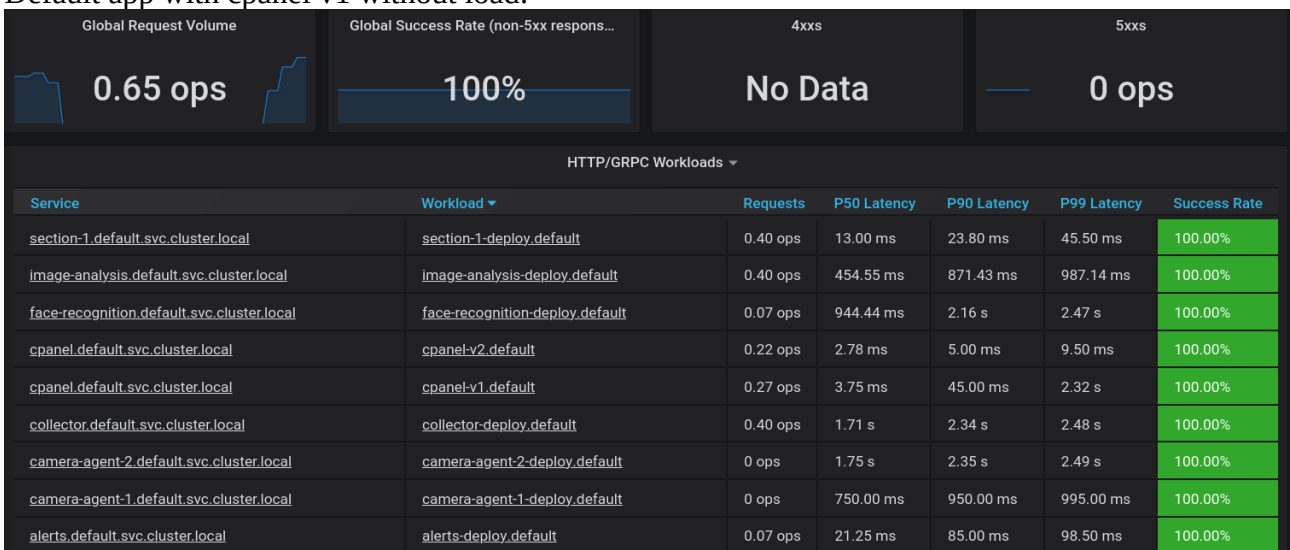
## Section 1

timestamp: 2020-02-25T14:35:38.204522Z  
 gender: male | age: 38-43 | event: exit

## Alert

timestamp: 2020-02-25T14:35:27.224857Z  
 section: 1  
 event: entry  
 name: **PersonX**

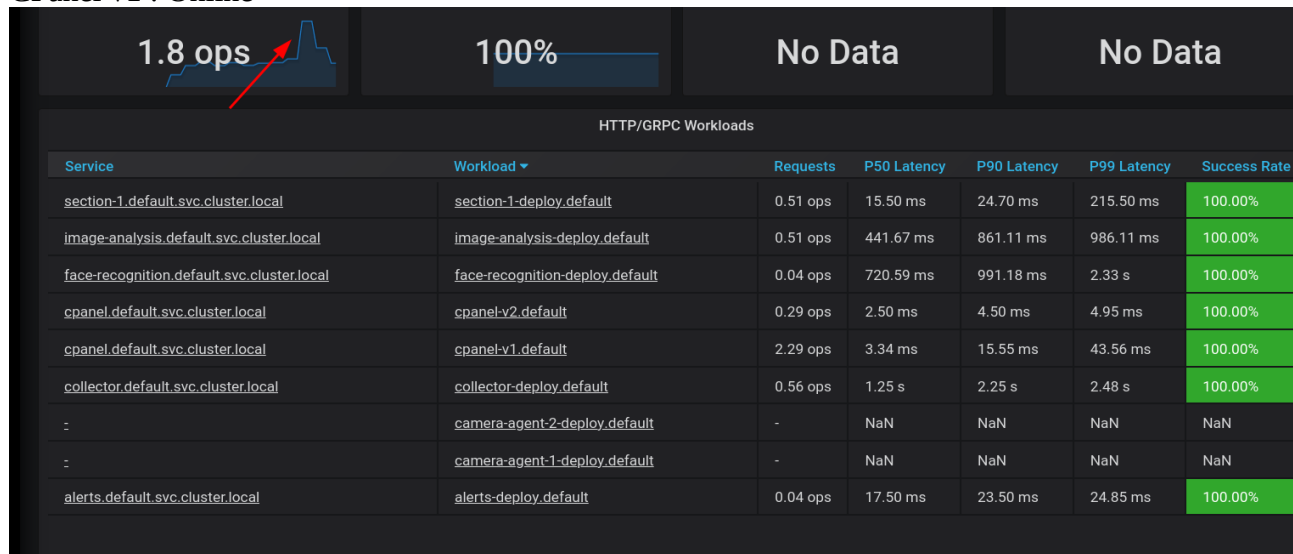
Default app with cpanel v1 without load:



\$ make load  
 for i in {1..100}; do sleep 0.2; curl http://192.168.99.113:31221/status; printf "\n"; done  
 CPanel v1 : Online

CPanel v1 : Online

CPanel v1 : Online



```
$ make cpanel-50-50
```

```
./kubectl apply -f istio/virt_svc_50-50.yaml
```

virtualservice.networking.istio.io/cpanel configured

check configuration

```
$ k get virtualservices cpanel -o yaml
```

route:

- destination:

host: cpanel.default.svc.cluster.local

port:

number: 8080

subset: v1

weight: 50

- destination:

host: cpanel.default.svc.cluster.local

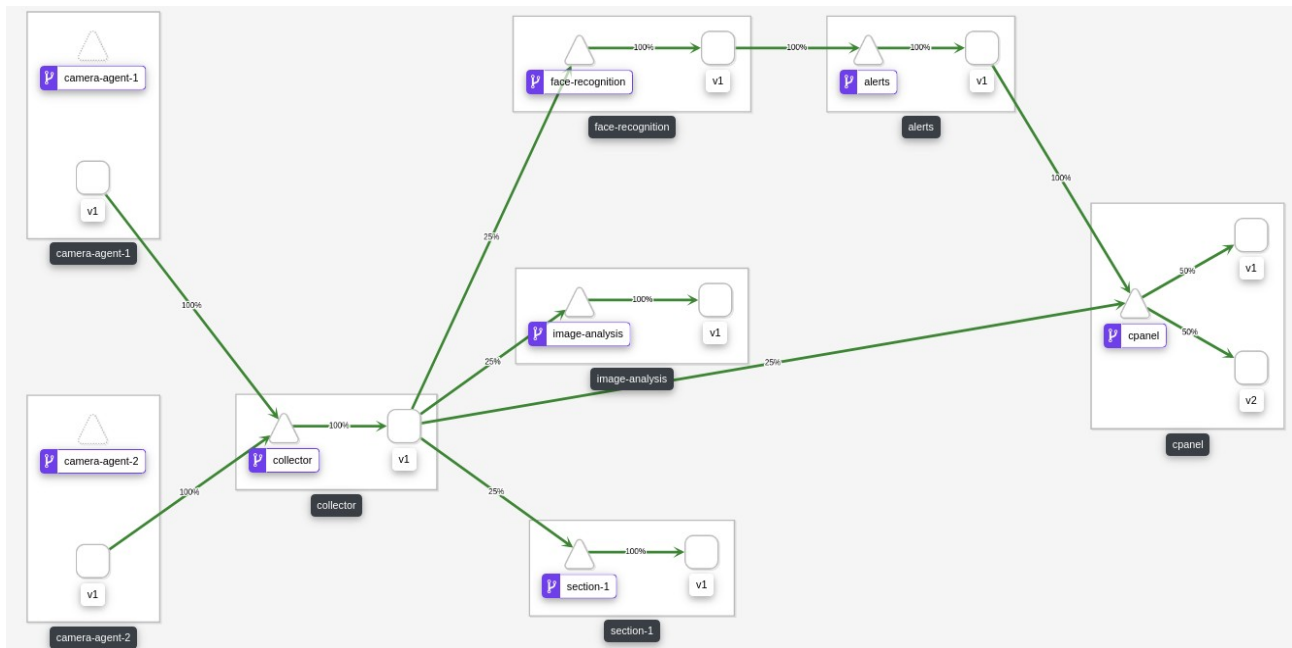
port:

number: 8080

subset: v2

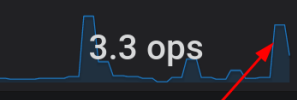
weight: 50

```
$ make start-cameras
```



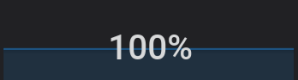
```
$ make load
for i in {1..100}; do sleep 0.2; curl http://192.168.99.113:31221/status; printf "\n"; done
CPanel v1 : Online
CPanel v1 : Online
CPanel v1 : Online
CPanel v2 : Online
CPanel v2 : Online
CPanel v2 : Online
```

Global Request Volume



3.3 ops

Global Success Rate (non-5xx responses...)



100%

4xxs

No Data

5xxs

No Data

HTTP/GRPC Workloads ▾

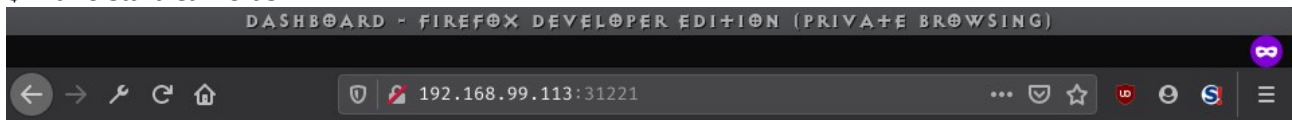
Service	Workload ▾	Requests	P50 Latency	P90 Latency	P99 Latency	Success Rate
<a href="#">section-1.default.svc.cluster.local</a>	<a href="#">section-1-deploy.default</a>	0.44 ops	83.33 ms	220.00 ms	450.00 ms	100.00%
<a href="#">image-analysis.default.svc.cluster.local</a>	<a href="#">image-analysis-deploy.default</a>	0.47 ops	461.54 ms	1.60 s	4.45 s	100.00%
<a href="#">face-recognition.default.svc.cluster.local</a>	<a href="#">face-recognition-deploy.default</a>	0.04 ops	857.14 ms	2.50 s	9.00 s	100.00%
<a href="#">cpanel.default.svc.cluster.local</a>	<a href="#">cpanel-v2.default</a>	1.40 ops	2.92 ms	6.93 ms	20.28 ms	100.00%
<a href="#">cpanel.default.svc.cluster.local</a>	<a href="#">cpanel-v1.default</a>	1.33 ops	3.19 ms	9.38 ms	42.50 ms	100.00%
<a href="#">collector.default.svc.cluster.local</a>	<a href="#">collector-deploy.default</a>	0.47 ops	1.59 s	2.49 s	8.95 s	100.00%
<a href="#">alerts.default.svc.cluster.local</a>	<a href="#">alerts-deploy.default</a>	0.04 ops	17.50 ms	23.50 ms	24.85 ms	100.00%

```
$ make cpanel-v2
./kubectl apply -f istio/virt_svc_v2.yaml
virtualservice.networking.istio.io/cpanel configured
check configuration
$ k get virtualservices cpanel -o yaml
```

```
route:
- destination:
    host: cpanel.default.svc.cluster.local
    port:
```

number: 8080  
subset: v1  
weight: 0  
- destination:  
host: cpanel.default.svc.cluster.local  
port:  
number: 8080  
subset: v2  
weight: 100

\$ make start-cameras



## Dashboard V2

### Section 1



timestamp: 2020-02-25T15:27:21.900453Z

**gender: male** | age: 25-32 | event: entry

**gender: male** | age: 25-32 | event: entry

### Alert



timestamp: 2020-02-25T15:26:55.022111Z

section: 1

event: exit

name: **George W**

\$ make load

for i in {1..100}; do sleep 0.2; curl http://192.168.99.113:31221/status; printf "\n"; done

CPanel v2 : Online

CPanel v2 : Online

CPanel v2 : Online

CPanel v2 : Online

