
KubeCon
CloudNativeCon
OPEN SOURCE SUMMIT
China 2019

Istio Performance in Large Scale Cluster And Best Practices

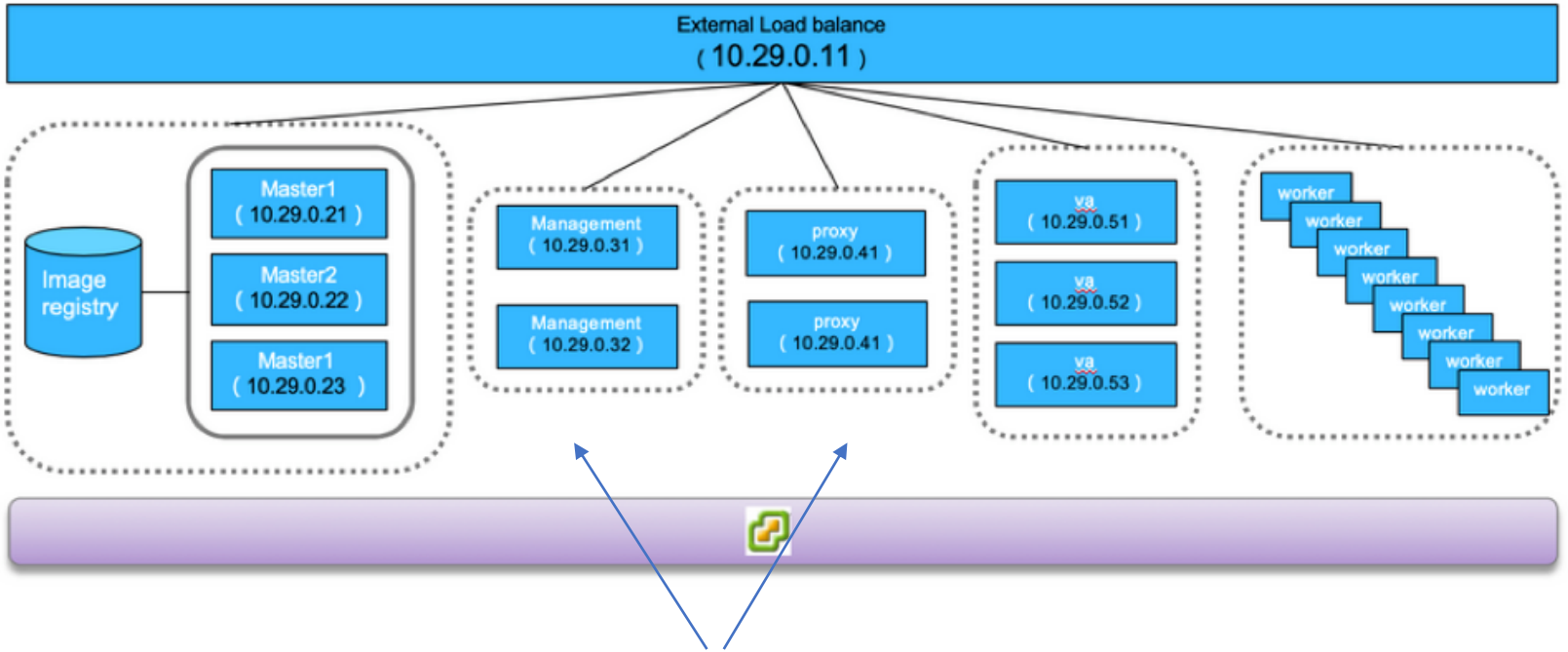
Chun Lin Yang (clyang@cn.ibm.com)
Senior Software Arch, IBM
IBM China Systems Lab
Twitter: @clyang11

Guang Ya Liu (liugya@cn.ibm.com)
STSM, IBM Multicloud Platform
IBM China Systems Lab
Twitter: @gyliu513

Agenda

- ❑ What Problem We Have
- ❑ What We Have Done
- ❑ Best Practices
- ❑ More Tuning Guidance

What Problem We Have



The istio control panel is running in management and proxy nodes.

What Problem We Have

Test 10000+ pods with 4000 services, including 4000 pods and 1000 services in 100 namespaces are not managed by istio, while 6000 pods and 3000 services in 100 namespaces are managed by istio.

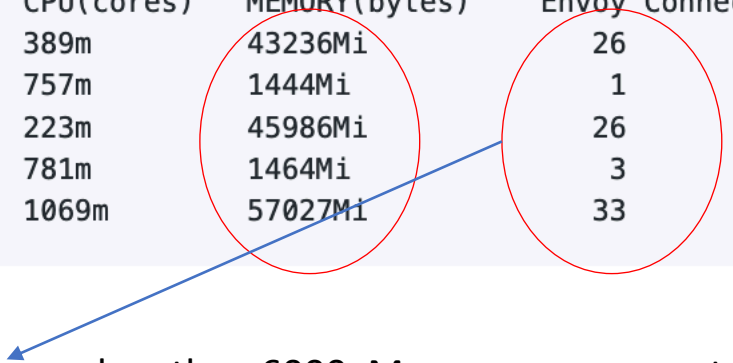
Istio components information when the cluster has 4000 pods and 1000 services and they are not managed by Istio

NAME	CPU(cores)	MEMORY(bytes)
istio-citadel-7d6ffd5d7f-kt2nq	1m	17Mi
istio-galley-7d5687fcc5-krvpf	57m	35Mi
istio-ingressgateway-78f6846c48-92hss	205m	498Mi
istio-pilot-ddc499798-t7hrh	172m	1232Mi
istio-policy-78588997b4-4wmk6	7m	179Mi
istio-sidecar-injector-58ff476d66-jrk5q	16m	8Mi
istio-telemetry-7556866cc8-2l9fr	7m	180Mi
prometheus-8469d98948-bpbcs	863m	3257Mi

What Problem We Have

Istio pilot component information after created 10000 pods and 4000 services

NAME	CPU(cores)	MEMORY(bytes)	Envoy Connections
istio-pilot-ddc499798-6cswb	389m	43236Mi	26
istio-pilot-ddc499798-9d4qz	757m	1444Mi	1
istio-pilot-ddc499798-bbmxv	223m	45986Mi	26
istio-pilot-ddc499798-qrqfw	781m	1464Mi	3
istio-pilot-ddc499798-t7hrh	1069m	57027Mi	33



We can see that the total envoy connections are less than 6000. Many envoys cannot connect to pilot

That means the memory of pilot will be increased in index increase following the increasing of pod/service/virtualservice.

What We Have Done

In istio 1.1, there is a new feature namespace isolation

`Sidecar` describes the configuration of the sidecar proxy that mediates inbound and outbound communication to the workload it is attached to. By default, Istio will program all sidecar proxies in the mesh with the necessary configuration required to reach every workload in the mesh, as well as accept traffic on all the ports associated with the workload. The Sidecar resource provides a way to fine tune the set of ports, protocols that the proxy will accept when forwarding traffic to and from the workload. In addition, it is possible to restrict the set of services that the proxy can reach when forwarding outbound traffic from the workload.

What We Have Done

Deploy the default global sidecar to enable the namespace isolation. In istio managed namespace, we also created the ingressgateway for each namespace, all the traffic for this namespace will use its ingressgateway to avoid the bottleneck of global ingressgateway.

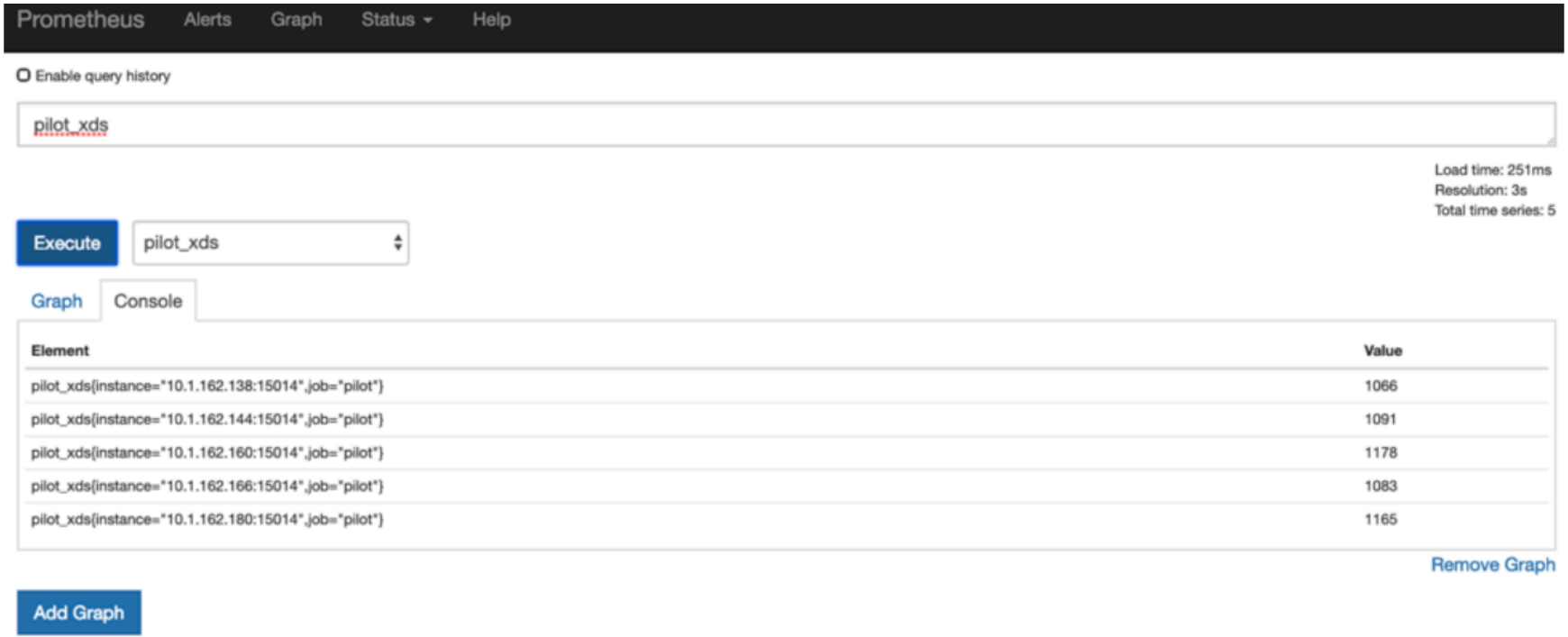
```
root@icp10b1:~# kubectl top pod -n istio-system
NAME                                      CPU(cores)   MEMORY(bytes)
istio-citadel-7d6ffd5d7f-7nlvd            54m          50Mi
istio-galley-7d5687fcc5-z775d             67m          38Mi
istio-ingressgateway-78f6846c48-9fpzm     15m          34Mi
istio-pilot-c56988865-5t4sd               3218m        1691Mi
istio-pilot-c56988865-9g9ng               2203m        1681Mi
istio-pilot-c56988865-cfsgm               3129m        1458Mi
istio-pilot-c56988865-pb4q9               3253m        1693Mi
istio-pilot-c56988865-q2dfp               3300m        1419Mi
istio-policy-78588997b4-x962b             45m          348Mi
istio-sidecar-injector-58ff476d66-gvrw8   21m          19Mi
istio-telemetry-5549d784c8-54q89          723m         479Mi
prometheus-748b7f5cf8-79lnf               5110m        66064Mi
```

What We Have Done

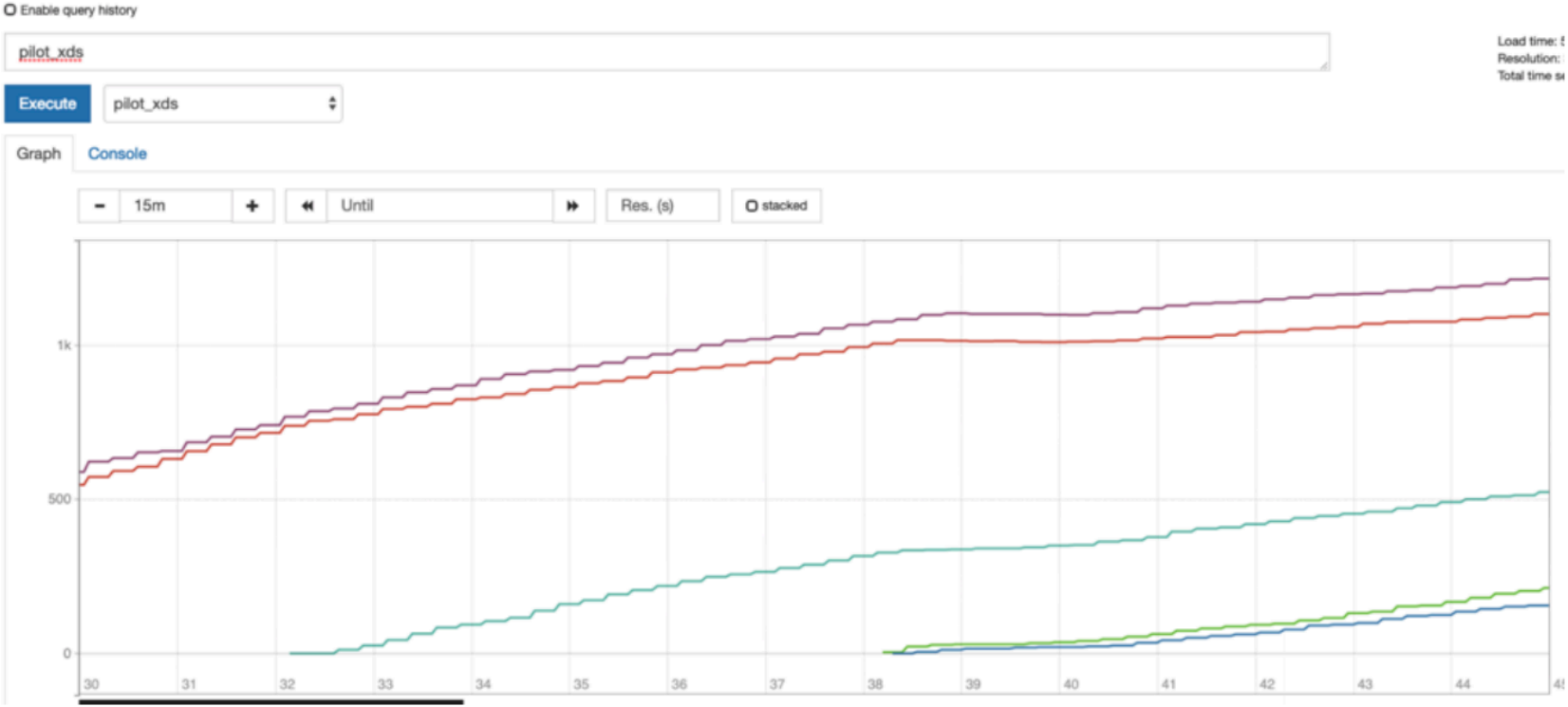
The istio-proxy cpu and memory information

```
root@icp10b1:~# kubectl top pod -n test-ns-201 --containers=true
POD                                          NAME          CPU(cores)   MEMORY(bytes)
istio-ingressgateway-7c5fc45c7f-mdml8        istio-proxy   3m           22Mi
olb-olb-java-deployment-1-557f8ff5b8-wxlqm   olb-java      15m          92Mi
olb-olb-java-deployment-1-557f8ff5b8-wxlqm   istio-proxy   5m           23Mi
olb-olb-java-deployment-10-6c457b5666-82hdn  olb-java      15m          90Mi
olb-olb-java-deployment-10-6c457b5666-82hdn  istio-proxy   5m           23Mi
olb-olb-java-deployment-11-59fcc7dfbd-lslwf  olb-java      12m          91Mi
olb-olb-java-deployment-11-59fcc7dfbd-lslwf  istio-proxy   7m           23Mi
olb-olb-java-deployment-12-945c7fdd6-wdxgn   olb-java      14m          92Mi
olb-olb-java-deployment-12-945c7fdd6-wdxgn   istio-proxy   5m           24Mi
olb-olb-java-deployment-13-7fcc5b596-8vxfw   olb-java      25m          91Mi
olb-olb-java-deployment-13-7fcc5b596-8vxfw   istio-proxy   6m           23Mi
olb-olb-java-deployment-14-5586d648db-2vf59  olb-java      14m          90Mi
```

What We Have Done



What We Have Done



What We Have Done

Base on the namespace isolation environment, using jmeter to distribute the requests. In this test case, I used ansible to run the jmeter in 10 hosts to simulate the real case.

1.Telemetry Information during testing

```
NAME                                     CPU(cores)   MEMORY(bytes)
istio-telemetry-5549d784c8-54q89         3515m        645Mi
```

What We Have Done

2.Jmeter output

```
...
2019/03/30 08:07:35 INFO  - jmeter.reporters.Summariser: summary +   5450 in     6s =  904.3/s Avg:   187 M
2019/03/30 08:07:35 INFO  - jmeter.reporters.Summariser: summary = 300024 in   300s =  999.0/s Avg:   196 M
```

```
2019/03/30 08:31:30 INFO  - jmeter.reporters.Summariser: summary +  25544 in  30.1s =  1280.8/s Avg:   117
2019/03/30 08:31:30 INFO  - jmeter.reporters.Summariser: summary = 243999 in   292s =  1335.1/s Avg:   117
```

We can see that 990/1335=74% which is better than the result in offical result. That is probably caused of having the ingressgateway in each namespace.

What We Have Done

https://istio.io/docs/concepts/performance-and-scalability/

Latency for Istio 1.1.8

The default configuration of Istio 1.1 adds 8ms to the 90th percentile latency of the data plane over the baseline. We obtained these results using the Istio benchmarks for the `http/1.1` protocol, with a 1 kB payload at 1000 requests per second using 16 client connections, 2 proxy workers and mutual TLS enabled.

In upcoming Istio releases we are moving `istio-policy` and `istio-telemetry` functionality into the proxy as `MixerV2`. This will decrease the amount data flowing through the system, which will in turn reduce the CPU usage and latency.

Best Practices

- Use Namespace Isolation feature in a large scale cluster.
- Install ingressgateway for each namespace
- Separate the telemetry component to the exclusive node to avoid more CPU consumption impaction.
- Recommended resource request for critical components to support 6000 pods and 3000 services
  - 6 pilot instances with 4vCPU and 4GB Memory
  - 1 or 2 telemetry instances with 4vCPU and 4GB Memory
- Disable the policy component to increase the traffic throughput.
- Prometheus occupied more CPU and Memory for large scale cluster, change the retention and scrapeInterval

More Tuning Guidance

#	Tuning Knob/Area	Value	Performance Symptoms	Tuning Suggestion
1	keepaliveMaxServerConnectionAge	Default is 30 mins	Uneven Pilot replica load distribution	If there is no uniform distribution of load to pilot replicas, adjust this knob
2	Concurrency	Default is 2	Side Car Resource Utilization and Application Latency	Adjust this parameter to control proxy side car worker threads to reduce resource utilization and also to reduce application latency and improve application throughput If set to 0 (default), then start worker thread for each CPU thread/core.

More Tuning Guidance

3	Telemetry Filters	Default collects all	Significant resource usage by Istio Control plane mainly from telemetry	There are 2 specific suggestions to reduce resources by removing rules and adopters 1) One can collect metrics by error condition 2) One can filter by various rules (stdio, Prom etc.)
4	Tracing	disable	Significant resource costs and latency/throughput impact	Disable tracing in production environments through configuration - Default profile of Istio does not have tracing
5	HPA Thresholds for Telemetry and Gateways	10m/30Mi Default 1000m (telemetry)	Impact on performance of the mesh	Need to adjust the thresholds for specific use cases (Istio proxy access logs are disabled by default)

THANK YOU !