PLATFORM V

Kubernetes Governance **as a Code**

Максим Чудновский & Александр Козлов,
СберТех

Кто мы?

- В разработке Service Mesh с 2018 года;
- Масштаб эксплуатации Service Mesh:
  - 300+ продуктовых команд;
  - 200+ кластеров Kubernetes;
  - 20K+ подов в Service Mesh;
- Дополнительно разрабатываю много полезного вокруг Service Mesh и Kubernetes;
- Подробнее о Synapse: getsynapse.io


Что такое
Kubernetes Governance?

База Kubernetes

- Security

База Kubernetes

- Security
- Networking

База Kubernetes

- Security
- Networking
- Scheduling & Topology

База Kubernetes

- Security
- Networking
- Scheduling & Topology
- Configuration Constraints

Платформенные расширения

- Service Mesh

Платформенные расширения

- Service Mesh
- Secret Vault

Платформенные расширения

- Service Mesh
- Secret Vault
- Audit & Logging

Платформенные расширения

- Service Mesh
- Secret Vault
- Audit & Logging
- Any k8s-native platform tool


Решение #1
От Команды

Идея

> Команда пишет код, напишет и конфиги под кубер

Решение

Заполнить values

```
release_version: "1.10.0"
config_version: "1.10.0"
component_type: service
replicas: 1
kubeAPIIP: 29.66.0.1
indexInstall: ""
kubePort: 443
kubeAPIResolution: STATIC
kubeEgressRouting: TLS
deploy: control
useControl: k8s
federateResolution: DNS
add_drs_scrape: 'true'
system_metrics: 'true'
system_metrics_ca_crt_name: ca.crt
projectedKubeAPI: 'true'
projectedKubeAPI_openshift_cm: 'false’
…
```

Решение

Заполнить values, продолжаем…

```
…
seccompProfile: 'true'
generate_ephemeral_storage: 'false'
affinity: 'true'
affinityMatchKey: 'kubernetes.io/arch'
affinityMatchValues:
- amd64
- ppc64le
- s390x
poddistruptionbudget: 'false'
enabledPriorityClassName: 'false'
priorityClassName: system-cluster-critical
use_storage_sm: 'true'
tenant: '0'
add_kind_sidecar: 'true'
add_kind_sidecar_cp: 'true'
peer_authentication:
…
```

Решение

Заполнить values, продолжаем… и еще…

```
…
k8s_system_metrics:
port: 8481
roleBindings:
useCustom:
install: "false"
index: ""
igeg:
add_credential_mount: 'false'
scrape_metrics_istio_operator:
scrape: 'false'
namespace: synapse-operator
enabled_metric_relabel_configs: 'false'
istio_operator_pods_relabel_regex: ''
add_serviceaccount: 'true'
enabled_metric_relabel_configs: 'true'
istio_pods_relabel_get_label:
…
```

Решение

Написать template

```
{{- $registry := .Values.global.registry }}
{{- $registry_path := .Values.registry_path }}
{{- if eq .Values.install_egress "true" }}
kind: Deployment
apiVersion: apps/v1
metadata:
  name: egw-e-metrics-{{ .Release.Namespace }}
  labels:
    app: egressgateway-e-metrics
    app.kubernetes.io/part-of: istio
    release: istio
    app.kubernetes.io/component: gateways
    istio: egressgateway-e-metrics
    app.kubernetes.io/name: gateways
    p.s.ru/productCode: "bigboss"
    p.s.ru/componentCode: "noup"
    p.s.ru/releaseVersion: "404"
spec:
```

Решение

Написать template

```
{{- $registry := .Values.global.registry }}
{{- $registry_path := .Values.registry_path }}
{{- if eq .Values.install_egress "true" }}
kind: Deployment
apiVersion: apps/v1
metadata:
  name: egw-e-metrics-{{ .Release.Namespace }}
  labels:
    app: egressgateway-e-metrics
    app.kubernetes.io/part-of: istio
    release: istio
    app.kubernetes.io/component: gateways
    istio: egressgateway-e-metrics
    app.kubernetes.io/name: gateways
    p.s.ru/productCode: "bigboss"
    p.s.ru/componentCode: "noup"
    p.s.ru/releaseVersion: "404"
spec:
```

И так еще 1200 строк…

Плюсы и минусы

- Плюсы:
  1. Тривиальное решение

Плюсы и минусы

- Плюсы:
  1. Тривиальное решение

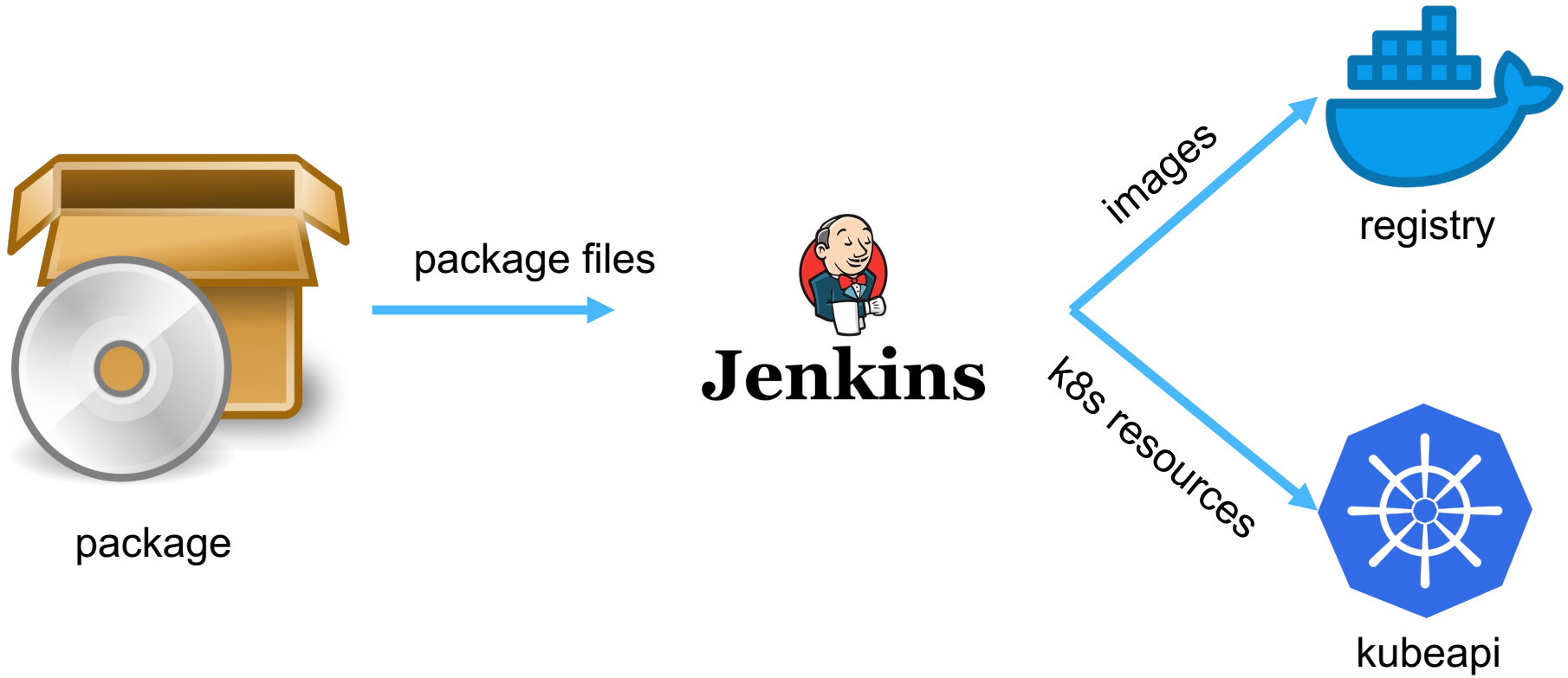
- Минусы
  1. K8s-ниндзя в командах
  2. Шумные соседи
  3. Дрифт API и бесконечная миграция


Решение #2
От Devops’а

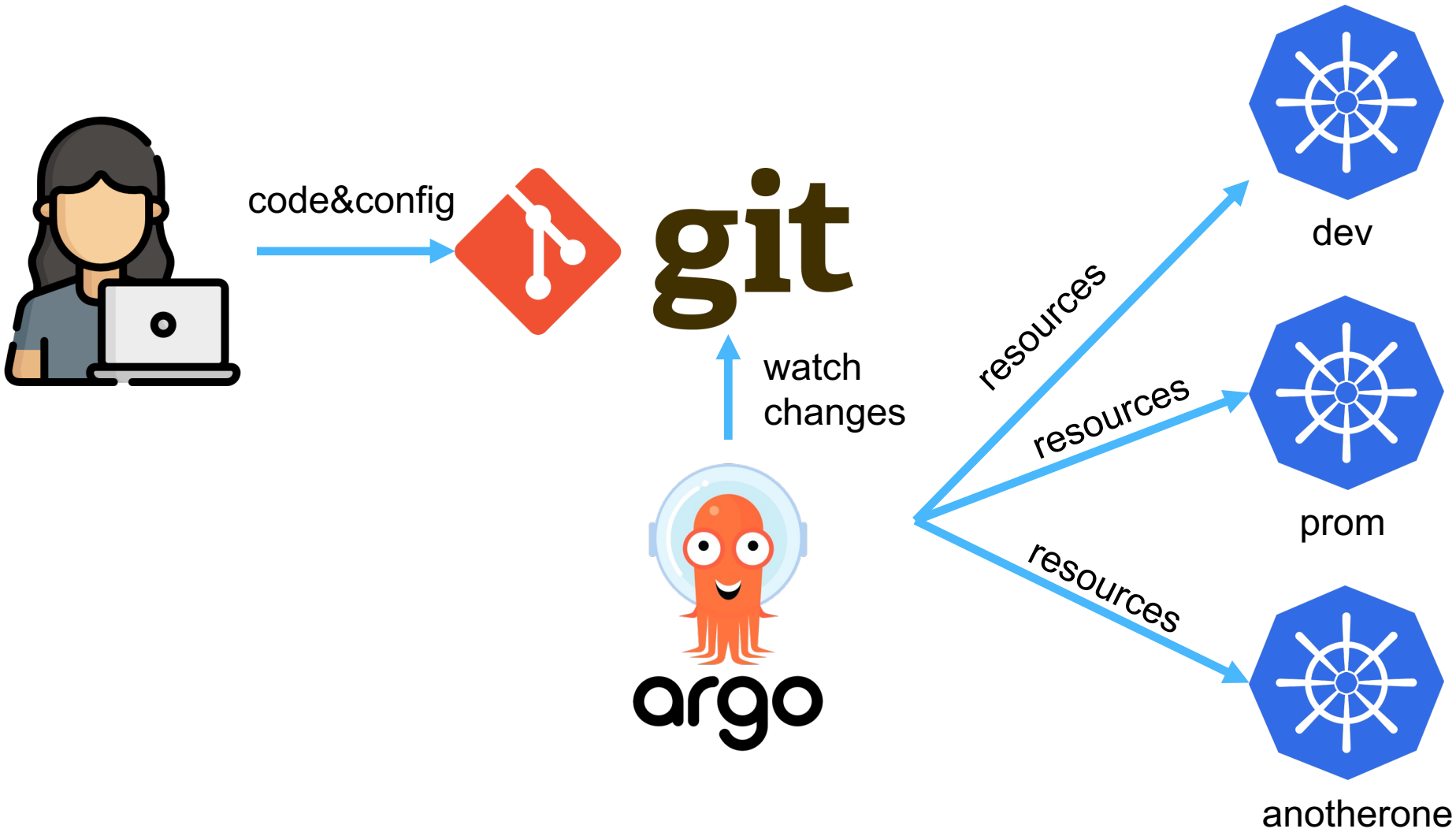
Идея

> DevOps настраивает CI/CD, настроит и кубер

Пример реализации



Пример реализации



Плюсы и минусы

- Плюсы:
  1. Меньшее вовлечение команд

Плюсы и минусы

- Плюсы:
  1. Меньшее вовлечение команд

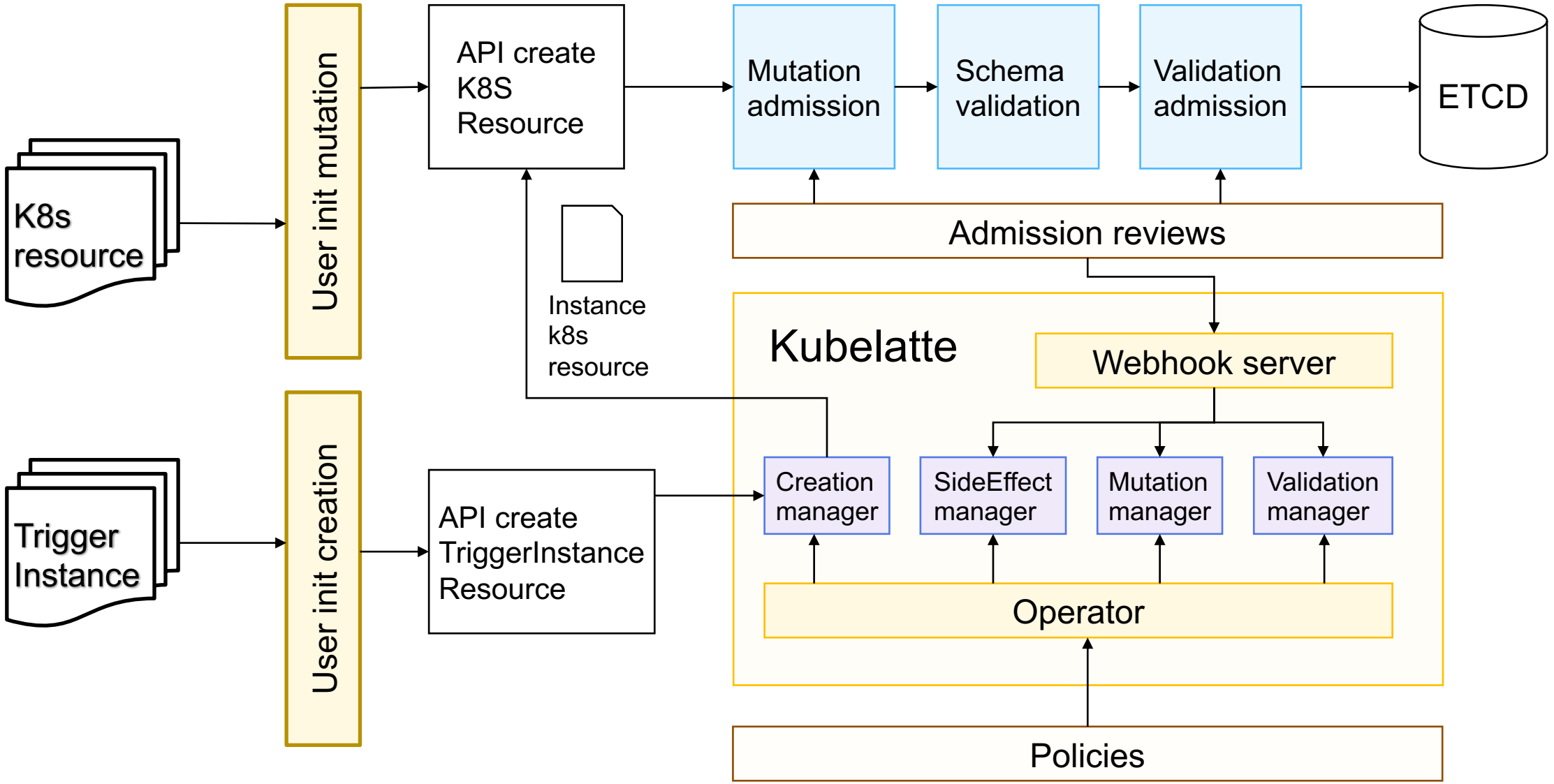
- Минусы
  1. Нетривиальная реализация
  2. Неполный скоуп
  3. Дрифт API и бесконечная миграция, но в CI/CD

Решение #3
От Платформы

Идея

> Если ваша платформа такая умная, то пусть и кубер настроит

Решение


K8s resource
User init mutation
API create K8S Resource
Mutation admission
Schema validation
Validation admission
ETCD
Admission reviews
Instance k8s resource
Kubelatte
Webhook server
Creation manager
SideEffect manager
Mutation manager
Validation manager
Operator
Trigger Instance
User init creation
API create TriggerInstance Resource
Policies

Плюсы и минусы

- Плюсы:
  1. Не требуется участие команд
  2. Не требуется модификация CI/CD
  3. Централизованные политики и контроли

Плюсы и минусы

- Плюсы:
  1. Не требуется участие команд
  2. Не требуется модификация CI/CD
  3. Централизованные политики и контроли

- Минусы
  1. Нужен Policy Engine

Как мы решили проблему?

Kubelatte

- Policy Engine для Kubernetes;
- Enjoy **Kube**rnetes as a Cup of **Latte**.

Почему Kubelatte?

- ~~Не смогли нагуглить про Kyverno~~

Почему Kubelatte?

- ~~Не смогли нагуглить про Kyverno~~

	API CRD	Mutation	Creation	Validation	SideEffect	Rego
Kyverno	+	+	+	+	-	-
GateKeeper	+	+	-	+	-	+
Kubelatte	+	+	+	+	+	+

Почему Kubelatte?

> Регистрация в РРПО,
> сертификация и т.п.
> А главное «Наше родное» ☺

Kubelatte Templates

- Шаблоны всех целевых ресурсов;

```yaml
apiVersion: kubelatte.synapse.sber/v1alpha1
kind: Template
metadata:
  name: any-deployment
spec:
  apiVersion: networking.k8s.io/v1
  kind: Ingress
  data: |-
      spec:
        template:
          spec:
            securityContext:
              runAsNonRoot: true
```

Kubelatte Templates

- Шаблоны всех целевых ресурсов;

➢ Сложные сценарии реализуются через GO-шаблонизацию.

```
data: |-
    spec:
      template:
        spec:
          containers:
           {{`{{% range .spec.template.spec.containers %}}`}}
            - name: {{`{{% .name %}}`}}
              imagePullPolicy: Always
           {{`{{% end %}}`}}
```

Kubelatte Triggers

- Определение точки мутации;
- Связь шаблонов с исходными ресурсами;
- Стратегии применения шаблонов – replace, merge, sideEffect;

```yaml
apiVersion: kubelatte.synapse.sber/v1alpha1
kind: Trigger
metadata:
  name: mutation-any-deployment
spec:
  mutationConfigs:
    - match:
        kinds:
          - apiGroups:
              - '*'
            kinds:
              - Deployment
              - StatefulSet
      name: all-deployment
      templateRefs:
        - ns/any-deployment
      updateStrategy: merge
```

Kubelatte TriggerInstances


```yaml
apiVersion: kubelatte.synapse.sber/v1alpha1
kind: TriggerInstance
metadata:
annotations:
  kblt/creation-ingress: enabled
spec: {}
```

- Аналог Trigger для ситуации, когда должен быть создан новый ресурс;
- Ресурс-якорь для работы встроенной уборки мусора.

Kubelatte Scopes

- Определение точки валидации;
- Определение правил и стратегии валидации;
- Поддержка простых правил и Rego-выражений

```yaml
apiVersion: kubelatte.synapse.sber/v1alpha1
kind: Scope
metadata:
  name: simple-validation
spec:
  type: validation
  items:
    - name: validation-rule
      match:
        kinds:
          - apiGroups:
              - '*'
            kinds:
              - Pod
      rule:
        simples:
          - name: rule-1
            path: spec.volumes[?configMap == null]
            value: .*
            action: deny
            message: “Обнаружены элементы, несоотв. политике”
```

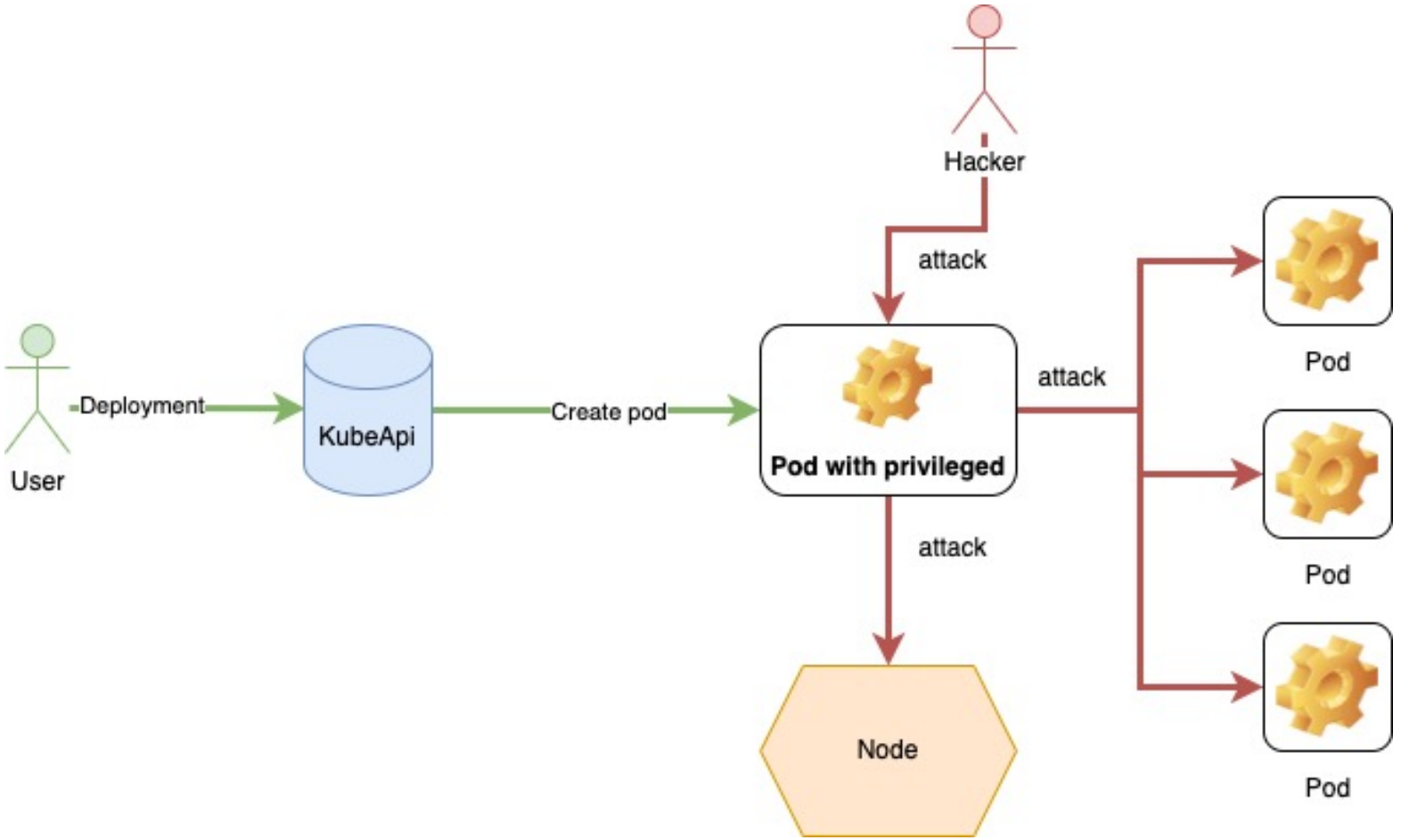
Kubernetes Governance via Kubelatte

База Kubernetes

- **Security**

Кейс – контроль запуска подов

Контроль и валидация securityContext



Кейс – контроль запуска подов

Контроль и валидация securityContext

Пример Pod:

```
kind: Pod
  metadata:
     name: nginx
spec:
  containers:
    - name: nginx
      image: nginx
      securityContext:
        privileged: true
```

Кейс – контроль запуска подов

Контроль и валидация securityContext

Пример Pod:

```
kind: Pod
  metadata:
     name: nginx
spec:
  containers:
    - name: nginx
      image: nginx
      securityContext:
        privileged: true
```

Кейс – контроль запуска подов

Контроль и валидация securityContext

Пример Scope:

```
kind: Scope
spec:
  type: validation
  items:
   - name: kb-ose-simples
     rule:
       simples:
        - action: deny
          message: 'Запрещено запускать привилегированные контейнеры
(securityContext.privileged: true)’
          name: securityContext-privileged
          path: 'to_string(contains(spec.containers[].securityContext.privil
          value: 'true'
```

Кейс – контроль запуска подов

Контроль и валидация securityContext

Без Kubelatte

- Запуск в привилегированном режиме
- Нет контроля лишних привилегий
- Нет контроля рутового пользователя
- Ручной контроль каждого пода

Кейс – контроль запуска подов

Контроль и валидация securityContext

Без Kubelatte

- Запуск в привилегированном режиме
- Нет контроля лишних привилегий
- Нет контроля рутового пользователя
- Ручной контроль каждого пода

С Kubelatte

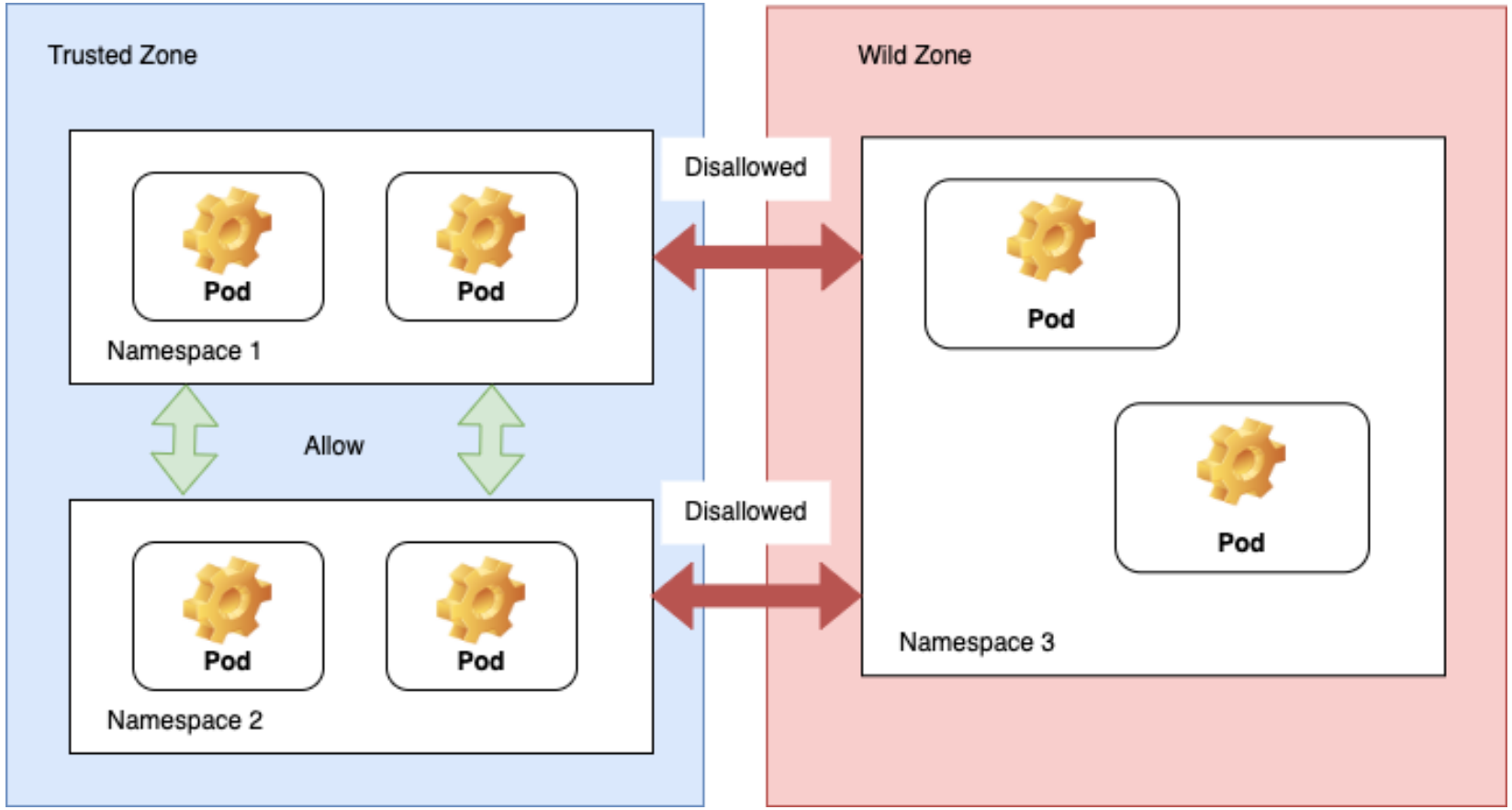
- 1 конфиг с политиками
- Контроль используемых привилегий
- Запрет повышения привилегий
- Автоматический контроль запускаемых подов

База Kubernetes

- ✓Security
- ➢**Networking**

Кейс – дефолтные network policies

Создание дефолтных NetworkPolicy при создании проекта k8s



Кейс – дефолтные network policies

Создание дефолтных NetworkPolicy при создании проекта k8s

Пример политики запрещающей все внешние подключения

```yaml
kind: NetworkPolicy
apiVersion: networking.k8s.io/v1
metadata:
  name: deny-by-default
spec:
  podSelector: {}
  policyTypes:
    - Ingress
```

Кейс – дефолтные network policies

Создание дефолтных NetworkPolicy при создании проекта k8s

Пример политики разрешающей внешнее подключение

```
kind: NetworkPolicy
apiVersion: networking.k8s.io/v1
metadata:
  name: allow-from-metrics
spec:
  podSelector: {}
  ingress:
    - from:
      - namespaceSelector:
        matchLabels:
        kubernetes.io/metadata.name:
kube-metrics-system
  policyTypes:
    - Ingress
```

Кейс – дефолтные network policies

Создание дефолтных NetworkPolicy при создании проекта k8s

Пример Template:

```
kind: Template
metadata:
  name: deny-all
spec:
  apiVersion: networking.k8s.io/v1
  kind: NetworkPolicy
  data: |-
    metadata:
      name: deny-all
    spec:
      podSelector: {}
      policyTypes:
        - Ingress
```

Кейс – дефолтные network policies

Создание дефолтных NetworkPolicy при создании проекта k8s

Пример Trigger:

```yaml
kind: Trigger
metadata:
  name: create-np-deny
spec:
  creationConfigs:
    - annotationNamespace: kblt.np
      annotationTrigger: deny
      name: creation-np-deny
      templateRefs:
        - ns/deny-all
```

Кейс – дефолтные network policies

Создание дефолтных NetworkPolicy при создании проекта k8s

Без Kubelatte

- Настройка под каждый проект сетевого доступа вручную
- Нет контроля что сетевой доступ ограничен

Кейс – дефолтные network policies

Создание дефолтных NetworkPolicy при создании проекта k8s

Без Kubelatte

- Настройка под каждый проект сетевого доступа вручную
- Нет контроля что сетевой доступ ограничен

С Kubelatte

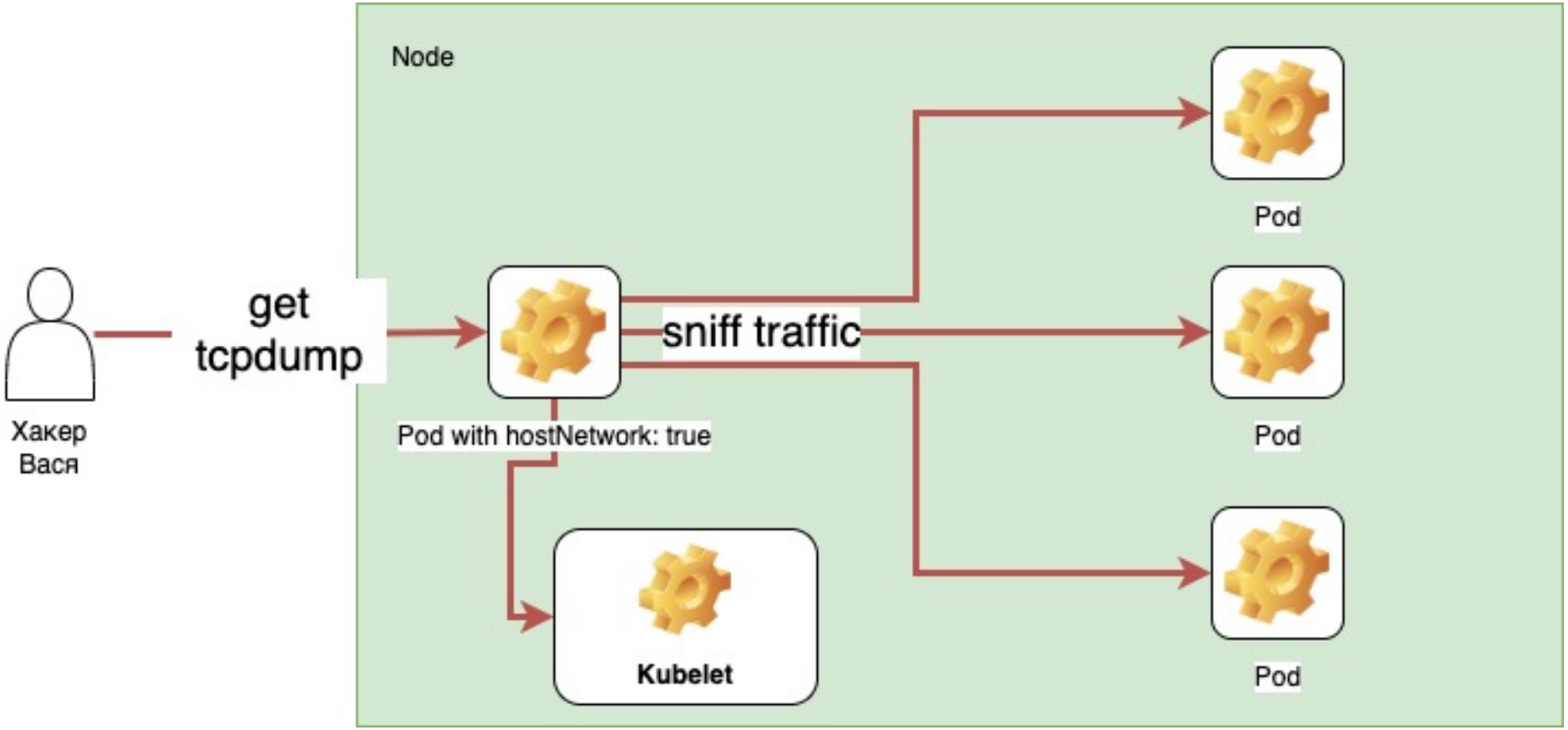
- Автоматическое создание из шаблонов с возможностью параметризации
- Централизованное управление политиками доступа

База Kubernetes

- ✓Security
- ✓Networking
- ➢**Scheduling & Topology**

Кейс – контроль HostNetwork Workload

Запрещено прибегать к HostNetwork



Кейс – контроль HostNetwork Workload

Запрещено прибегать к HostNetwork

Пример Pod:

```
kind: Pod
  metadata:
     name: nginx
spec:
  hostNetwork: true
  containers:
    – name: nginx
      image: nginx
      …
```

Кейс – контроль HostNetwork Workload

Запрещено прибегать к HostNetwork

Пример Pod:

```
kind: Pod
  metadata:
     name: nginx
spec:
  hostNetwork: true
  containers:
    - name: nginx
      image: nginx
      …
```

Кейс – контроль HostNetwork Workload

Запрещено прибегать к HostNetwork

Пример Scope:

```
kind: Scope
spec:
  type: validation
  items:
   - name: kb-ose-simples
     rule:
       simples:
        - action: deny
          message: 'Запрещено использовать hostNetwork (hostNetwork: true)'
          name: hostNetwork-deny
          path: 'to_string((spec.hostNetwork == `true`))'
          value: 'true'
```

Кейс – контроль HostNetwork Workload

Запрещено прибегать к HostNetwork

Без Kubelatte

- Использование флага где не надо
- Нет возможности запрещения использования

Кейс – контроль HostNetwork Workload

Запрещено прибегать к HostNetwork

Без Kubelatte

- Использование флага где не надо
- Нет возможности запрещения использования

С Kubelatte

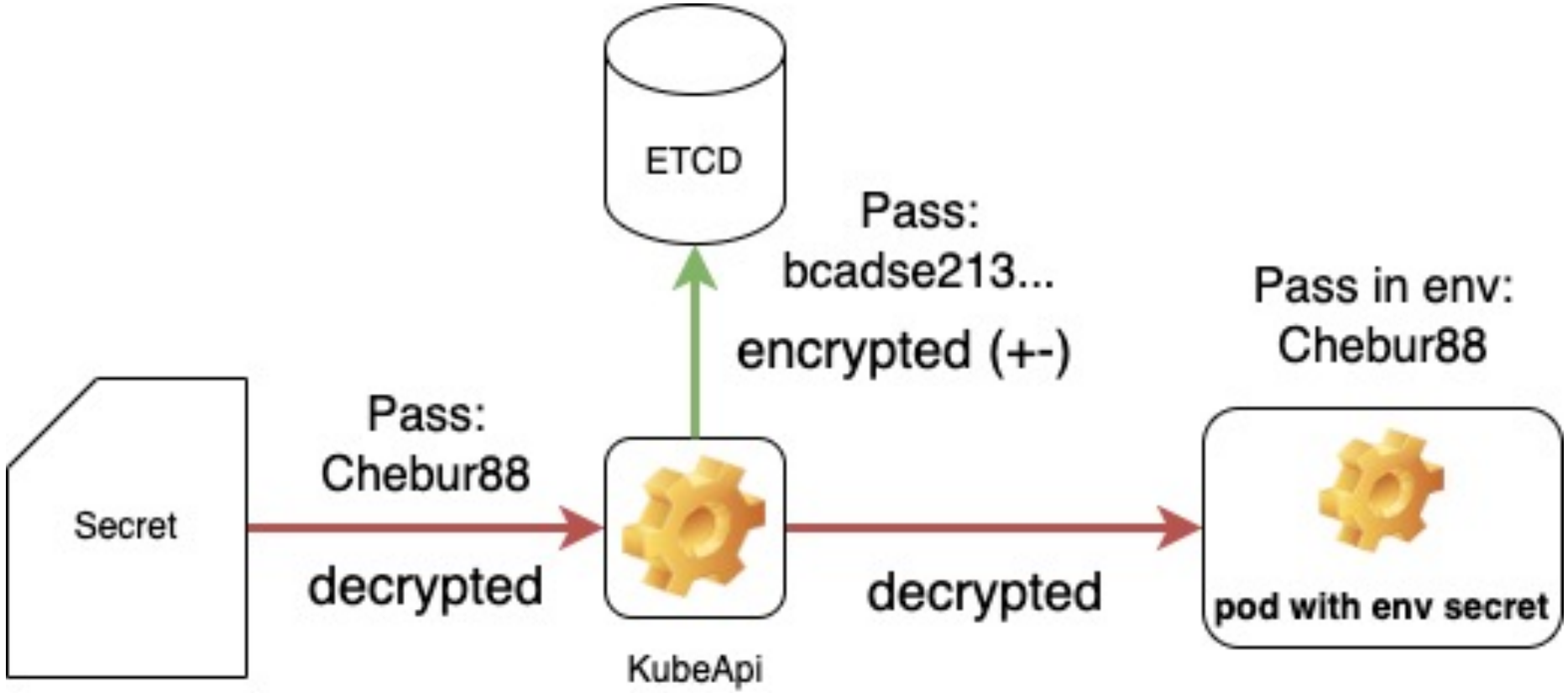
- Контроль исключений
- Запрет в общих случаях использования

База Kubernetes

- ✓Security
- ✓Networking
- ✓Scheduling & Topology
- ➢**Configuration Constraints**

Кейс – контроль секретов в CM

Запрещаем экспортировать секреты в ENV



Кейс – контроль секретов в CM

Запрещаем экспортировать секреты в ENV

Пример Pod:

```
kind: Pod
  metadata:
     name: nginx
spec:
  containers:
    - name: nginx
      image: nginx
      envFrom:
        - secretRef:
            name: my-password
      …
```

Кейс – контроль секретов в CM

Запрещаем экспортировать секреты в ENV

Пример Pod:

```
kind: Pod
  metadata:
     name: nginx
spec:
  containers:
    - name: nginx
      image: nginx
      envFrom:
        - secretRef:
            name: my-password
      …
```

Кейс – контроль секретов в CM

Запрещаем экспортировать секреты в ENV

Пример Scope:

```
kind: Scope
spec:
  type: validation
  items:
   - name: kb-ose-simples
     rule:
       simples:
        - action: deny
          message: 'Запрещено использовать env from secret’
          name: env-secret-deny
          path: ‘spec.containers[].envFrom[].secretRef’
          value: ‘.*'
```

Кейс – контроль секретов в CM

Запрещаем экспортировать секреты в ENV

Без Kubelatte

- Нет системы контроля за безопасной доставкой секретов в контейнер
- Есть вероятность компрометации доступов через переменные окружения

Кейс – контроль секретов в CM

Запрещаем экспортировать секреты в ENV

Без Kubelatte

- Нет системы контроля за безопасной доставкой секретов в контейнер
- Есть вероятность компрометации доступов через переменные окружения

С Kubelatte

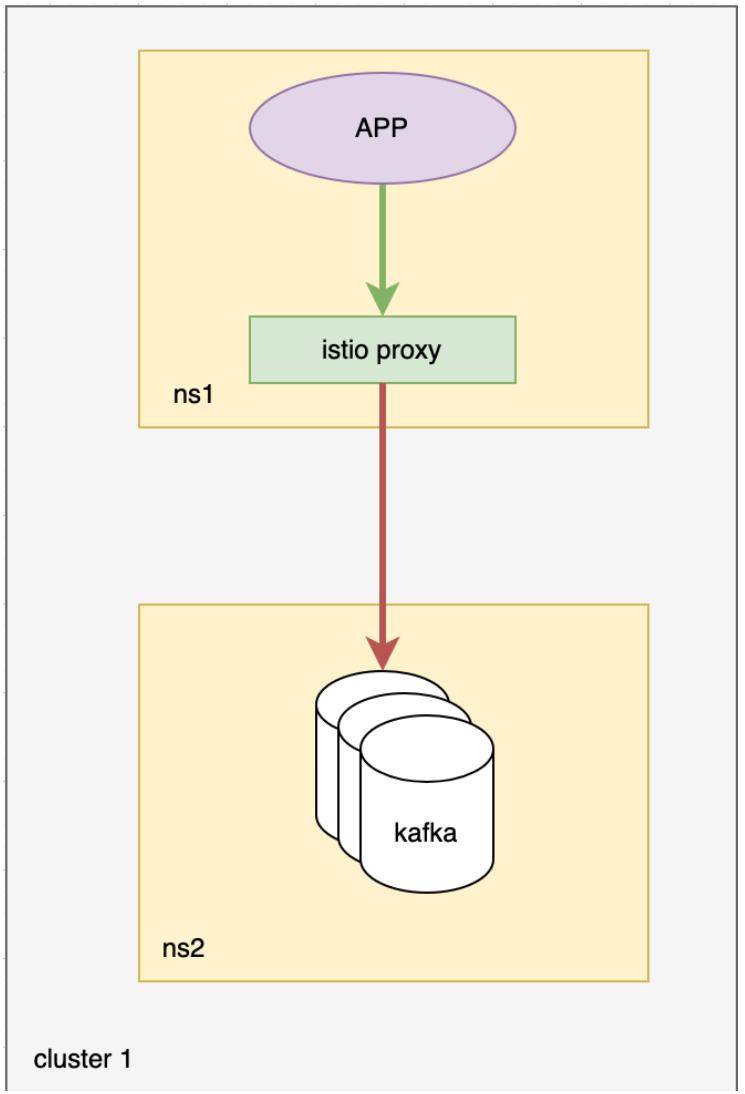
- Подключение секретов к контейнеру контролируется политиками безопасности установленными на кластере
- Вероятность компрометации доступов через переменные окружения снижена

Платформенные расширения

- Service Mesh

Кейс – дефолтные маршруты Istio

Деплой приложения с интеграцией с кафкой



Но нет доступа

Кейс – дефолтные маршруты Istio

Деплой приложения с интеграцией с кафкой

Конфигурации Istio

- ServiceEntry

  …

Кейс – дефолтные маршруты Istio

Деплой приложения с интеграцией с кафкой

Конфигурации Istio

- ServiceEntry

  …

- DestinationRule
- VirtualService
- Gateway

Кейс – дефолтные маршруты Istio

Деплой приложения с интеграцией с кафкой

Пример Template:

```
kind: Template
metadata:
  name: kafka-se
spec:
  apiVersion: networking.istio.io/v1
  kind: ServiceEntry
  data: |-
    metadata:
      name: kafka-se
    spec:
      hosts:
       - bootstrap1.example.dev
      location: MESH_EXTERNAL
      ports:
       - number: 9092
         name: kafka-9092
         protocol: kafka
      resolution: DNS
      endpoints:
         …
```

Кейс – дефолтные маршруты Istio

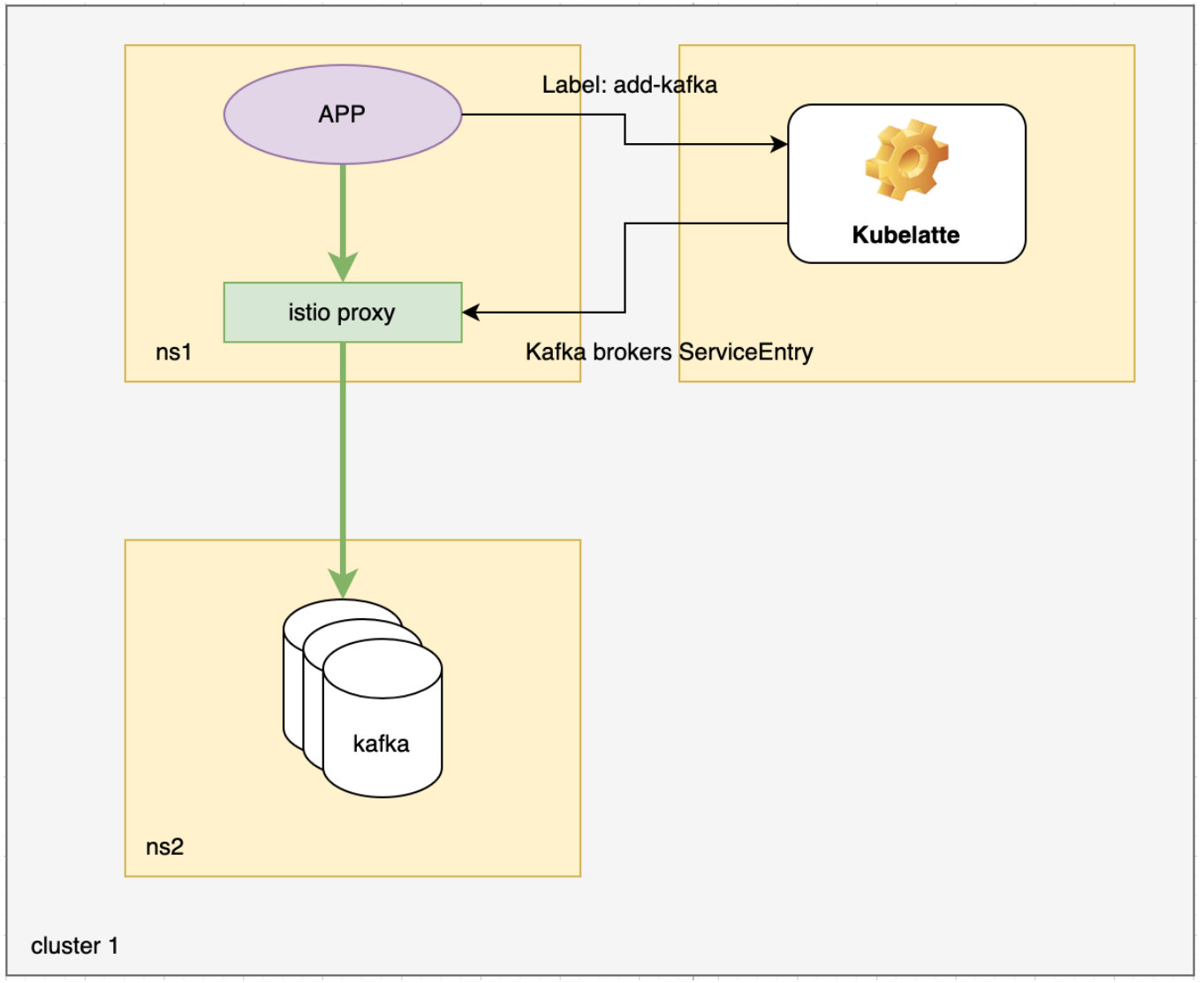
Деплой приложения с интеграцией с кафкой

Пример Trigger:

```yaml
kind: Trigger
metadata:
  name: create-kafka-se
spec:
  creationConfigs:
    - annotationNamespace: kblt.kafka
      annotationTrigger: se
      name: create-kafka-se
      templateRefs:
        - ns/kafka-se
```

Кейс – дефолтные маршруты Istio

Деплой приложения с интеграцией с кафкой



Кейс – дефолтные маршруты Istio

Деплой приложения с интеграцией с кафкой

Без Kubelatte

- Экспертиза настройки Istio в каждой команде
- Надо знать где брокеры все командам
- Сертификаты и настройка дополнительных конфигураций для подключения

Кейс – дефолтные маршруты Istio

Деплой приложения с интеграцией с кафкой

Без Kubelatte

- Экспертиза настройки Istio в каждой команде
- Надо знать где брокеры все командам
- Сертификаты и настройка дополнительных конфигураций для подключения

С Kubelatte

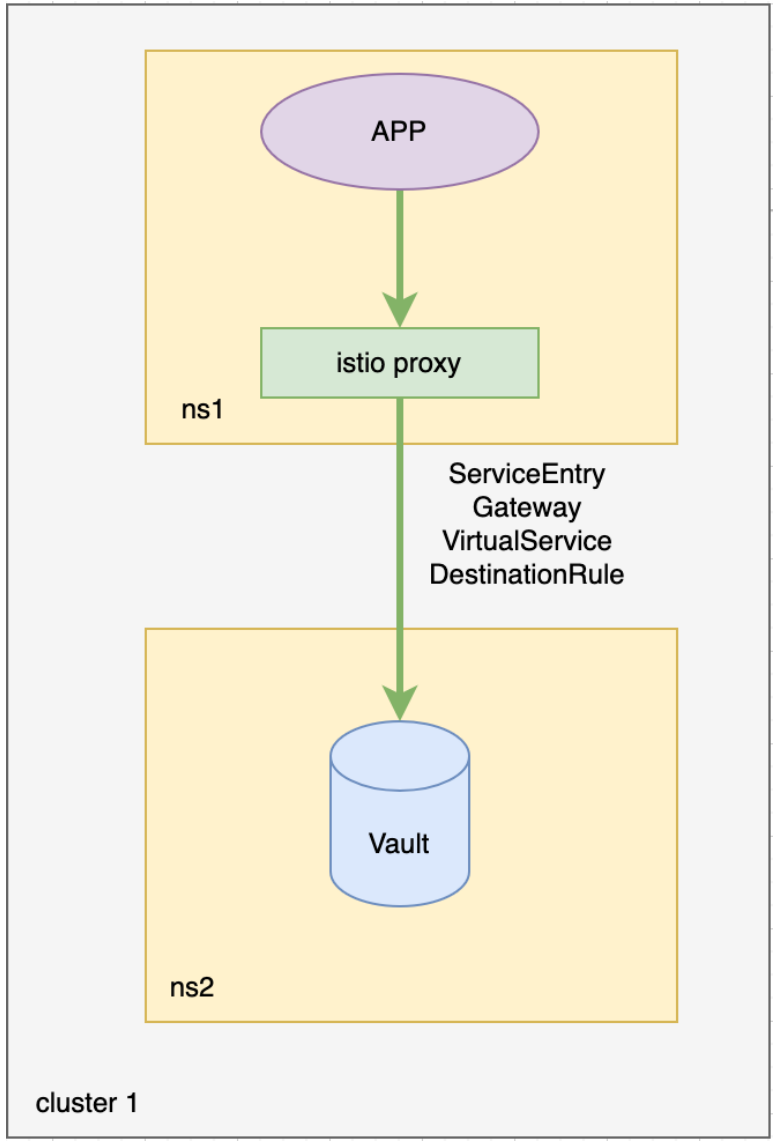
- Не надо уметь в Istio
- Не надо знать топологию стенда
- Не надо вручную настраивать безопасность подключения

Платформенные расширения

- ✓Service Mesh
- ➢**Secret Vault**

Кейс – подключение vault

Подключение к vault

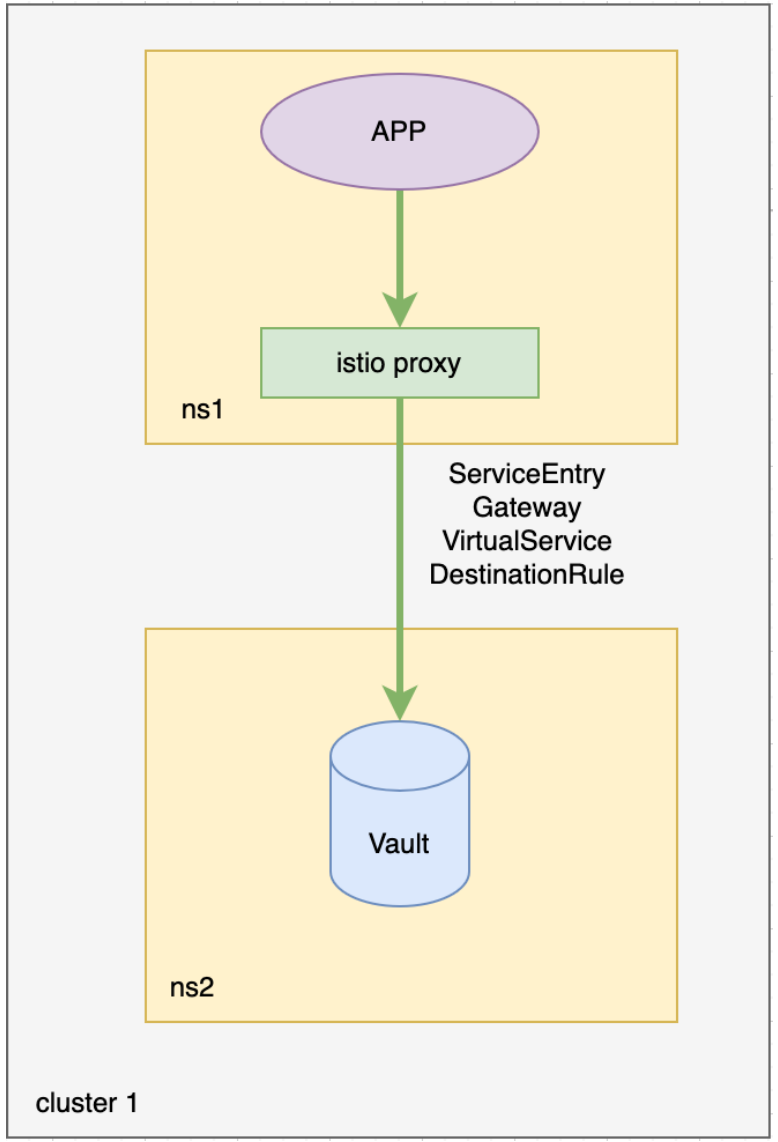


Кейс – подключение vault

Подключение к vault

- Конфигурационный ад

```
annotations:
  vault.hashicorp.com/namespace: 'ns'
  vault.hashicorp.com/role: 'role'
  vault.hashicorp.com/auth-path: ''
  vault.hashicorp.com/agent-pre-populate-only: 'false'
  vault.hashicorp.com/agent-run-as-same-user: 'true'
  vault.hashicorp.com/agent-inject: 'true'
  vault.hashicorp.com/agent-init-first: 'false'
  vault.hashicorp.com/agent-pre-populate: 'true'
  vault.hashicorp.com/agent-limits-cpu: 100m
  vault.hashicorp.com/agent-limits-mem: 100m
  vault.hashicorp.com/agent-requests-cpu: 100m
  vault.hashicorp.com/agent-requests-mem: 100m
  vault.hashicorp.com/agent-inject-secret-
vault_ca.cer: 'true'
  …
```

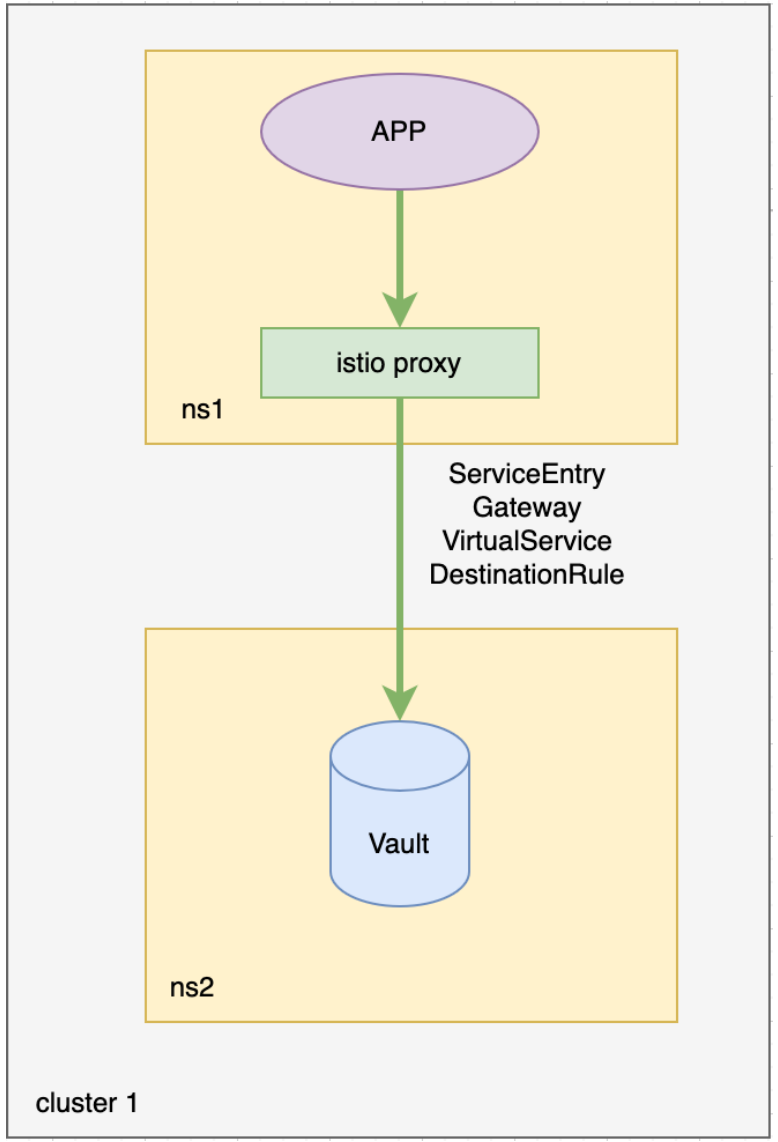


Кейс – подключение vault

Подключение к vault

- Конфигурационный ад … и это только часть

```
annotations:
  vault.hashicorp.com/namespace: 'ns'
  vault.hashicorp.com/role: 'role'
  vault.hashicorp.com/auth-path: ''
  vault.hashicorp.com/agent-pre-populate-only: 'false'
  vault.hashicorp.com/agent-run-as-same-user: 'true'
  vault.hashicorp.com/agent-inject: 'true'
  vault.hashicorp.com/agent-init-first: 'false'
  vault.hashicorp.com/agent-pre-populate: 'true'
  vault.hashicorp.com/agent-limits-cpu: 100m
  vault.hashicorp.com/agent-limits-mem: 100m
  vault.hashicorp.com/agent-requests-cpu: 100m
  vault.hashicorp.com/agent-requests-mem: 100m
  vault.hashicorp.com/agent-inject-secret-
vault_ca.cer: 'true'
  …
```



Кейс – подключение vault

Подключение к vault

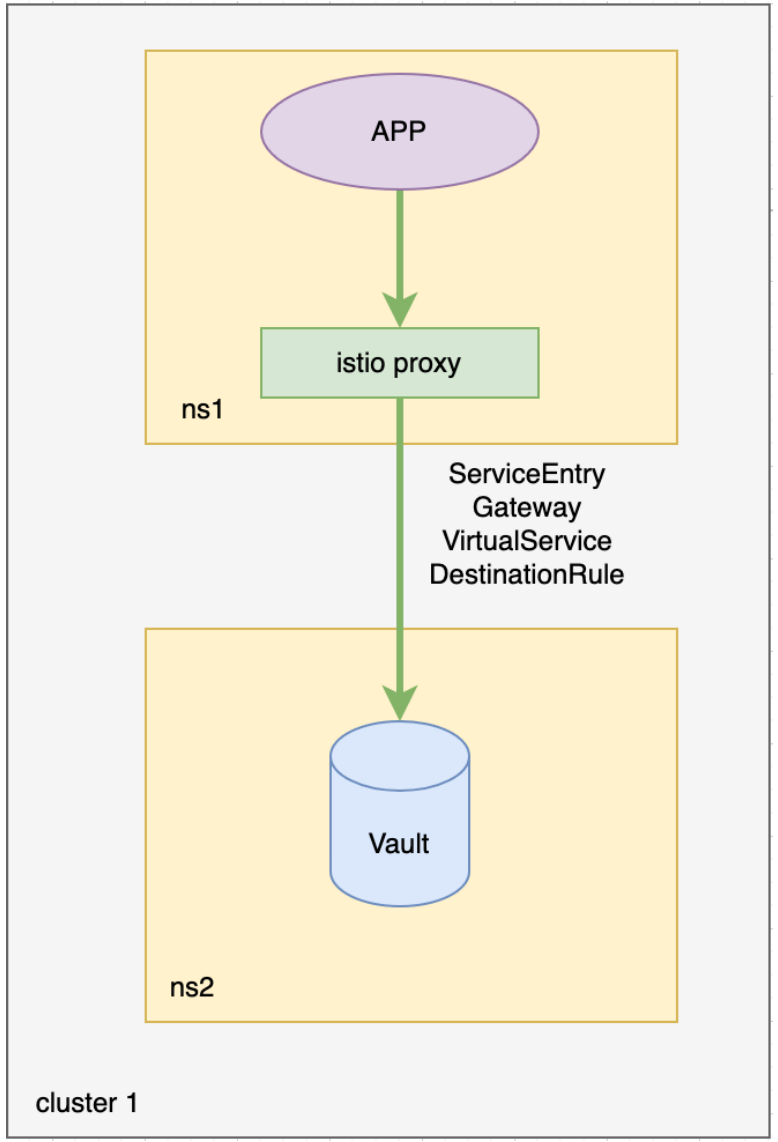
- Конфигурационный ад
- Набор конфигураций Istio

ServiceEntry

Gateway

VirtualService

Destination rule



Кейс – подключение vault

Подключение к vault

Пример Template для конфигурации подключения:

```
kind: Template
metadata:
  name: vault-gw
spec:
  apiVersion: networking.istio.io/v1
  kind: Gateway
  data: |-
    metadata:
      name: vault-gw
    spec:
      selector:
        app: egress
      servers:
        - hosts:
           - vault.example.dev
          port:
            name: tls-54320
            number: 54320
            protocol: TLS
          tls:
            mode: ISTIO_MUTUAL
```

Кейс – подключение vault

Подключение к vault

Пример Template для мутации пода:

```
kind: Template
metadata:
  name: vault-mt
spec:
 data: |-
   metadata:
     annotations:
       vault.hashicorp.com/namespace:
'ns’
       vault.hashicorp.com/role: 'role’
       vault.hashicorp.com/auth-path: ''
...
```

Кейс – подключение vault

Подключение к vault

Пример Trigger для создания конфигураций Istio:

```yaml
kind: Trigger
metadata:
  name: create-vault-configs
spec:
  creationConfigs:
    - annotationNamespace: kblt.vault
      annotationTrigger: cfg
      name: create-vault-configs
      templateRefs:
        - ns/vault-gw
        - ns/vault-se
        - ns/vault-dr
        - ns/vault-vs
```

Кейс – подключение vault

Подключение к vault

Пример Trigger для мутации пода:

```
kind: Trigger
metadata:
  name: vault-mt
spec:
  mutationConfigs:
    - match:
       …selector
      operations:
        - CREATE
        - UPDATE
      scope: Namespaced
      name: add-vault
      templateRefs:
        - ns/vault-mt
      updateStrategy: merge
```

Кейс – подключение vault

Подключение к vault

Без Kubelatte

- Экспертиза настройки и подключения к vault в каждой команде
- Очень много аннотаций, вероятность ошибки высока
- Каждой команде надо построить маршрут к vault

Кейс – подключение vault

Подключение к vault

Без Kubelatte

- Экспертиза настройки и подключения к vault в каждой команде
- Очень много аннотаций, вероятность ошибки высока
- Каждой команде надо построить маршрут к vault

С Kubelatte

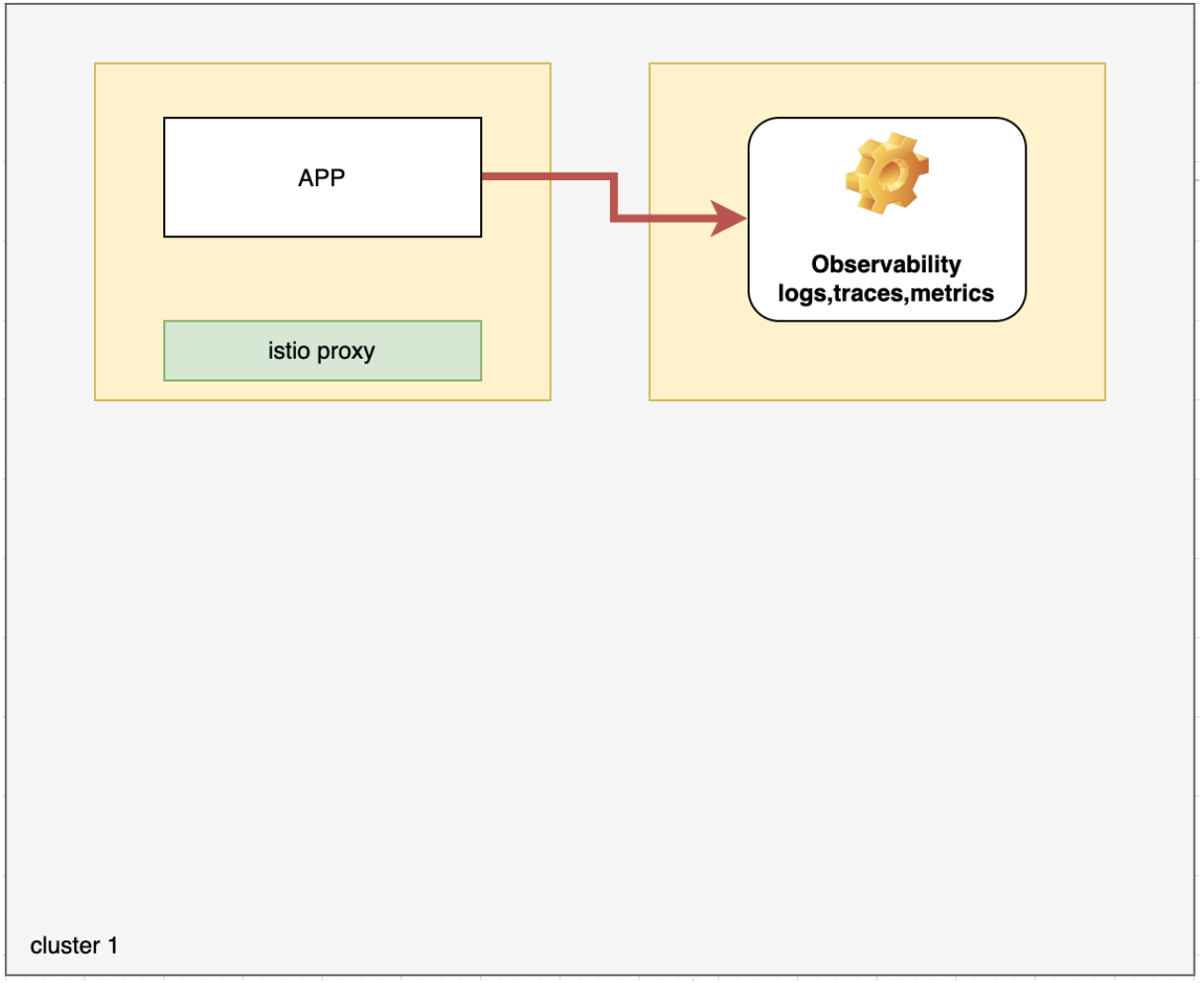
- Генерация аннотаций по шаблону
- Автоматическое создание маршрута
- Актуальность конфигурации

Платформенные расширения

- ✓Service Mesh
- ✓Secret Vault
- ➢**Audit & Logging**

Кейс – logging sidecar

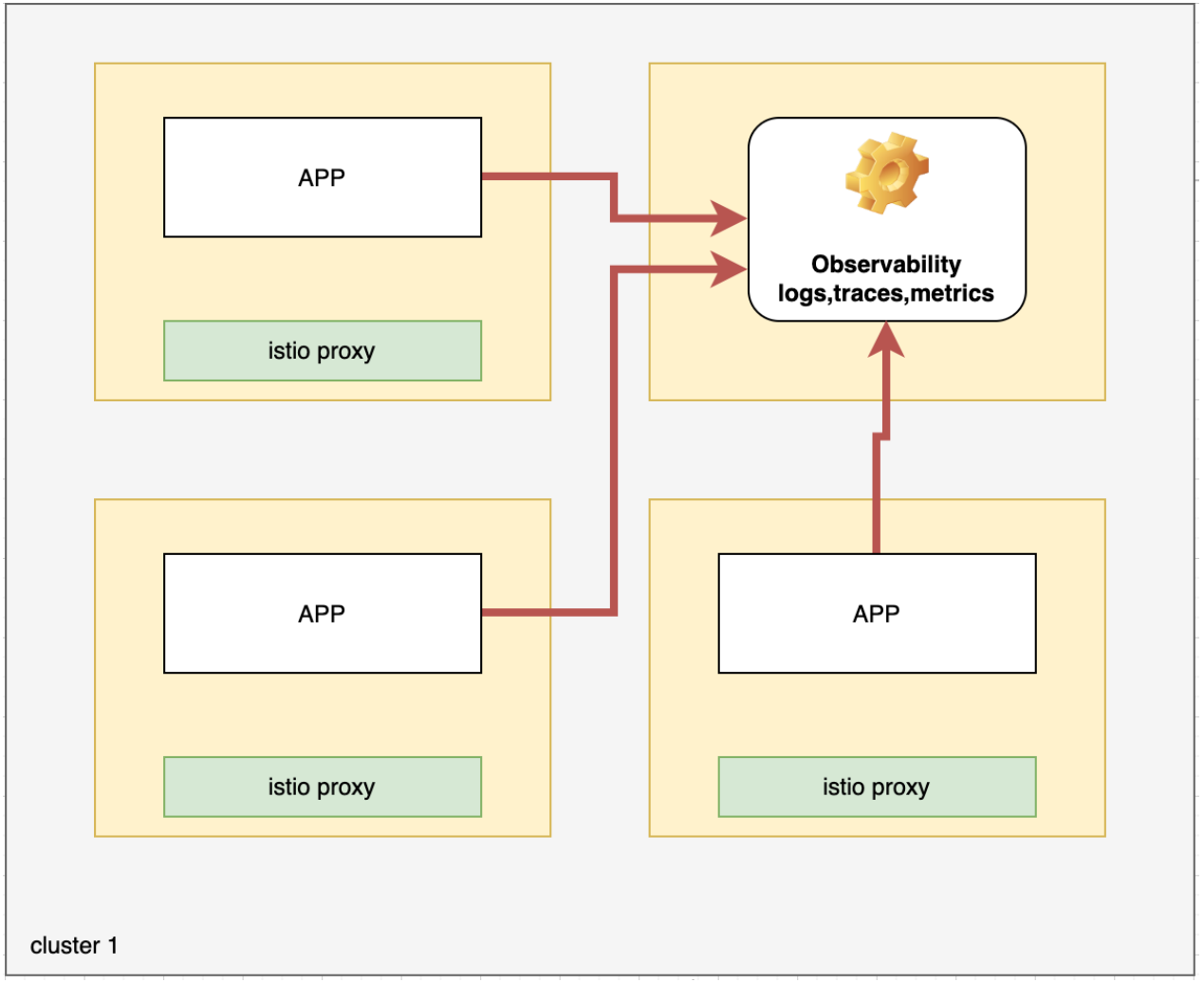
Подключение к системам observability



Кейс – logging sidecar

Подключение к системам observability

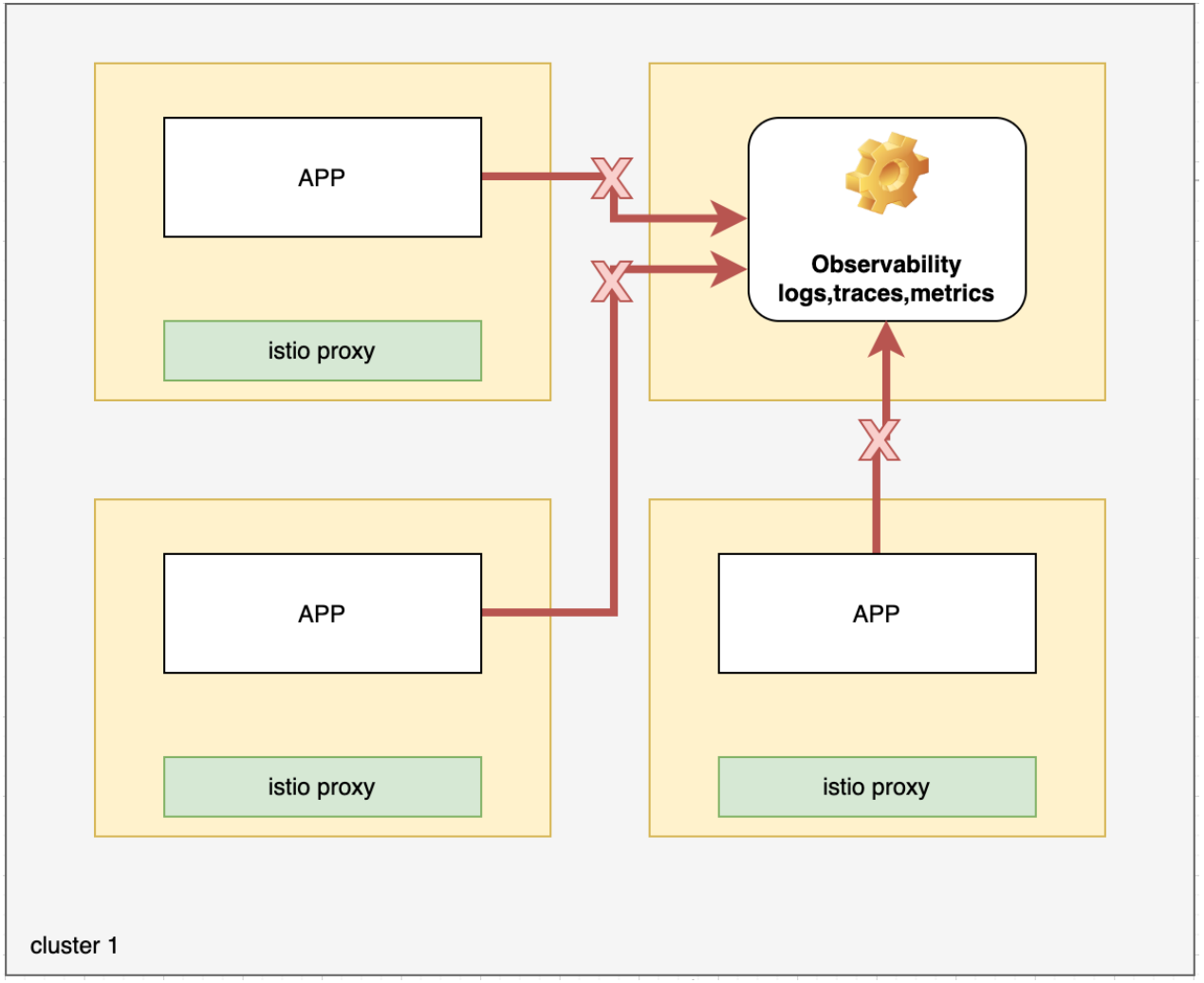
- Приложений много



Кейс – logging sidecar

Подключение к системам observability

- Приложений много
- Сложная конфигурация



Кейс – logging sidecar

Подключение к системам observability

Пример Template для конфигурации сайдкара:

```
kind: Template
metadata:
  name: logger-sc-cm
spec:
  apiVersion: v1
  kind: ConfigMap
  data: |-
    metadata:
      name: logger-sc-cm
    data:
      loggerLevel: INFO
      path: /var/log
```

Кейс – logging sidecar

Подключение к системам observability

Пример Template для мутации пода:

```
kind: Template
metadata:
  name: logger-sc-mount
spec:
 data: |-
   spec:
     volumes:
      - name: kblt-logs-agent
        configMap:
          name: logger-sc-cm
          defaultMode: 420
     containers:
      - name: kblt-logs-agent
        image: logger-image:0.0.1
     resources:
       …
     volumeMounts:
      - name: kblt-logs-agent
        mountPath: /var/logs-agent
        readOnly: true
```

Кейс – logging sidecar

Подключение к системам observability

Пример Trigger для создания конфигураций логгера:

```yaml
kind: Trigger
metadata:
  name: create-logger-configs
spec:
  creationConfigs:
    - annotationNamespace: kblt.logger
      annotationTrigger: cfg
      name: create-logger-configs
      templateRefs:
        - ns/logger-sc-cm
```

Кейс – logging sidecar

Подключение к системам observability

Пример Trigger для мутации пода:

```yaml
kind: Trigger
metadata:
  name: add-logger
spec:
  mutationConfigs:
    - match:
       …selector
      operations:
        - CREATE
        - UPDATE
      scope: Namespaced
      name: add-logger
      templateRefs:
        - ns/logger-sc-mount
      updateStrategy: merge
```

Кейс – logging sidecar

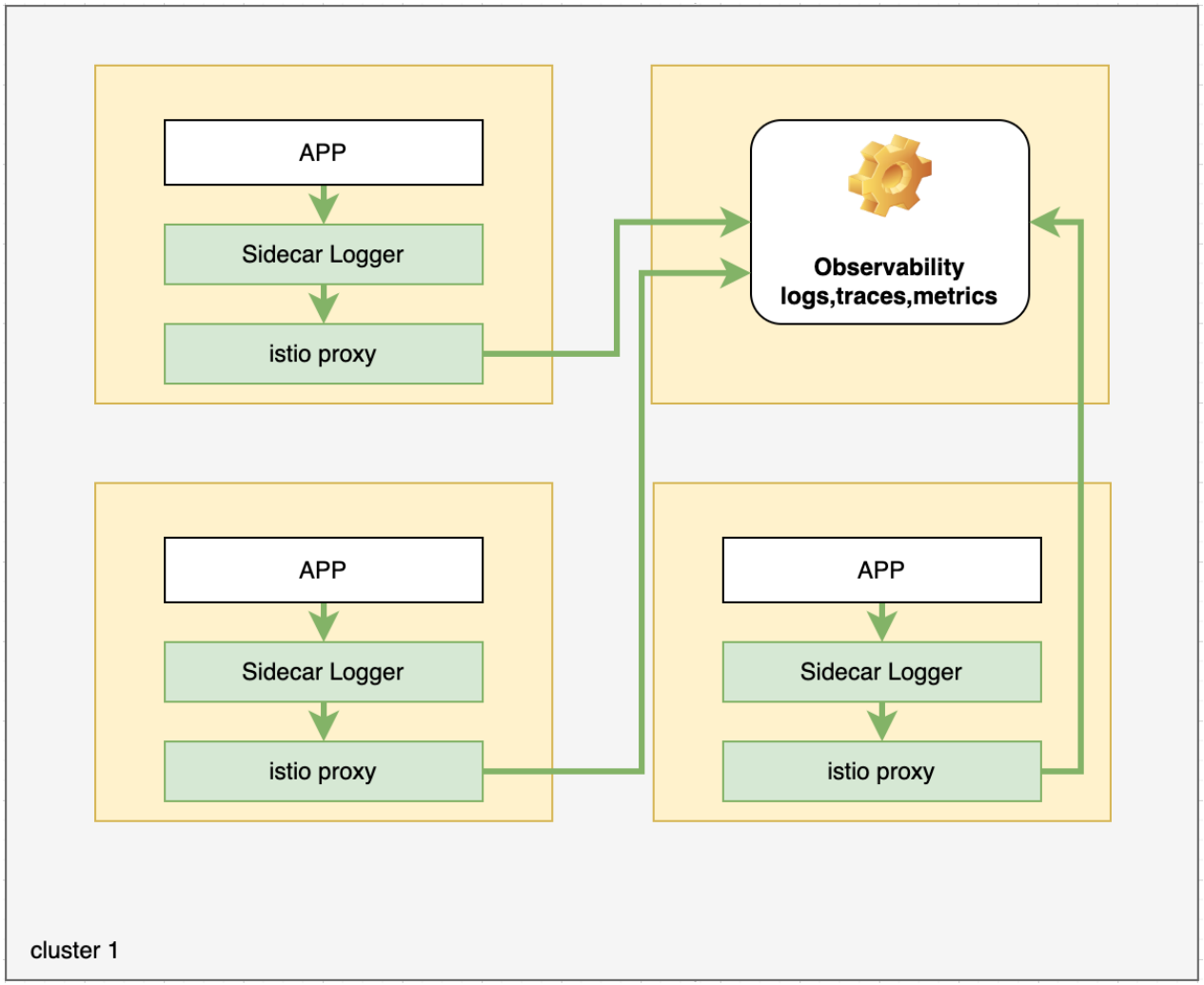
Подключение к системам observability с латте

Триггер

- Мутация Pod
- Создание ConfigMap

Шаблон

- Секция контейнера логгера
- Подключение ENV
- Подключение директорий
- Структура ConfigMap



Кейс – logging sidecar

Подключение к системам observability

Без Kubelatte

- Все команды должны знать топологию
- Ручная синхронизация точек подключения
- Ручное подключение систем мониторинга

Кейс – logging sidecar

Подключение к системам observability

Без Kubelatte

- Все команды должны знать топологию
- Ручная синхронизация точек подключения
- Ручное подключение систем мониторинга

С Kubelatte

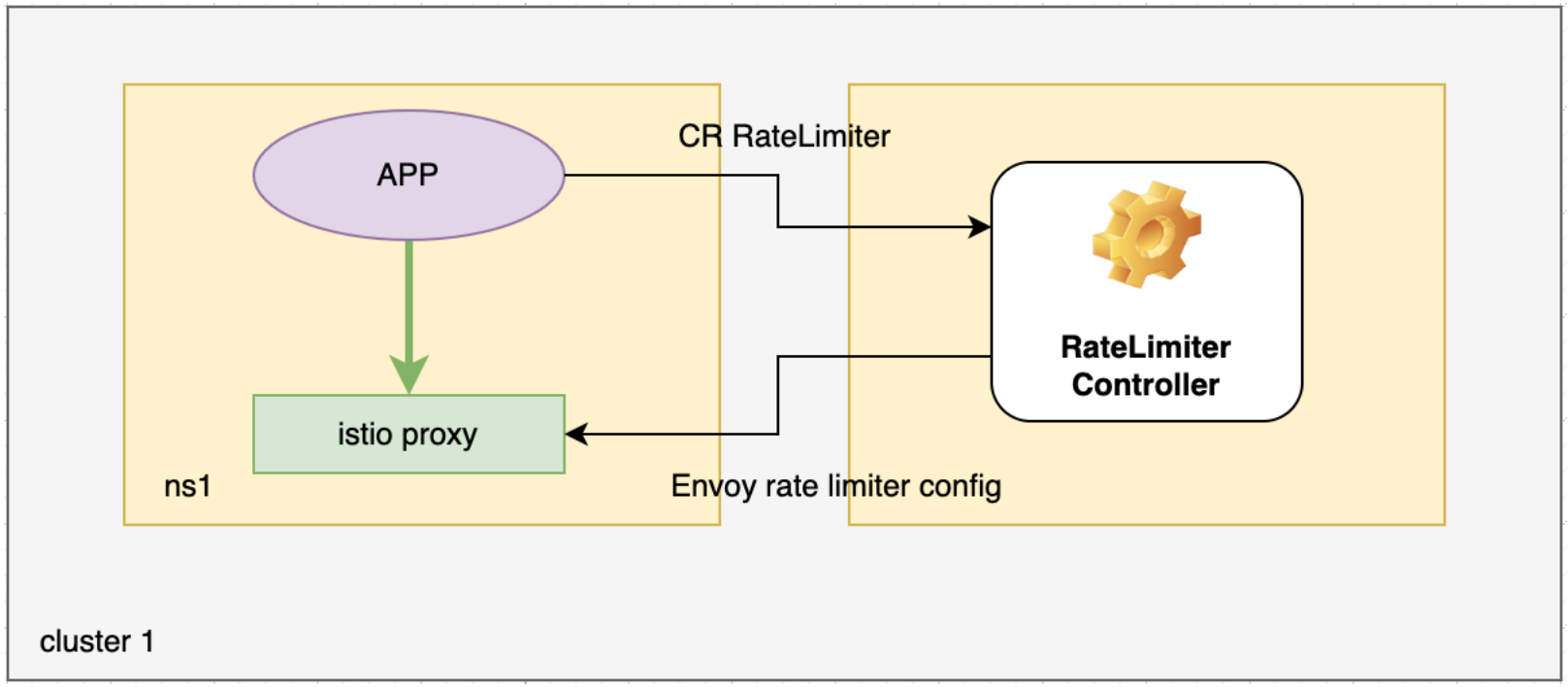
- Автоматическое подключение приложений к мониторингу
- Синхронизация конфигураций подключения
- Единообразная настройка передачи логов

Платформенные расширения

- ✓Service Mesh
- ✓Secret Vault
- ✓Audit & Logging
- ➢**Any k8s-native platform tool**

Кейс – работа с операторами k8s

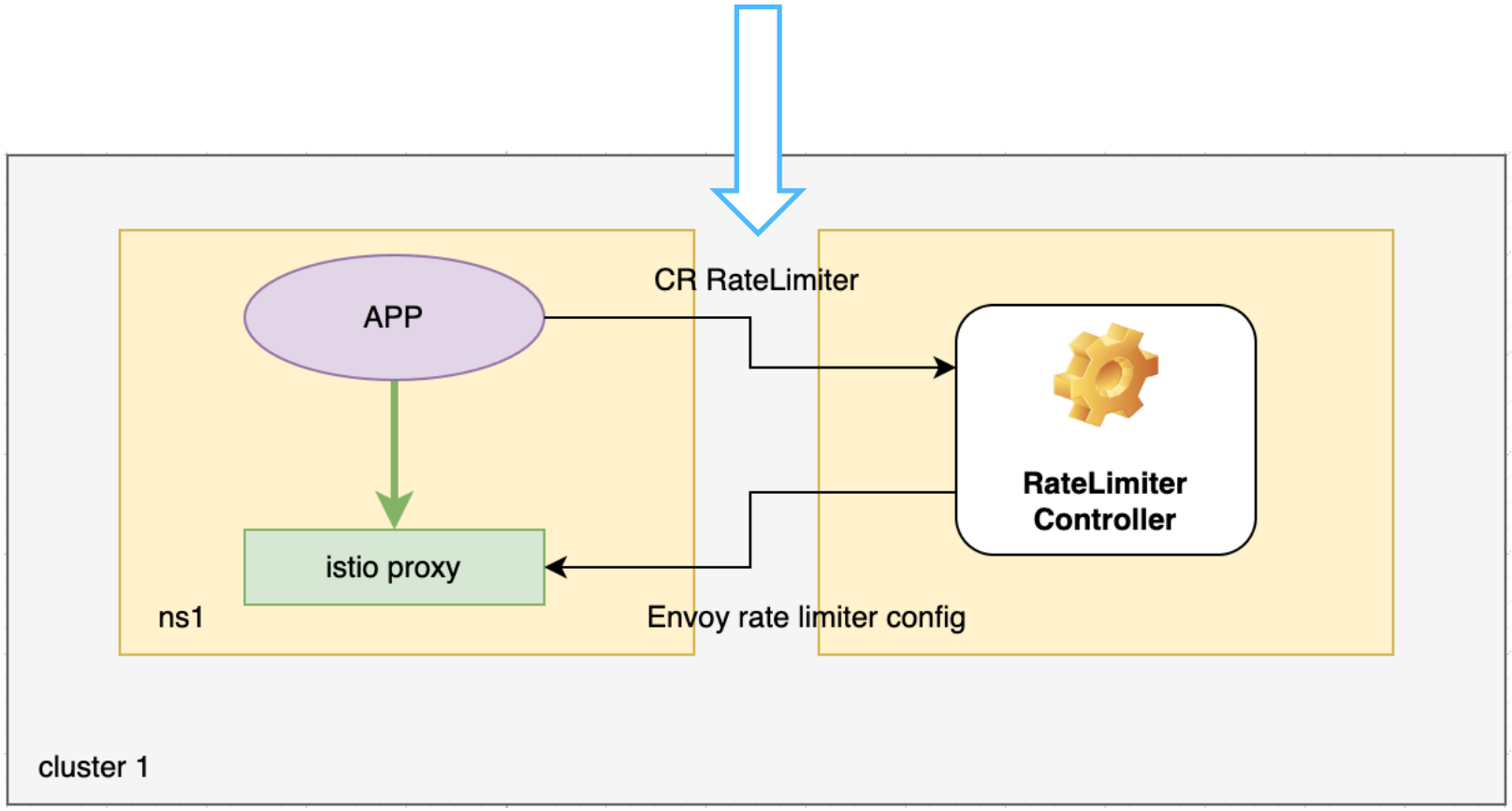
Настройка кастомного RateLimiter



Кейс – работа с операторами k8s

Настройка кастомного RateLimiter

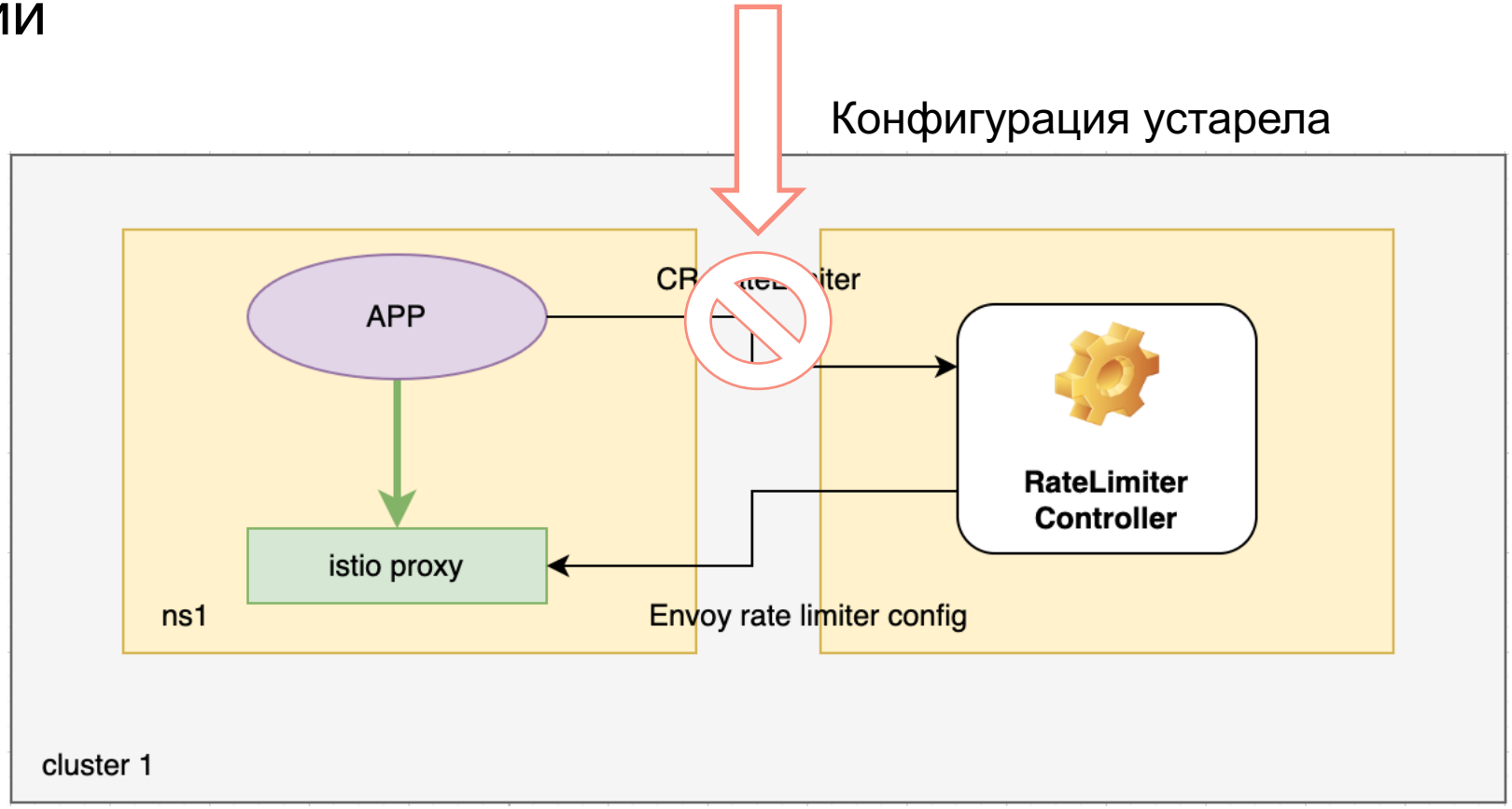
- api внешнего приложения



Кейс – работа с операторами k8s

Настройка кастомного RateLimiter

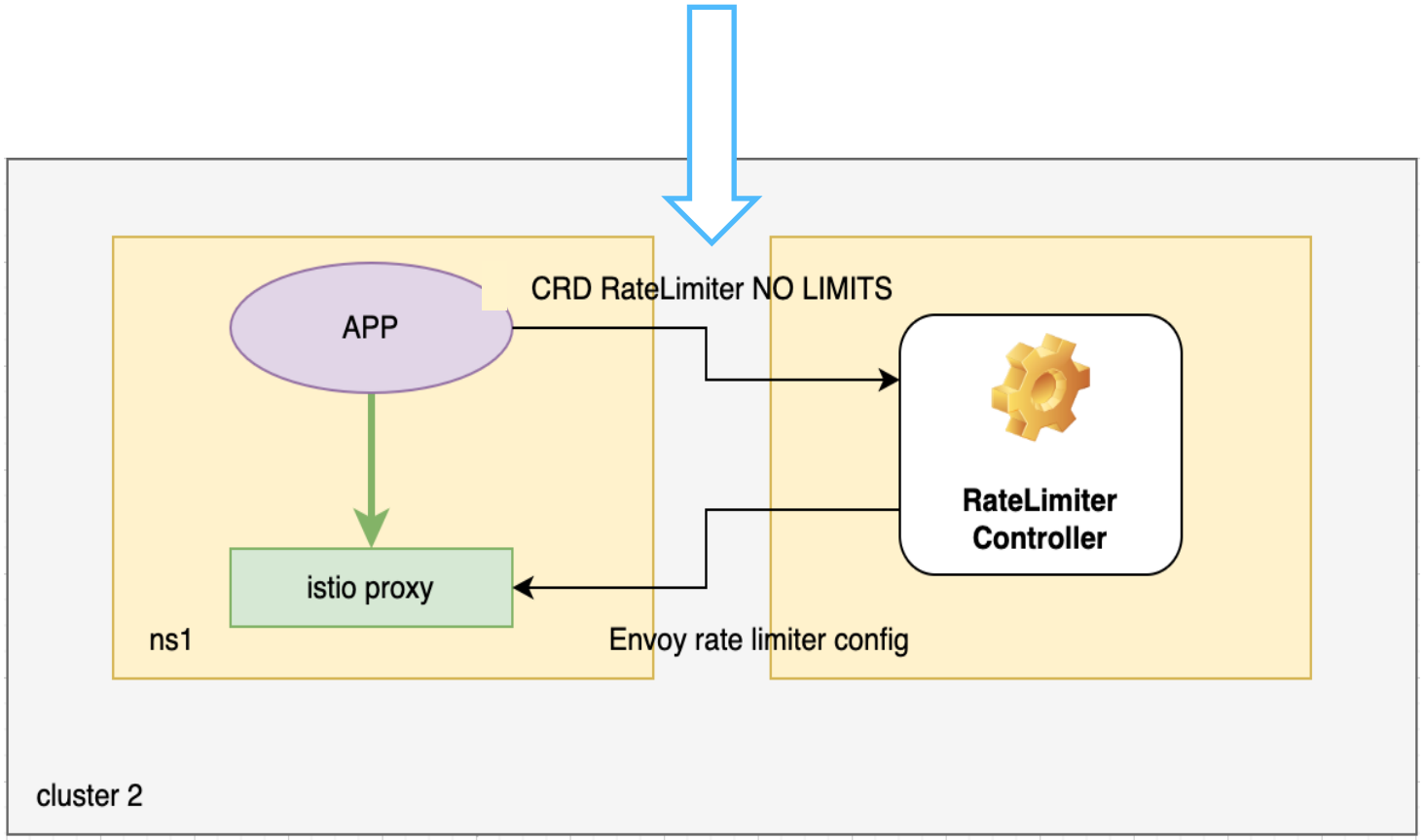
- api внешнего приложения
- актуальность версий



Кейс – работа с операторами k8s

Настройка кастомного RateLimiter

- api внешнего приложения
- актуальность версий
- динамичные значения



Кейс – работа с операторами k8s

Настройка кастомного RateLimiter

Пример Template для создания CustomResource:

```
kind: Template
metadata:
  name: ratelimit-cr
spec:
  apiVersion: dev.my.rate/v1
  kind: RateLimiter
  data: |-
    metadata:
      name: ratelimit-cr-myname
    spec:
      hosts:
        - name: my.example.com
          limit: 1000
          timeout: 10s
```

Кейс – работа с операторами k8s

Настройка кастомного RateLimiter

Пример Trigger для создания CustomResource:

```yaml
kind: Trigger
metadata:
  name: create-ratelimit-cr
spec:
  creationConfigs:
    - annotationNamespace: kblt.rate
      annotationTrigger: limiter
      name: create-ratelimit-cr
      templateRefs:
        - ns/ratelimit-cr
```

Кейс – работа с операторами k8s

Настройка кастомного RateLimiter с латте

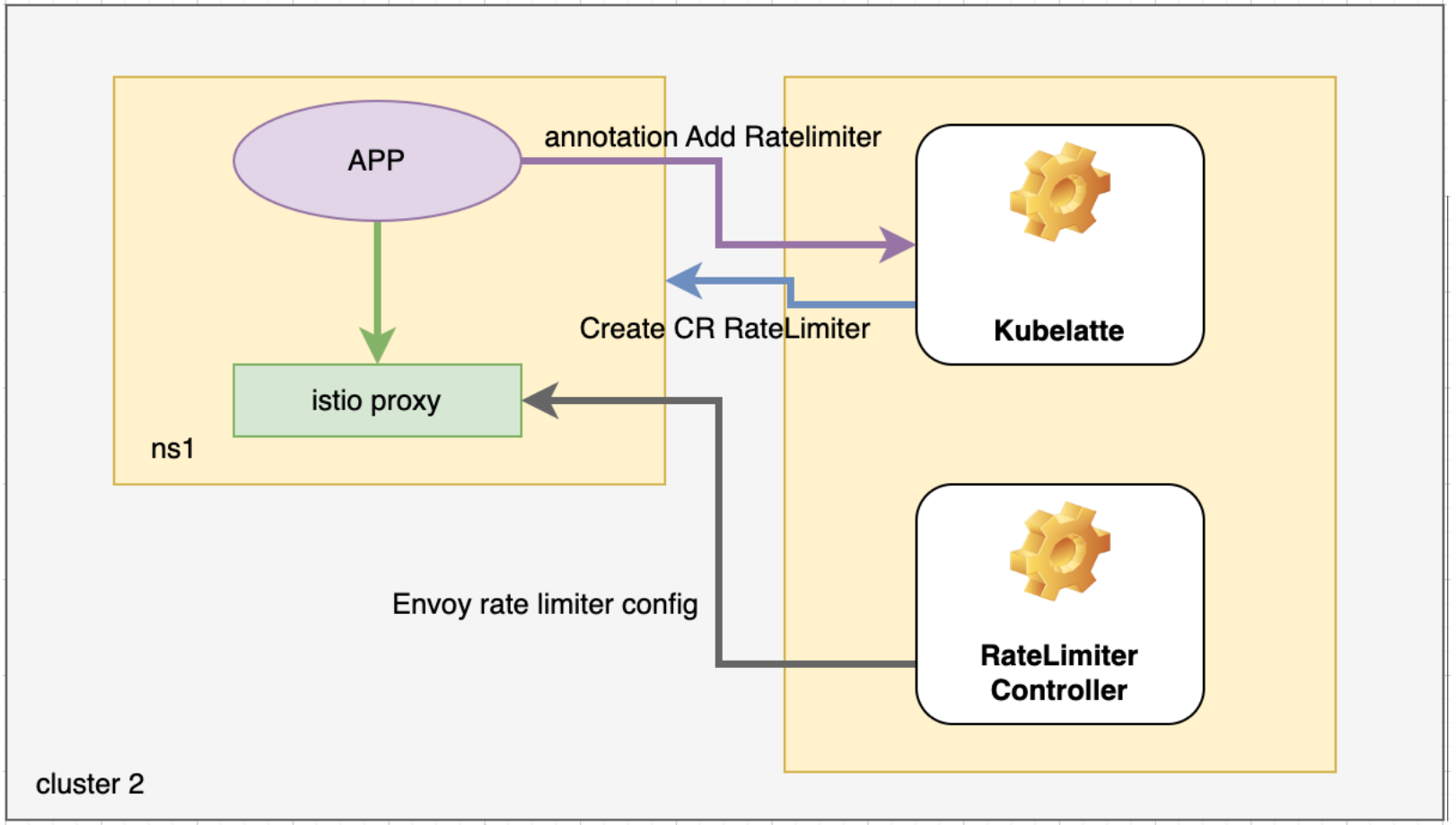
Триггер

- SideEffect

создание CR RateLimiter

Шаблон

- Структура RateLimiter



Кейс – работа с операторами k8s

Настройка кастомного RateLimiter

Без Kubelatte

- Экпертиза в конфигурации смежной системы
- Ручной контроль актуальности API

Кейс – работа с операторами k8s

Настройка кастомного RateLimiter

Без Kubelatte

- Экпертиза в конфигурации смежной системы
- Ручной контроль актуальности API

С Kubelatte

- Сокрытие API за аннотацией
- Всегда валидное и актуальное использование API

Политики Kubelatte

Кейс использования	Комментарий
контроль запуска подов	Контроль спецификации пода и запрещение использования привилегированного запуска пода
дефолтные network policies	Автоматическая настройка сетевых политик при создании проекта в кластере
использование HostNetwork	Запрет использования хостовой сети ноды для пересечения перехвата трафика приложением
монтирование секретов в Deployment	Контроль использования секретов и защита от компрометированния данных
дефолтные маршруты Istio	Сокрытие низкоуровнего API Istio и упрощение получения базовых маршрутов к сервисам
подключение vault	Подключение безопасного хранилища секретов по запросу, автоматическая мутация пода
logging sidecar	Создание конфигураций сайдкара логгера и автоматическое подключение к поду
работа с операторами k8s	Упрощение использования API смежных систем

Ищите новые идеи для Policy Management, а также делитесь своими сценариями на GitVerse!


Итоги

Итоги

- Governance для Kubernetes – критически важный процесс и должен быть реализован в инфраструктуре организации;
- Процесс должен быть автоматизирован и не требовать вовлечения команд;
- Policy Engine хорошо подходит для задач governance и доступен в OpenSource;