

НАХРЕНА КОЗЕ БАЯН?



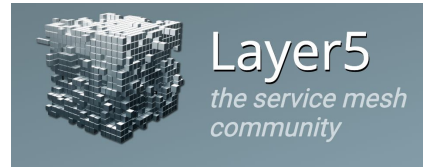
Или Зачем Нам Сервисное Сито

Ant Weiss, Otomato Software



WHOAMI: АНТОН (ЭНТ) БАЙС

 @antweiss



OTOMATO SOFTWARE DELIVERY

<http://otomato.link>

ГОД НАЗАД (ПОСЛЕ РАССКАЗА ОБ ISTIO)

БУДУЩЕЕ СИТА:

- Альтернативы:
 - Conduit (from Buoyant)
 - Consul Connect (Hashicorp)
- Коммерческий вариант:
 - Aspen Mesh (F5 incubator)
- ???



СЕГОДНЯ:

- Istio 1.3 (managed service on GKE)



СЕГОДНЯ:

- Istio 1.3 (managed service on GKE)
- Linkerd 2.0 (вместо Conduit)



СЕГОДНЯ:

- Istio 1.3 (managed service on GKE)
- Linkerd 2.0 (вместо Conduit)
- Consul Connect + Envoy sidecar-proxy



СЕГОДНЯ:

- Istio 1.3 (managed service on GKE)
- Linkerd 2.0 (вместо Conduit)
- Consul Connect + Envoy sidecar-proxy
- Maesh (from Containous)



СЕГОДНЯ:

- Istio 1.3 (managed service on GKE)
- Linkerd 2.0 (вместо Conduit)
- Consul Connect + Envoy sidecar-proxy
- Maesh (from Containous)
- Kuma (from Kong)



СЕГОДНЯ:

- Istio 1.3 (managed service on GKE)
- Linkerd 2.0 (вместо Conduit)
- Consul Connect + Envoy sidecar-proxy
- Maesh (from Containous)
- Kuma (from Kong)
- Mesher (part of Apache Servicecomb)



СЕГОДНЯ:

- Istio 1.3 (managed service on GKE)
- Linkerd 2.0 (вместо Conduit)
- Consul Connect + Envoy sidecar-proxy
- Maesh (from Containous)
- Kuma (from Kong)
- Mesher (part of Apache Servicecomb)
- NSM



СЕГОДНЯ:

- AWS App Mesh
- Aspen Mesh
- Octarine
- Grey Matter
- A10 Secure Service Mesh
- VMWare NSX
- Glasnostic



ВОПРОСЫ:



ВОПРОСЫ:

- Что Такое Сервисное Сито?



ВОПРОСЫ:

- Что Такое Сервисное Сито?
- Зачем Нам Сервисное Сито?



ВОПРОСЫ:

- Что Такое Сервисное Сито?
- Зачем Нам Сервисное Сито?
- Какое Сито Брать?

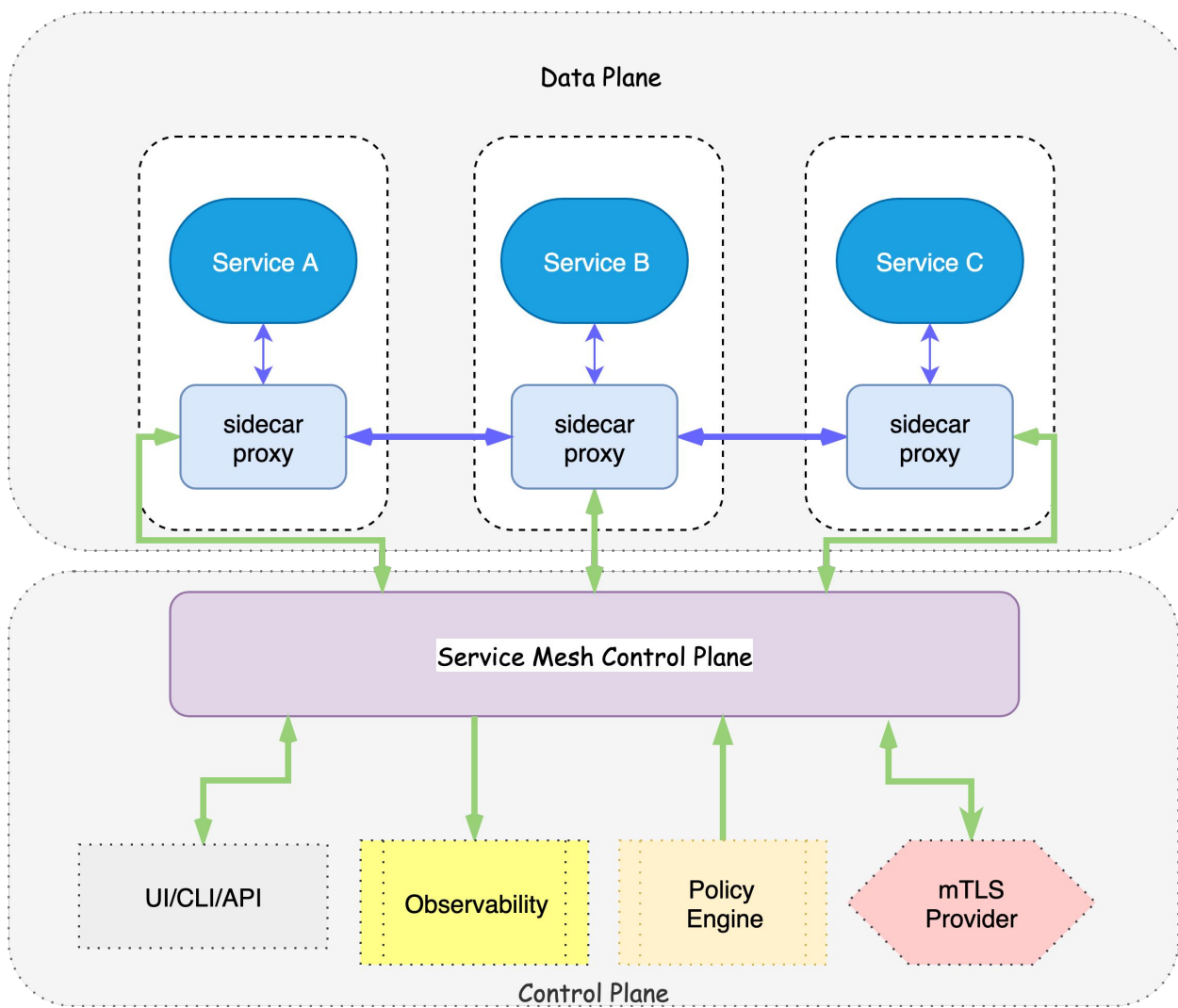


ВОПРОСЫ:

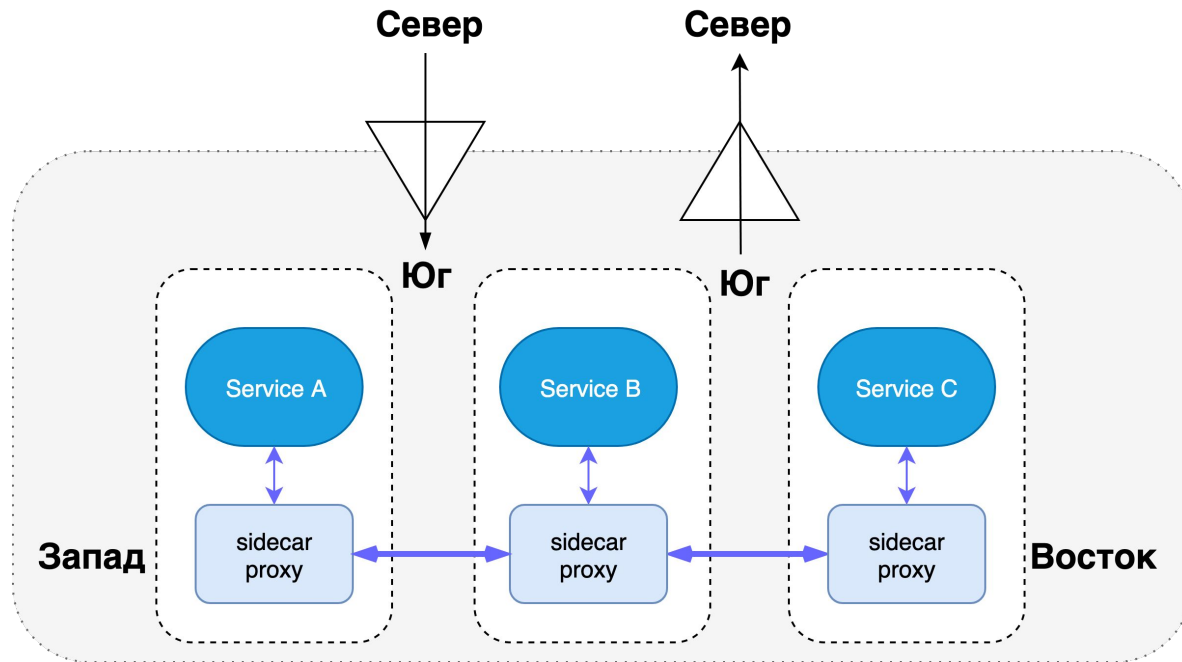
- Что Такое Сервисное Сито?
- Зачем Нам Сервисное Сито?
- Какое Сито Брать?
- Сколько Мне Это Будет Стоить?



4TO?



СИТО ИЛИ ШЛЮЗ?



SMI - THE SERVICE MESH INTERFACE

In partnership with:



Тотальная CRD-изация для Сервисных Сум

SMI - THE SERVICE MESH INTERFACE

Traffic Spec - Как выглядит Трафик?

```
apiVersion: specs.smi-spec.io/v1alpha1
kind: HTTPRouteGroup
metadata:
  name: the-routes
matches:
- name: metrics
  pathRegex: "/metrics"
  methods:
  - GET
- name: health
  pathRegex: "/healthz"
  methods: ["*"]
```

SMI - THE SERVICE MESH INTERFACE

Traffic Access

Control -

Кому Куда Можно?

```
kind: TrafficTarget
apiVersion:
  access.smi-spec.io/v1alpha1
metadata:
  name: path-specific
destination:
  kind: ServiceAccount
  name: service-a
  port: 8080
specs:
- kind: HTTPRouteGroup
  name: the-routes
  matches:
  - metrics
sources:
- kind: ServiceAccount
  name: prometheus
  namespace: default
```

SMI - THE SERVICE MESH INTERFACE

Traffic Split - Разделяй и Властвуй.

```
apiVersion: split.smi-spec.io/v1alpha1
kind: TrafficSplit
metadata:
  name: my-app-split
spec:
  # The root service that clients request
  service: my-app
  # Services inside the namespace with their own selectors, etc
  backends:
  - service: my-app-v1
    weight: 50
  - service: my-app-v2
    weight: 50
```

SMI - THE SERVICE MESH INTERFACE

Traffic Metrics -
Меряем Трафик

```
apiVersion: metrics.smi-spec.io/v1alpha1
kind: TrafficMetrics
resource:
  name: foo-775b9cbd88-ntxs1
  namespace: foobar
  kind: Pod
edge:
  direction: to
  resource:
    name: baz-577db7d977-lsk2q
    namespace: foobar
    kind: Pod
```

SMI - THE SERVICE MESH INTERFACE

Traffic Metrics -
Меряем Трафик

```
apiVersion: metrics.smi-spec.io/v1alpha1
kind: TrafficMetrics
resource:
  name: foo-775b9cbd88-ntxs1
  namespace: foobar
  kind: Pod
edge:
  direction: from
  resource:
    name: baz-577db7d977-lsk2q
    namespace: foobar
    kind: Pod
```


SMI - THE SERVICE MESH INTERFACE

Traffic Metrics -
Меряем Трафик

```
kind: TrafficMetrics
spec:
...
timestamp: 2019-04-08T22:25:55Z
window: 30s
metrics:
- name: p99_response_latency
  unit: seconds
  value: 10m
- name: p90_response_latency
  unit: seconds
  value: 10m
- name: p50_response_latency
  unit: seconds
  value: 10m
- name: success_count
  value: 100
- name: failure_count
  value: 100
```

ЗАЧЕМ?

Управляемость

ЗАЧЕМ?

Управляемость

Наблюдаемость

ЗАЧЕМ?

Управляемость

Наблюдаемость

Устойчивость

ЗАЧЕМ?

Управляемость

Наблюдаемость

Устойчивость

Безопасность

ЗАЧЕМ?

Управляемость

Наблюдаемость

Устойчивость

Безопасность

Прогрессивная
Доставка



КАКОЕ СИТО БРАТЬ?

ISTIO



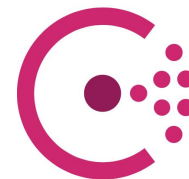
- Эталонная реализация
- Гибридные конфигурации (?)
- Прокси-коляска: Envoy
- Проблемы Перформанса
- Сложность (снижается - 20 CRDs in version 1.3.1)
- Модулярность
- Поддержка OpenTracing (with adapter)
- Интеграция с Prometheus (with adapter)

LINKERD 2.0

- От Первопроходцев
- Только для Куба
- Прокси-коляска: Linkerd (Rust)
- Минимум Конфигурации
- Встроенная телеметрия
- Высокая производительность
- Простота



CONSUL CONNECT



HashiCorp

Consul

- Испытанный боем инструмент
- Гетерогенные Окружения
- Прокси-коляска: Envoy
- HCL/json/UI вместо yaml
- Телеметрия - напрямую с проксей
- SMI - только Traffic Access

MAESH

- Только для Куба
- Прокси: Traefik
- Без Колясок!
- SMI compliant (implements Spec, Acces, Split)
- Конфиг: by annotations on K8S objects
- Доступ к сервисам через `<service>.<namespace>.maesh`



KUMA (KONG MESH)

- Прокси-коляска: Envoy
- Platform-agnostic (Universal and Kubernetes-native)
- Без привязки к SMI
- Для энтерпрайза (?)



Kuma

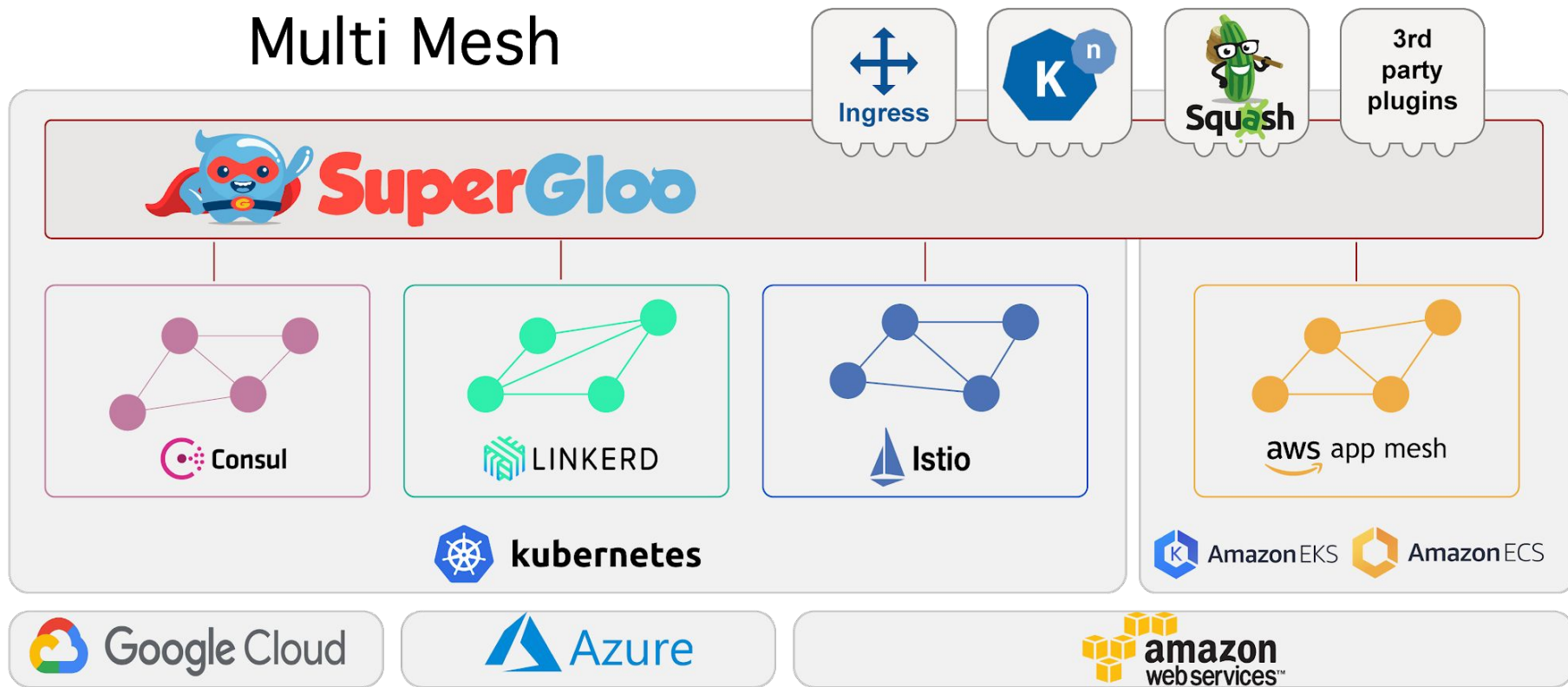
NSM - NETWORK SERVICE MESH

- Сетевое Сито
- Поддержка гибридных и мульти-облачных сценариев
- Соединяем контейнеры в разных кластерах
- Поддержка 'экзотических' протоколов
- Тоннели 'из коробки'
- Динамическая аллокация интерфейсов
- NFV платформа для облаков



ЭКОСИСТЕМА!

Multi Mesh



ЭКОСИСТЕМА!



Layer5

*the service mesh
community*



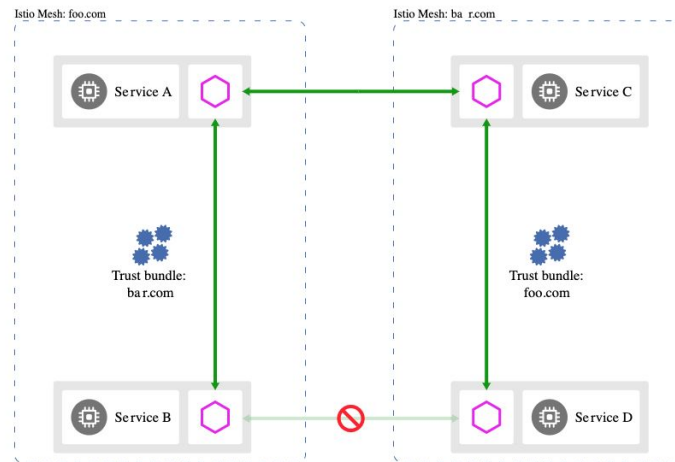
MESHERY

- Ландшафт
- Песочница
- Сравнительный Анализ

ЭКОСИСТЕМА!

<https://github.com/istio-ecosystem>

dns-discovery



Multiple service meshes

БЕНЧМАРКИ

- Kinvolk's service mesh benchmark suite :

<https://github.com/kinvolk/service-mesh-benchmark>

Istio vs. Linkerd 2.0 :

<https://kinvolk.io/blog/2019/05/performance-benchmark-analysis-of-istio-and-linkerd/>

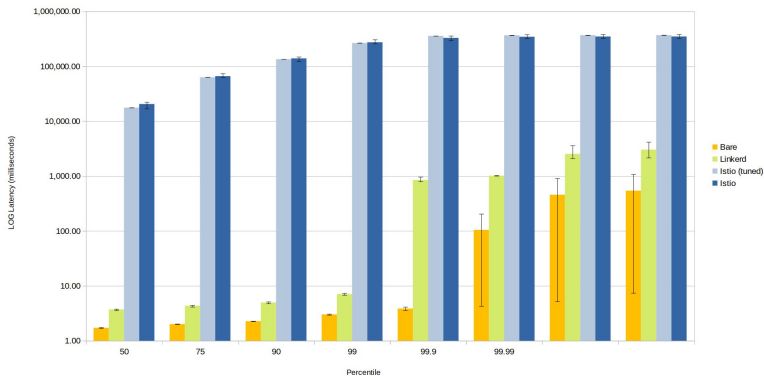
- Layer5 Service Mesh Performance Benchmark Spec :

<https://github.com/layer5io/service-mesh-benchmark-spec>

БЕНЧМАРКИ

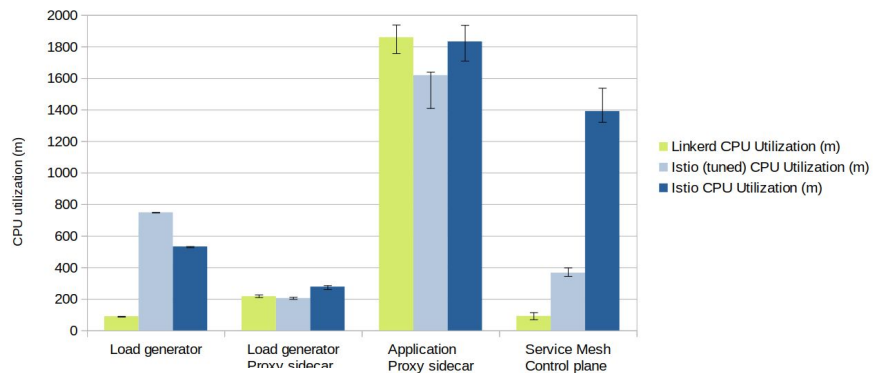
Latency percentiles - bare / Linkerd / Istio / tuned Istio

600RPS, 30m runtime, 4 test runs (2x on 2 clusters each)



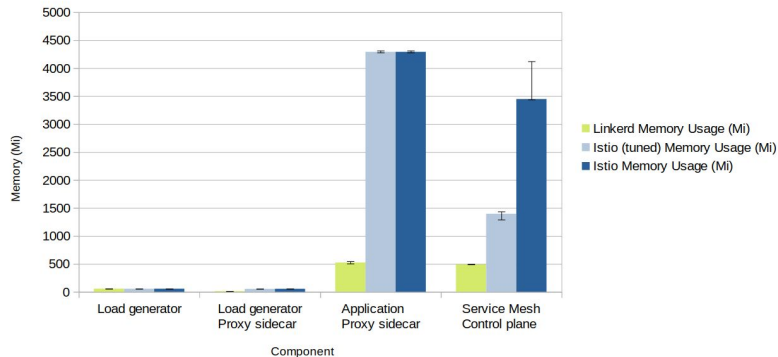
CPU Utilization- Linkerd / Istio / tuned Istio

600RPS, 30s runtime, 4 test runs (2x on 2 clusters each)

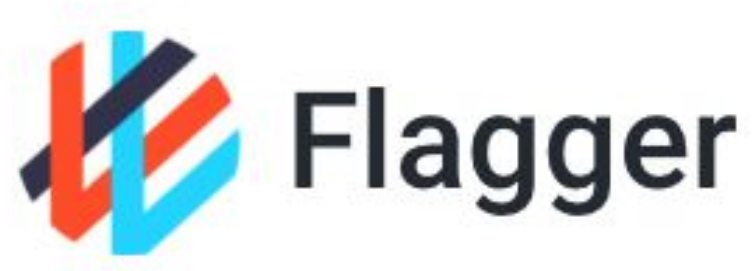


Memory usage - Linkerd / Istio / tuned Istio

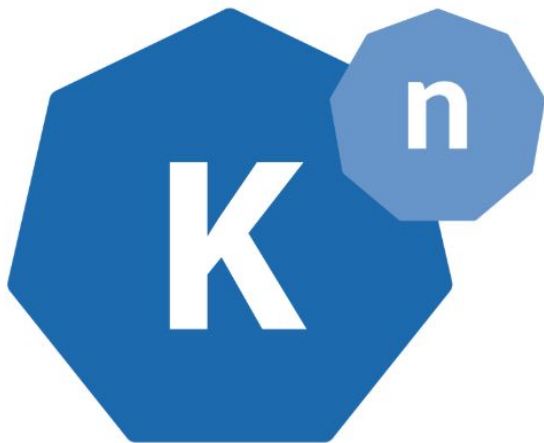
600RPS, 30s runtime, 4 test runs (2x on 2 clusters each)



OTOMATORS



Для БЕССЕРВЕРНИКОВ



Istio

PLONK!

linkerd

ВНЕДРЯЕМ СИТО

- Начинаем с Шлюзов
- Сначала Байпас + Телеметрия
- Сеём Хаос (GlooShot)
- Огин неймспейс
- Думаем о : authz, authn, CA, change control, troubleshooting, upgrade path
- Огин кластер
- Весь мир - одно большое сито