2 ИЮНЯ 2026, МОСКВА, ЛОФТ ГОЭЛРО

БЕКОН LUNTRY '26

ЕДИНСТВЕННАЯ КОНФЕРЕНЦИЯ ПО БЕЗОПАСНОСТИ
КОНТЕЙНЕРОВ И КОНТЕЙНЕРНЫХ СРЕД

GitOps: мозг для политик безопасности

Андрей Орехов | Swordfish Security

БЕКОН'26

КОНФЕРЕНЦИЯ ПО БЕЗОПАСНОСТИ КОНТЕЙНЕРОВ И КОНТЕЙНЕРНЫХ СРЕД

GitOps: мозг для политик безопасности

Андрей Орехов

Руководитель группы безопасности контейнеризации | Swordfish Security

О себе

Раньше

Kaspersky, BI.ZONE, PT – DFIR Specialist

Astra – DevOps Engineer

Сейчас

Руководитель группы безопасности контейнеризации

Swordfish Security

Кому полезен доклад

Платформенным инженерам

которые управляют флотом из множества кластеров и хотят единые правила вместо ручной настройки каждого

DevSecOps-командам

которые внедряют политики безопасности и хотят централизованный контроль с аудит-trail через Git

БЕКОН'26

ПРОБЛЕМА

00

Жизнь без централизации

У каждой команды — свои политики. Или их нет вообще.

Жизнь без централизации

У каждой команды — свои политики. Или их нет вообще.

Новый кластер

Кто-то вручную ставит Gatekeeper, кто-то забывает. Один и тот же контур настроен по-разному.

Расхождение конфигураций

Один кластер защищён, другой — нет. Узнаём об этом случайно.

Жизнь без централизации

У каждой команды — свои политики. Или их нет вообще.

Новый кластер

Кто-то вручную ставит Gatekeeper, кто-то забывает. Один и тот же контур настроен по-разному.

Аудит — раз в квартал

«Разберёмся перед проверкой». Между проверками — слепая зона.

Расхождение конфигураций

Один кластер защищён, другой — нет. Узнаём об этом случайно.

Compliance-fails

Аудитор приходит — и каждый раз заново собираем доказательства.

Жизнь без централизации

У каждой команды — свои политики. Или их нет вообще.

Разные среды, разные исключения, разные версии политик — формируется зоопарк.

Новый кластер

Кто-то вручную ставит Gatekeeper, кто-то забывает. Один и тот же контур настроен по-разному.

Аудит — раз в квартал

«Разберёмся перед проверкой». Между проверками — слепая зона.

Нет единой картины

Что реально применено прямо сейчас? На каких кластерах? В какой версии?

Расхождение конфигураций

Один кластер защищён, другой — нет. Узнаём об этом случайно.

Compliance-fails

Аудитор приходит — и каждый раз заново собираем доказательства.

Инциденты в проде

Привилегированный pod, hostPath, образ из неизвестного registry — всё это должны были отсечь политики.

Compliance — это не событие. Это состояние.

БЕКОН'26

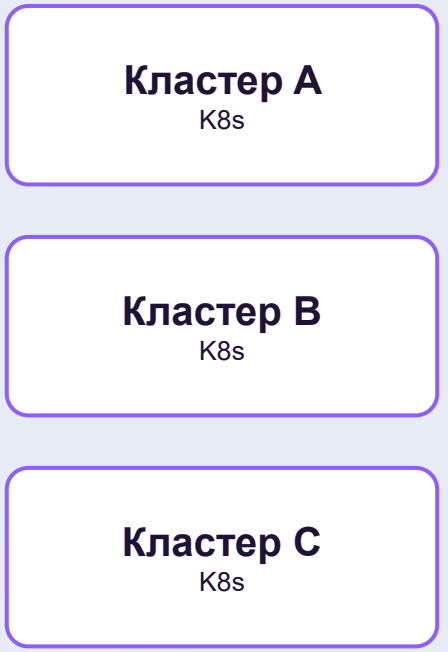
АРХИТЕКТУРА РЕШЕНИЯ

01

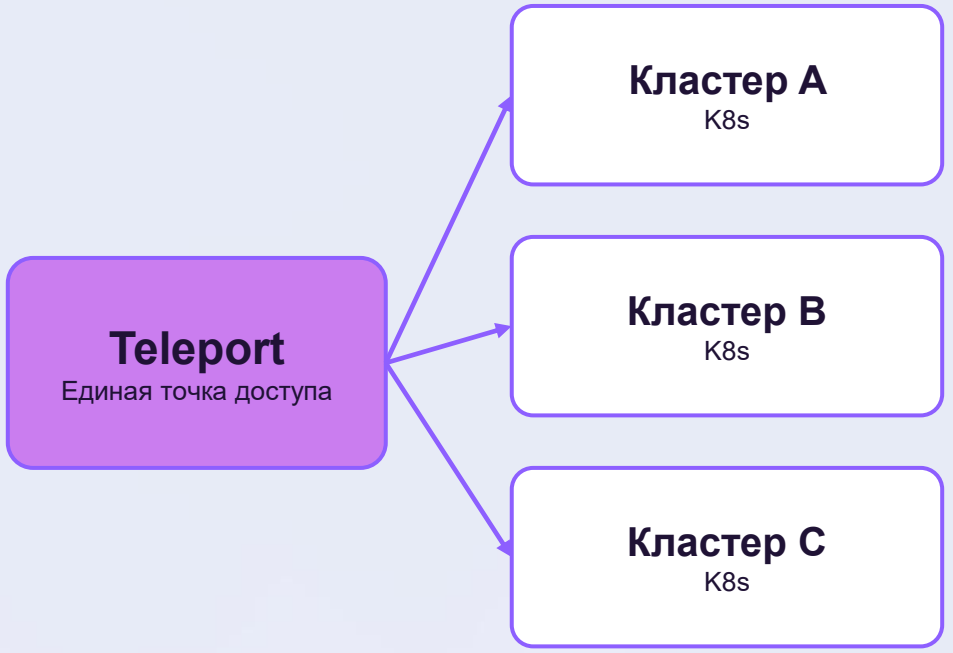
Стек: с чего начинаем

Есть кластеры.
В каждом — Gatekeeper.
Нужно доставлять политики
централизованно и автоматически.

★ **Gatekeeper — наш выбор. У вас может быть Kyverno или VAP.**



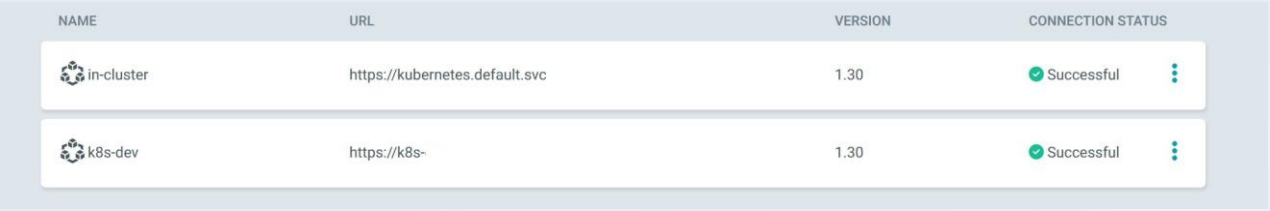
Стек: Teleport — единая точка доступа

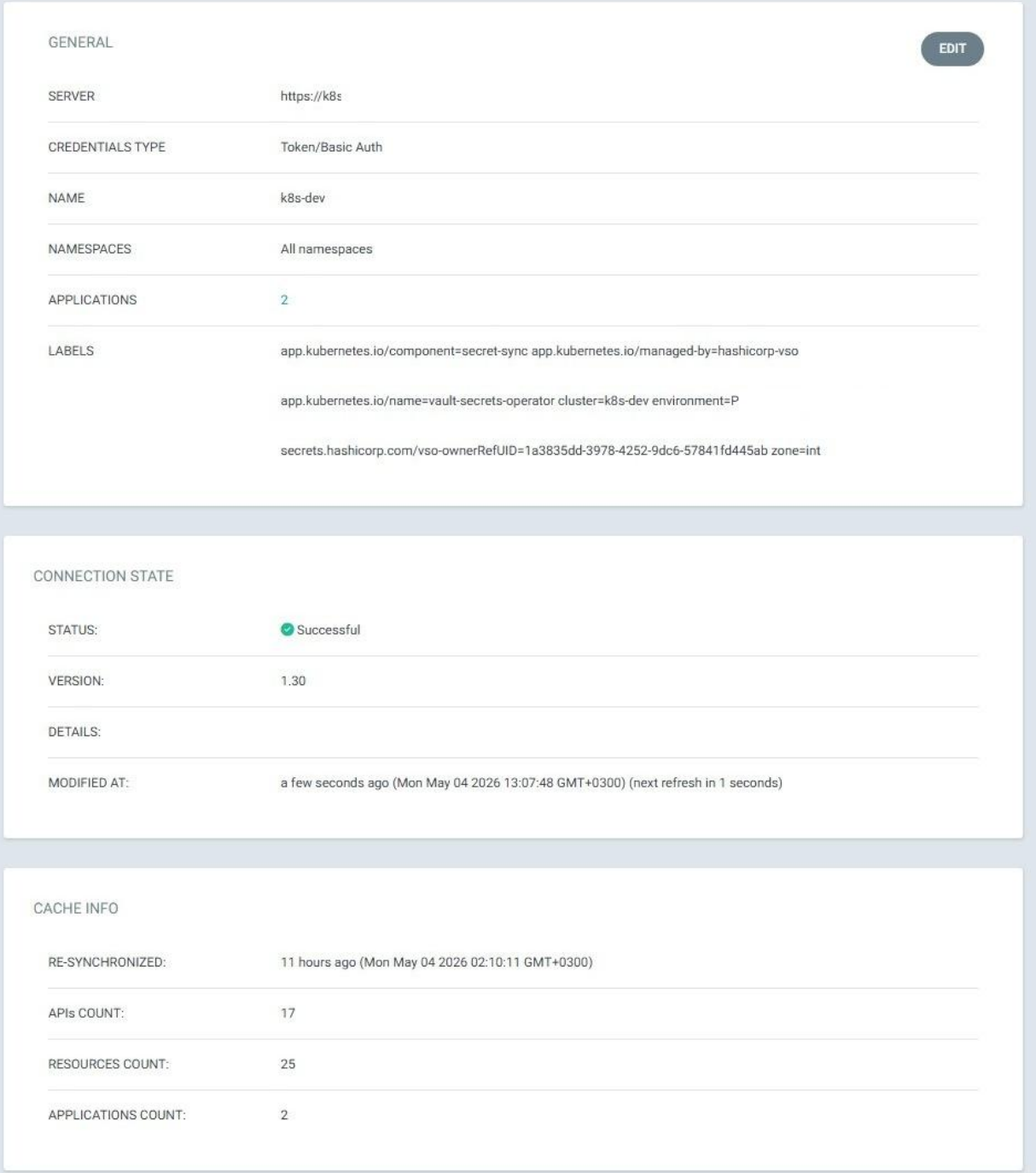


Единая точка доступа: Teleport / IAM

- **Teleport подключает внешние кластеры через execProviderConfig**
- tbot получает credentials и кладёт kubeconfig — ArgoCD использует его
- Под капотом argo-cd использует наш скрипт дважды — 1-ый раз в argocd-server (для непосредственной доставки конфигураций) и 2-ой раз в argocd-application-controller, если мы из ui пытаемся посмотреть/изменить и тд объекты

* ArgoCD 1.8.0

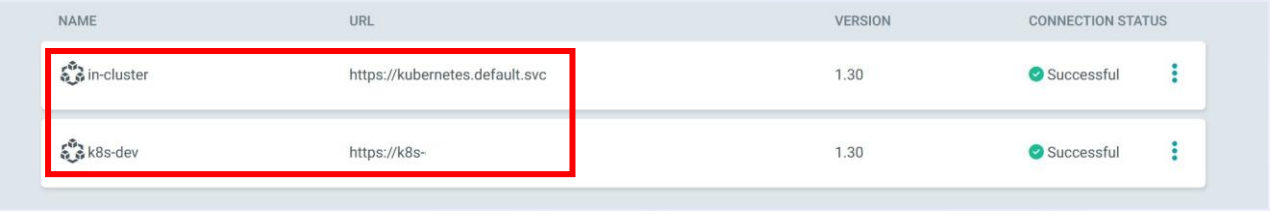


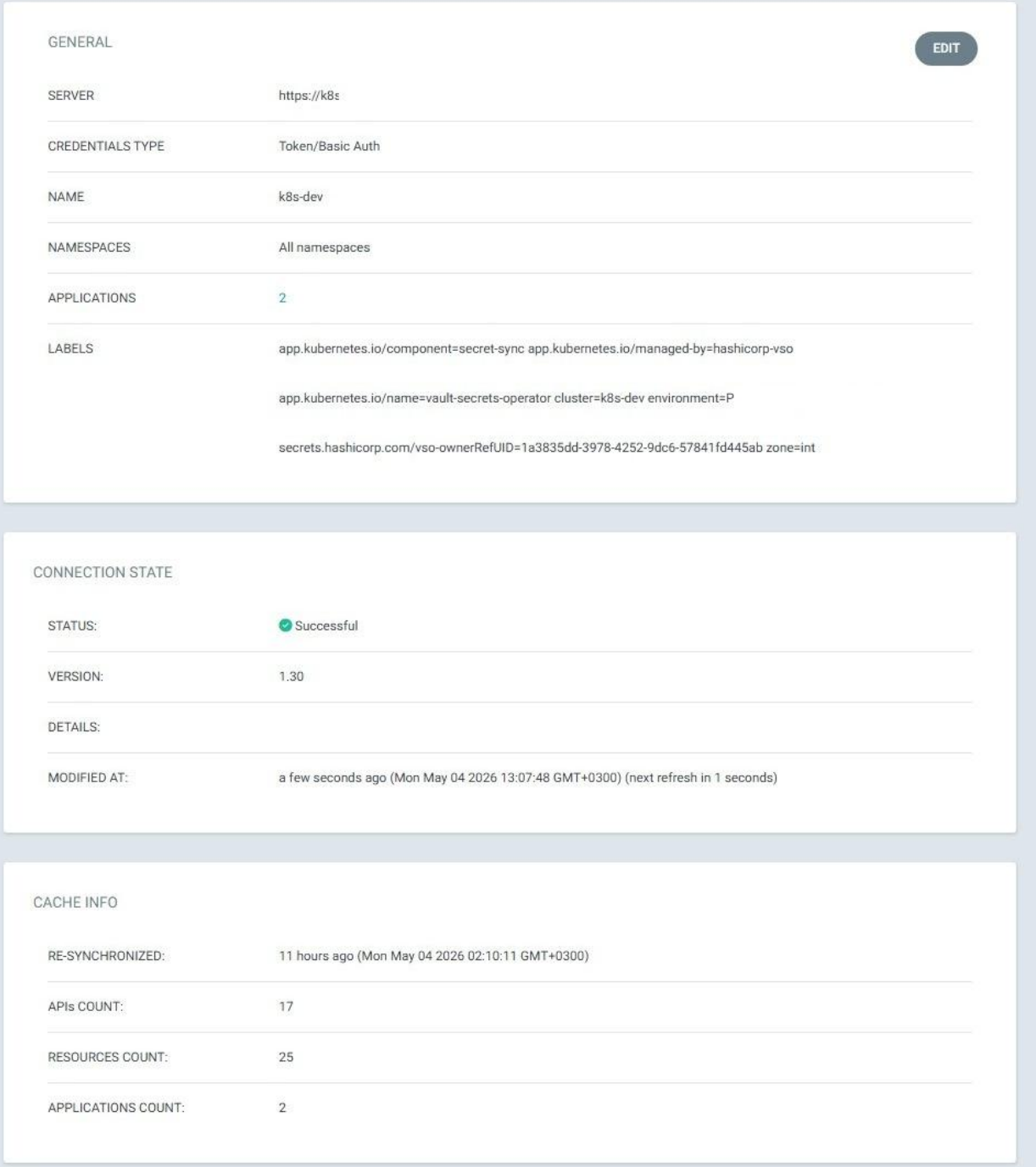


Единая точка доступа: Teleport / IAM

- **Teleport подключает внешние кластеры через execProviderConfig**
- tbot получает credentials и кладёт kubeconfig — ArgoCD использует его
- Под капотом argo-cd использует наш скрипт дважды — 1-ый раз в argocd-server (для непосредственной доставки конфигураций) и 2-ой раз в argocd-application-controller, если мы из ui пытаемся посмотреть/изменить и тд объекты

* ArgoCD 1.8.0

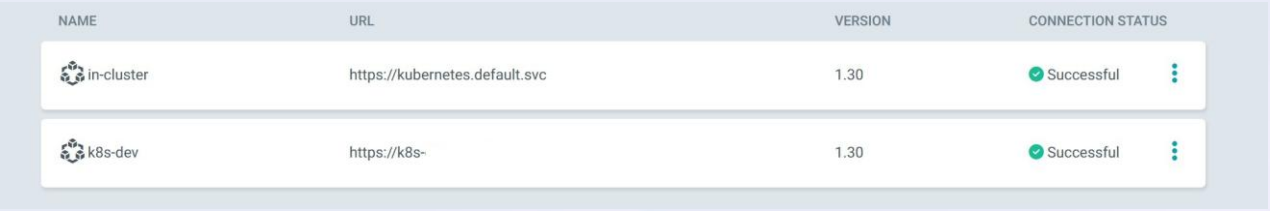


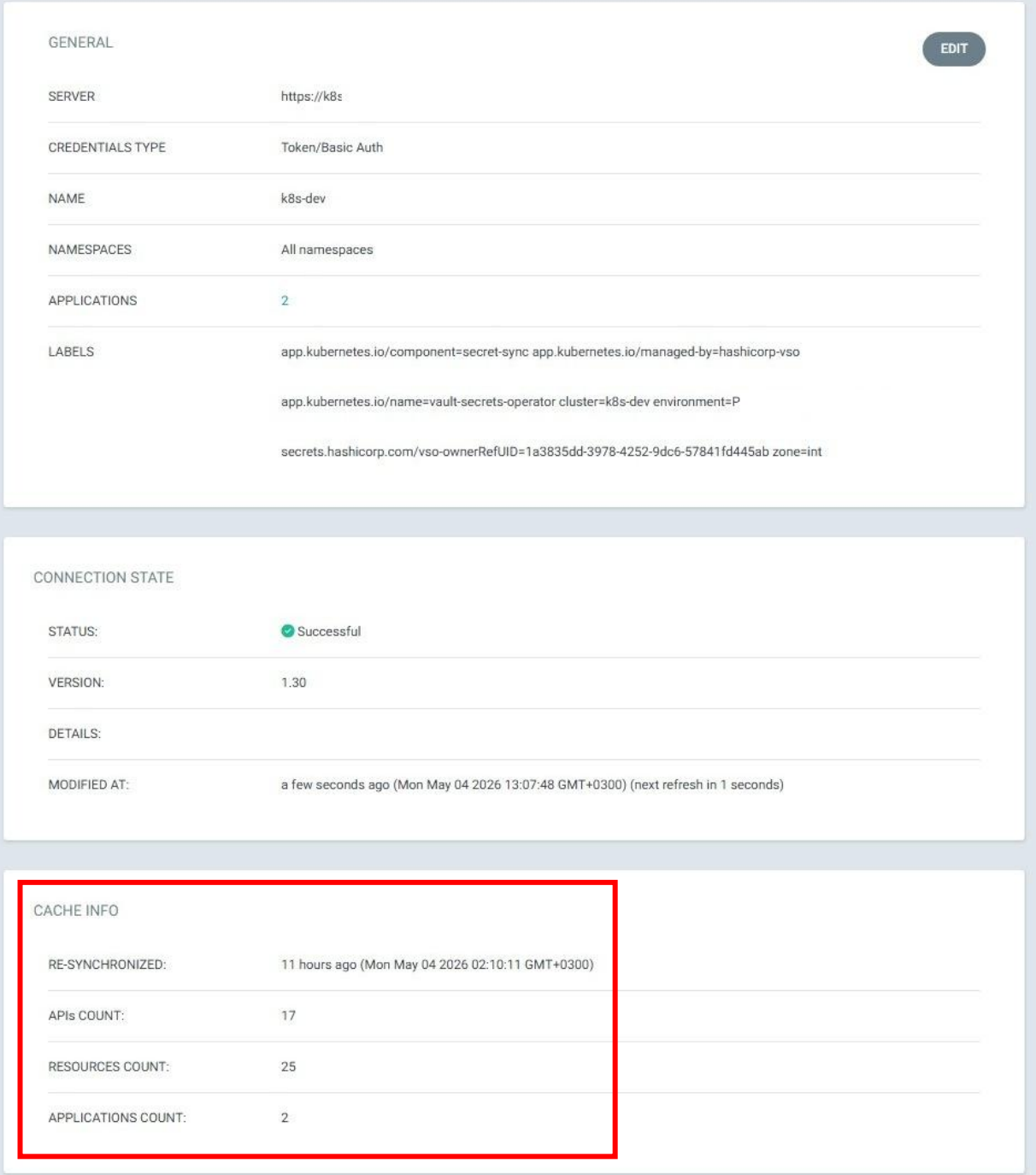


Единая точка доступа: Teleport / IAM

- **Teleport подключает внешние кластеры через execProviderConfig**
- tbot получает credentials и кладёт kubeconfig — ArgoCD использует его
- Под капотом argo-cd использует наш скрипт дважды — 1-ый раз в argocd-server (для непосредственной доставки конфигураций) и 2-ой раз в argocd-application-controller, если мы из ui пытаемся посмотреть/изменить и тд объекты

* ArgoCD 1.8.0





Единая точка доступа: Teleport / IAM


```
{
  "execProviderConfig": {
    "command": "/tmp/script.sh",
    "args": ['                                    ", "k8s-
                          ", "k8s-dev"],
    "apiVersion": "client.authentication.k8s.io/v1beta1",
    "installHint": "SFS"
  },
  "tlsClientConfig": {
    "serverName": "kube-teleport-proxy
                          ,
    "insecure": false,
    "caData":
"LS0tLS1CRUdJTiBDRVJUSUZJQ0FURS0tLS0tCk1JSUV0VENDQTUyZ0F3SUJBZ0lSQVBTdU
```

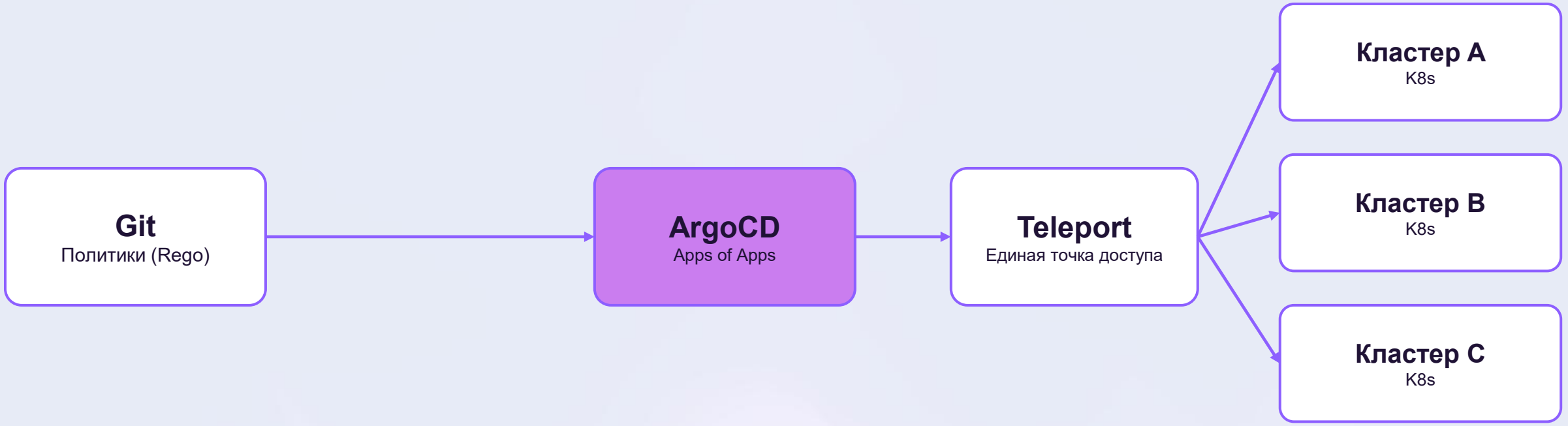
Единая точка доступа: Teleport / IAM

```
#!/usr/bin/bash

# export TBOT_TOKEN="$1"
# export AUTH_SERVER="$2" # teleport.endpoint:443
# export CLUSTER_NAME="$3" # k8s-cluster-name

mkdir -p "/tmp/$3"

cat <<EOF >/tmp/$3/tbot.yaml
auth_server: $2
oneshot: true
onboarding:
  join_method: token
storage:
  memory: true
destinations:
  - directory: /tmp/$3
    kubernetes_cluster: $3
EOF

echo "Preparing tbot start" >/tmp/$3/log.start.txt

/tmp/binaries/tbot start -c /tmp/$3/tbot.yaml --token=${1} 2>>/tmp/$3/log.start.txt

if grep -q "Successfully renewed impersonated certificates" "/tmp/$3/log.start.txt"; then
    echo "Preparing kube credentials" >/tmp/$3/log.txt
    /tmp/binaries/tbot kube credentials --destination-dir="/tmp/$3" 2>>/tmp/$3/log.txt
else
    echo '{"kind":"ExecCredential","apiVersion":"client.authentication.k8s.io/v1beta1","spec":{"interactive":false},"status":false}'
fi
```

Единая точка доступа: Teleport / IAM

```
#!/usr/bin/bash

# export TBOT_TOKEN="$1"
# export AUTH_SERVER="$2" # teleport.endpoint:443
# export CLUSTER_NAME="$3" # k8s-cluster-name

mkdir -p "/tmp/$3"

cat <<EOF >/tmp/$3/tbot.yaml
auth_server: $2
oneshot: true
onboarding:
  join_method: token
storage:
  memory: true
destinations:
  - directory: /tmp/$3
    kubernetes_cluster: $3
EOF

echo "Preparing tbot start" >/tmp/$3/log.start.txt

/tmp/binaries/tbot start -c /tmp/$3/tbot.yaml --token=${1} 2>>/tmp/$3/log.start.txt

if grep -q "Successfully renewed impersonated certificates" "/tmp/$3/log.start.txt"; then
    echo "Preparing kube credentials" >/tmp/$3/log.txt
    /tmp/binaries/tbot kube credentials --destination-dir="/tmp/$3" 2>>/tmp/$3/log.txt
else
    echo '{"kind":"ExecCredential","apiVersion":"client.authentication.k8s.io/v1beta1","spec":{"interactive":false},"status":false}'
fi
```

Стек: ArgoCD — доставка из Git



ArgoCD + Apps of Apps: иерархия зон

БЕКОН'26

Структура репозиториев в Git

- wide/
  └— политики для ВСЕХ кластеров
- zone-prod/
  └— только prod-кластеры
- zone-test/
  └— только test-кластеры
- cluster-name/
  └— конкретный кластер

Как это работает

- Лейбл на кластере → ArgoCD ApplicationSet выбирает нужный репозиторий
- Apps of Apps: один Application управляет деревом дочерних Applications
- Monorepo для политик безопасности — единая точка правды

ArgoCD + Apps of Apps: иерархия зон

БЕКОН'26

[redacted] / AC-POLICIES-CLUSTER-WIDE

Политики для отдельных оркестраторов контейнеров

Source

master ··· | AC-POLICIES-CLUSTER-WIDE / policies / gatekeeper / [redacted] / **constraints** /

Source

..

k8sblockdefaultnamespace.yml

k8sBlockEmptySecurityContext.yml

K8sCsrBlockEmptySecurityContext.yml

[redacted] / AC-POLICIES-CLUSTER-WIDE

Политики для отдельных оркестраторов контейнеров

Source

master ··· | AC-POLICIES-CLUSTER-WIDE / policies / gatekeeper / [redacted] / **constraintTemplates** /

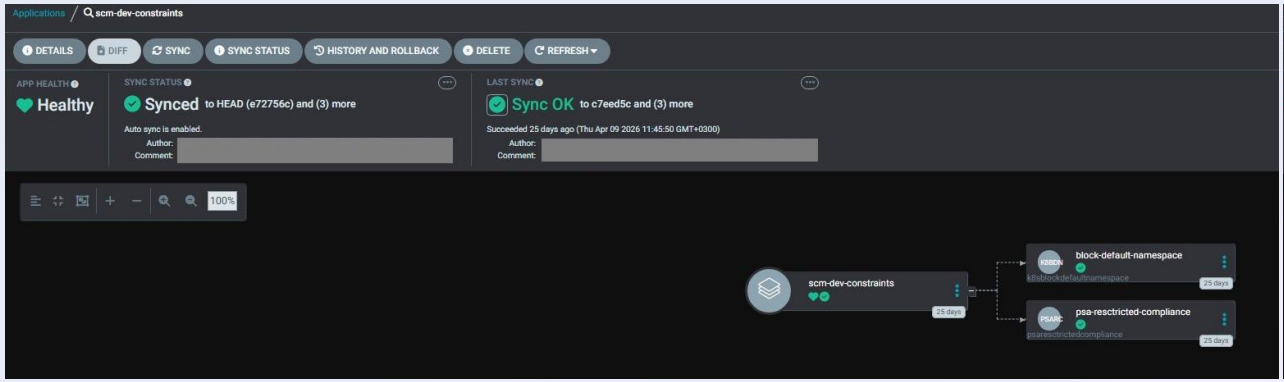
Source

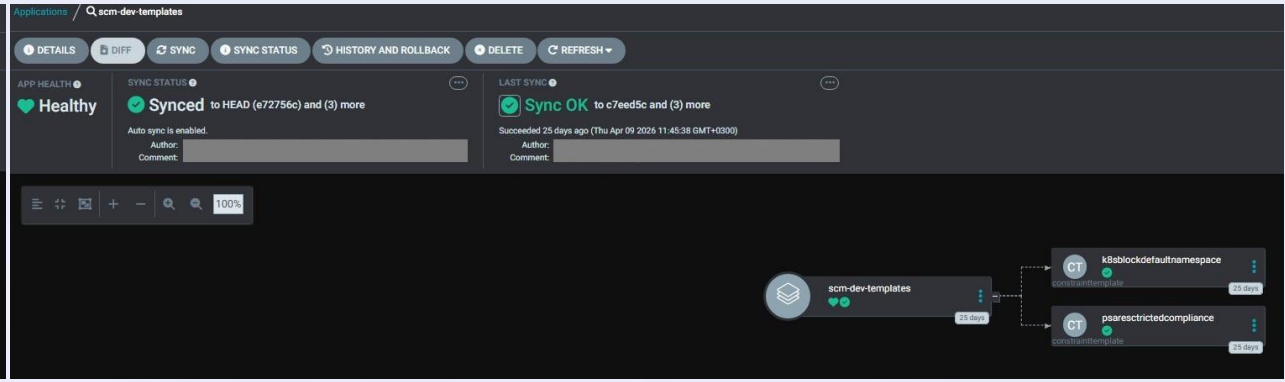
..

k8sblockdefaultnamespace.yml

k8sBlockEmptySecurityContext.yml

K8sCsrBlockEmptySecurityContext.yml





ArgoCD + Apps of Apps: иерархия зон

БЕКОН'26

[redacted] / AC-POLICIES-CLUSTER-WIDE

Политики для отдельных оркестраторов контейнеров

Source

master ··· | AC-POLICIES-CLUSTER-WIDE / policies / gatekeeper / [redacted] / **constraints** /

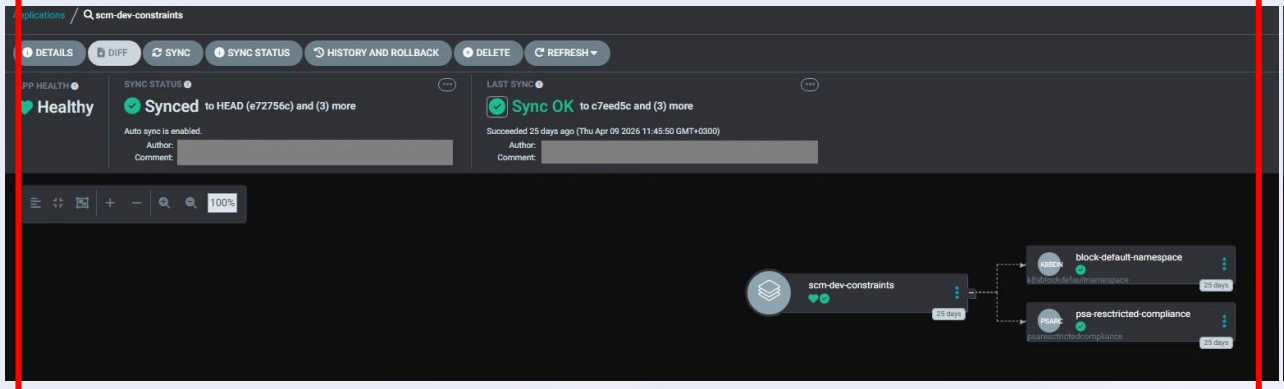
Source

..

k8sblockdefaultnamespace.yml

k8sBlockEmptySecurityContext.yml

K8sCsrBlockEmptySecurityContext.yml



[redacted] / AC-POLICIES-CLUSTER-WIDE

Политики для отдельных оркестраторов контейнеров

Source

master ··· | AC-POLICIES-CLUSTER-WIDE / policies / gatekeeper / [redacted] / **constraintTemplates** /

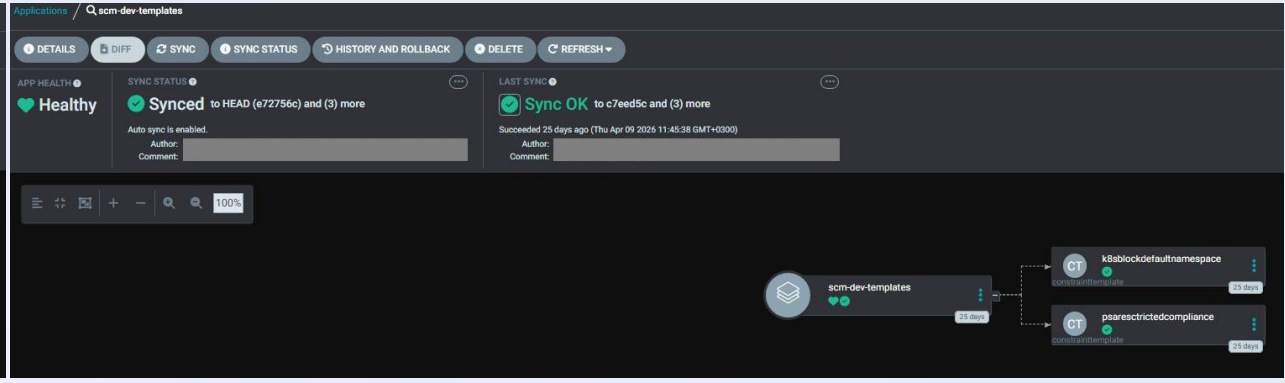
Source

..

k8sblockdefaultnamespace.yml

k8sBlockEmptySecurityContext.yml

K8sCsrBlockEmptySecurityContext.yml



ArgoCD + Apps of Apps: иерархия зон

БЕКОН'26

[redacted] / AC-POLICIES-CLUSTER-WIDE

Политики для отдельных оркестраторов контейнеров

Source

master ··· | AC-POLICIES-CLUSTER-WIDE / policies / gatekeeper / [redacted] / **constraints** /

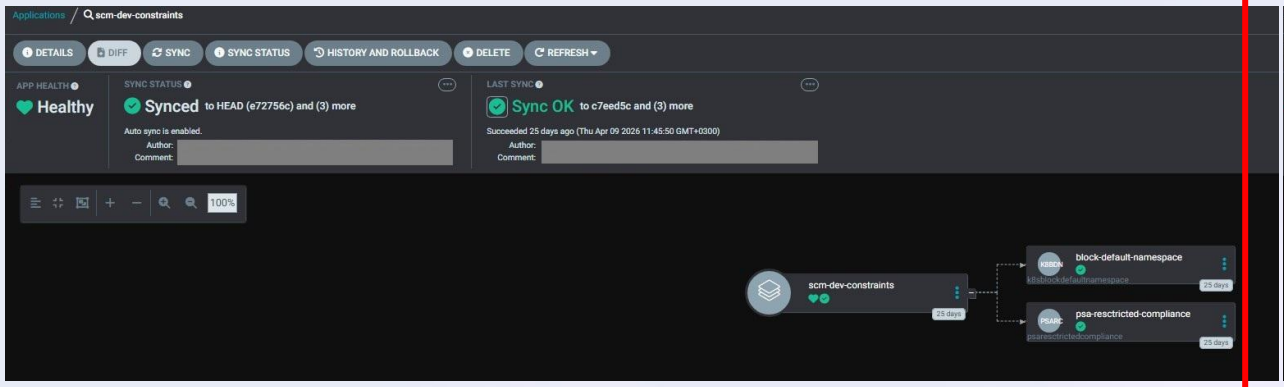
Source

..

k8sblockdefaultnamespace.yml

k8sBlockEmptySecurityContext.yml

K8sCsrBlockEmptySecurityContext.yml



[redacted] / AC-POLICIES-CLUSTER-WIDE

Политики для отдельных оркестраторов контейнеров

Source

master ··· | AC-POLICIES-CLUSTER-WIDE / policies / gatekeeper / [redacted] / **constraintTemplates** /

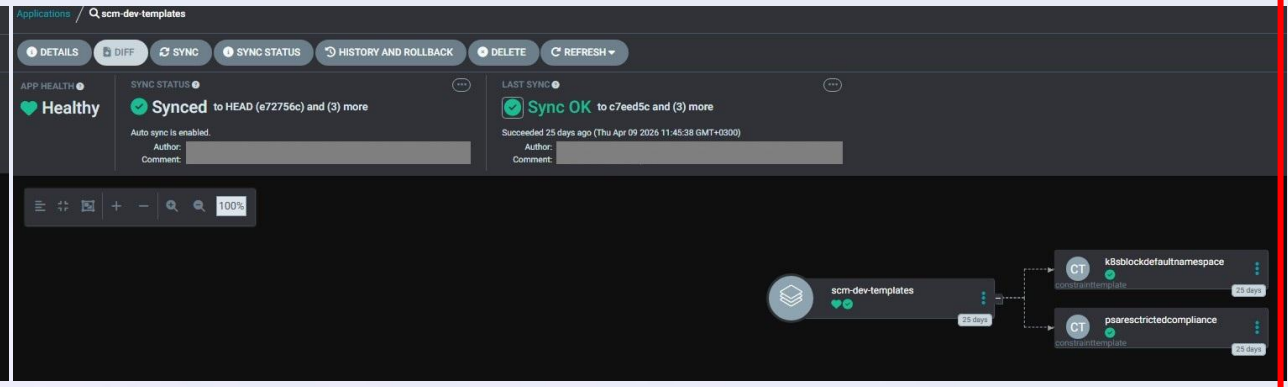
Source

..

k8sblockdefaultnamespace.yml

k8sBlockEmptySecurityContext.yml

K8sCsrBlockEmptySecurityContext.yml



Роли и доступы в ArgoCD

AppProject: ограничивает, в какие кластеры и namespace ArgoCD может деплоить

RBAC внутри ArgoCD: кто может что делать с Applications и AppSets

Проблема по умолчанию: Argo запрашивает права на Deployments, Services и т.д.

В подключённых кластерах Argo отвечает ТОЛЬКО за Constraints и ConstraintTemplates

Exceptions: настройка resource.inclusions — whitelist вместо exclusions

Сложности RBAC в ArgoCD

deny всегда приоритетнее allow — порядок правил в policy.csv не важен, deny срабатывает независимо от более специфичных allow

resource.inclusions — whitelist, не blacklist: если задан хоть один inclusion — всё остальное молча игнорируется ArgoCD, включая стандартные ресурсы

Итог при дефолтном AppProject: без явного clusterResourceWhitelist ArgoCD разрешает деплой кластерных ресурсов — ловушка для тех, кто ожидает deny-by-default

```
resource.inclusions: |
  - apiGroups:
      - "secrets.hashicorp.com"
    kinds:
      - "*"
    clusters:
      - https://kubernetes.default.svc
  - apiGroups:
      - "argoproj.io"
    kinds:
      - "Application"
      - "ApplicationSet"
      - "AppProject"
    clusters:
      - https://kubernetes.default.svc
  - apiGroups:
      - "constraints.gatekeeper.sh"
      - "templates.gatekeeper.sh"
      - "mutations.gatekeeper.sh"
    kinds:
      - "*"
    clusters:
      - https://kubernetes.default.svc
  - apiGroups:
      - "constraints.gatekeeper.sh"
      - "templates.gatekeeper.sh"
      - "mutations.gatekeeper.sh"
    kinds:
      - "*"
    clusters:
      - "*"
```

Роли и доступы в ArgoCD

БЕКОН'26

```
rbac:
  policy.csv: |
    # Права для роли user в argo-cd и соответствующей группы          Users
    p, role:user, applications, get,      policies-admission-control/*, allow
    p, role:user, applications, update,      policies-admission-control/*, allow
    p, role:user, applications, sync,      policies-admission-control/*, allow

    p, role:user, applicationsets, get,      policies-admission-control/*, allow
    p, role:user, applicationsets, update,      policies-admission-control/*, allow

    p, role:user, clusters, get,      policies-admission-control, allow
    p, role:user, projects, get,      policies-admission-control, allow
    p, role:user, repositories, get,      policies-admission-control/*, allow
    p, role:user, logs, get,      policies-admission-control/*, allow
    g,          Users, role:user

    # Права для роли secadmin в argo-cd и соответствующей группы          SecAdm
    p, role:secadmin, applications, get, */*, allow
    p, role:secadmin, applicationsets, get, */*, allow
    p, role:secadmin, clusters, get, *, allow
    p, role:secadmin, projects, get, *, allow
    p, role:secadmin, repositories, get, *, allow
    p, role:secadmin, accounts, get, *, allow
    p, role:secadmin, certificates, get, *, allow
    p, role:secadmin, gpgkeys, get, *, allow
    p, role:secadmin, logs, get, */*, allow
    g,          SecAdm, role:secadmin
```

```
# Права для роли admin в argo-cd и соответствующей группы          Admins
p, role:admin, applications, get, */*, allow
p, role:admin, applications, create, */*, allow
p, role:admin, applications, update, */*, allow
p, role:admin, applications, delete, */*, allow
p, role:admin, applications, sync, */*, allow
p, role:admin, applications, action, */*, allow
p, role:admin, applications, override, */*, allow

p, role:admin, applicationsets, get, */*, allow
p, role:admin, applicationsets, create, */*, allow
p, role:admin, applicationsets, update, */*, allow
p, role:admin, applicationsets, delete, */*, allow

p, role:admin, clusters, get, *, allow
p, role:admin, clusters, create, *, allow
p, role:admin, clusters, update, *, allow
p, role:admin, clusters, delete, *, allow

p, role:admin, projects, get, *, allow
p, role:admin, projects, create, *, allow
p, role:admin, projects, update, *, allow
p, role:admin, projects, delete, *, allow

p, role:admin, repositories, get, *, allow
p, role:admin, repositories, create, *, allow
p, role:admin, repositories, update, *, allow
p, role:admin, repositories, delete, *, allow

p, role:admin, accounts, get, *, allow
```

Роли и доступы в ArgoCD

БЕКОН'26

```
rbac:
  policy.csv: |
    # Права для роли user в argo-cd и соответствующей группы          Users
    p, role:user, applications, get,      policies-admission-control/*, allow
    p, role:user, applications, update,      policies-admission-control/*, allow
    p, role:user, applications, sync,      policies-admission-control/*, allow

    p, role:user, applicationsets, get,      policies-admission-control/*, allow
    p, role:user, applicationsets, update,      policies-admission-control/*, allow

    p, role:user, clusters, get,      policies-admission-control, allow
    p, role:user, projects, get,      policies-admission-control, allow
    p, role:user, repositories, get,      policies-admission-control/*, allow
    p, role:user, logs, get,      policies-admission-control/*, allow
    g,           Users, role:user

    # Права для роли secadmin в argo-cd и соответствующей группы          SecAdm
    p, role:secadmin, applications, get, */*, allow
    p, role:secadmin, applicationsets, get, */*, allow
    p, role:secadmin, clusters, get, *, allow
    p, role:secadmin, projects, get, *, allow
    p, role:secadmin, repositories, get, *, allow
    p, role:secadmin, accounts, get, *, allow
    p, role:secadmin, certificates, get, *, allow
    p, role:secadmin, gpgkeys, get, *, allow
    p, role:secadmin, logs, get, */*, allow
    g,           SecAdm, role:secadmin
```

```
# Права для роли admin в argo-cd и соответствующей группы          Admins
p, role:admin, applications, get, */*, allow
p, role:admin, applications, create, */*, allow
p, role:admin, applications, update, */*, allow
p, role:admin, applications, delete, */*, allow
p, role:admin, applications, sync, */*, allow
p, role:admin, applications, action, */*, allow
p, role:admin, applications, override, */*, allow

p, role:admin, applicationsets, get, */*, allow
p, role:admin, applicationsets, create, */*, allow
p, role:admin, applicationsets, update, */*, allow
p, role:admin, applicationsets, delete, */*, allow

p, role:admin, clusters, get, *, allow
p, role:admin, clusters, create, *, allow
p, role:admin, clusters, update, *, allow
p, role:admin, clusters, delete, *, allow

p, role:admin, projects, get, *, allow
p, role:admin, projects, create, *, allow
p, role:admin, projects, update, *, allow
p, role:admin, projects, delete, *, allow

p, role:admin, repositories, get, *, allow
p, role:admin, repositories, create, *, allow
p, role:admin, repositories, update, *, allow
p, role:admin, repositories, delete, *, allow

p, role:admin, accounts, get, *, allow
```

Роли и доступы в ArgoCD

БЕКОН'26

```
rbac:
  policy.csv: |
    # Права для роли user в argo-cd и соответствующей группы          Users
    p, role:user, applications, get,      policies-admission-control/*, allow
    p, role:user, applications, update,      policies-admission-control/*, allow
    p, role:user, applications, sync,      policies-admission-control/*, allow

    p, role:user, applicationsets, get,      policies-admission-control/*, allow
    p, role:user, applicationsets, update,      policies-admission-control/*, allow

    p, role:user, clusters, get,      policies-admission-control, allow
    p, role:user, projects, get,      policies-admission-control, allow
    p, role:user, repositories, get,      policies-admission-control/*, allow
    p, role:user, logs, get,      policies-admission-control/*, allow
    g,          Users, role:user

    # Права для роли secadmin в argo-cd и соответствующей группы          SecAdm
    p, role:secadmin, applications, get, */*, allow
    p, role:secadmin, applicationsets, get, */*, allow
    p, role:secadmin, clusters, get, *, allow
    p, role:secadmin, projects, get, *, allow
    p, role:secadmin, repositories, get, *, allow
    p, role:secadmin, accounts, get, *, allow
    p, role:secadmin, certificates, get, *, allow
    p, role:secadmin, gpgkeys, get, *, allow
    p, role:secadmin, logs, get, */*, allow
    g,          SecAdm, role:secadmin
```

```
# Права для роли admin в argo-cd и соответствующей группы          Admins
p, role:admin, applications, get, */*, allow
p, role:admin, applications, create, */*, allow
p, role:admin, applications, update, */*, allow
p, role:admin, applications, delete, */*, allow
p, role:admin, applications, sync, */*, allow
p, role:admin, applications, action, */*, allow
p, role:admin, applications, override, */*, allow

p, role:admin, applicationsets, get, */*, allow
p, role:admin, applicationsets, create, */*, allow
p, role:admin, applicationsets, update, */*, allow
p, role:admin, applicationsets, delete, */*, allow

p, role:admin, clusters, get, *, allow
p, role:admin, clusters, create, *, allow
p, role:admin, clusters, update, *, allow
p, role:admin, clusters, delete, *, allow

p, role:admin, projects, get, *, allow
p, role:admin, projects, create, *, allow
p, role:admin, projects, update, *, allow
p, role:admin, projects, delete, *, allow

p, role:admin, repositories, get, *, allow
p, role:admin, repositories, create, *, allow
p, role:admin, repositories, update, *, allow
p, role:admin, repositories, delete, *, allow

p, role:admin, accounts, get, *, allow
```

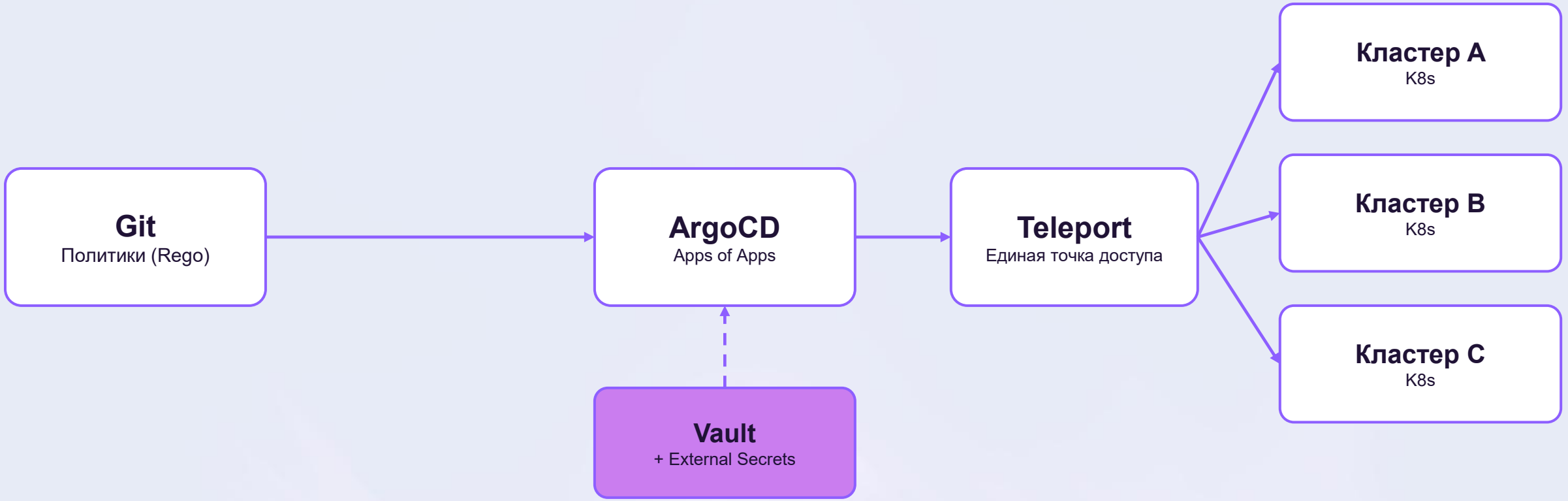
Роли и доступы в ArgoCD

БЕКОН'26

```
rbac:
  policy.csv: |
    # Права для роли user в argo-cd и соответствующей группы          Users
    p, role:user, applications, get,      policies-admission-control/*, allow
    p, role:user, applications, update,      policies-admission-control/*, allow
    p, role:user, applications, sync,      policies-admission-control/*, allow

    p, role:user, applicationsets, get,      policies-admission-control/*, allow
    p, role:user, applicationsets, update,      policies-admission-control/*, allow

    p, role:user, clusters, get,      policies-admission-control, allow
    p, role:user, projects, get,      policies-admission-control, allow
    p, role:user, repositories, get,      policies-admission-control/*, allow
    p, role:user, logs, get,      policies-admission-control/*, allow
    g,          Users, role:user

    # Права для роли secadmin в argo-cd и соответствующей группы          SecAdm
    p, role:secadmin, applications, get, */*, allow
    p, role:secadmin, applicationsets, get, */*, allow
    p, role:secadmin, clusters, get, *, allow
    p, role:secadmin, projects, get, *, allow
    p, role:secadmin, repositories, get, *, allow
    p, role:secadmin, accounts, get, *, allow
    p, role:secadmin, certificates, get, *, allow
    p, role:secadmin, gpgkeys, get, *, allow
    p, role:secadmin, logs, get, */*, allow
    g,          SecAdm, role:secadmin
```

```
# Права для роли admin в argo-cd и соответствующей группы          Admins
p, role:admin, applications, get, */*, allow
p, role:admin, applications, create, */*, allow
p, role:admin, applications, update, */*, allow
p, role:admin, applications, delete, */*, allow
p, role:admin, applications, sync, */*, allow
p, role:admin, applications, action, */*, allow
p, role:admin, applications, override, */*, allow

p, role:admin, applicationsets, get, */*, allow
p, role:admin, applicationsets, create, */*, allow
p, role:admin, applicationsets, update, */*, allow
p, role:admin, applicationsets, delete, */*, allow

p, role:admin, clusters, get, *, allow
p, role:admin, clusters, create, *, allow
p, role:admin, clusters, update, *, allow
p, role:admin, clusters, delete, *, allow

p, role:admin, projects, get, *, allow
p, role:admin, projects, create, *, allow
p, role:admin, projects, update, *, allow
p, role:admin, projects, delete, *, allow

p, role:admin, repositories, get, *, allow
p, role:admin, repositories, create, *, allow
p, role:admin, repositories, update, *, allow
p, role:admin, repositories, delete, *, allow

p, role:admin, accounts, get, *, allow
```

Стек: Vault — секреты для подключения



Vault + External Secrets: секреты централизованно

БЕКОН'26

Зачем нужен Vault здесь

- Секрет подключения кластера содержит:
- • tbot token
- • tlsClientConfig (caData)
- • execProviderConfig

- Хранить в Git нельзя — это sensitive data

- VaultStaticSecret → External Secrets Operator
- → Kubernetes Secret в namespace ArgoCD

Как ArgoCD видит кластеры

- ArgoCD читает Secret с типом
- argocd.argoproj.io/secret-type=cluster

- При добавлении нового кластера:
- 1. Добавить секрет в Vault
- 2. Создать VaultStaticSecret манифест
- 3. ESO создаёт Kubernetes Secret
- 4. ArgoCD подхватывает кластер автоматически

- **⚠ Vault как точка отказа?**
- В нашем случае Vault — отдельный сервис с отдельной командой и SLA. Он не является частью этого стека: при недоступности Vault новые кластеры не подключатся, но уже работающие продолжат получать политики через ArgoCD.

Vault + External Secrets: секреты централизованно

БЕКОН'26

```yaml
apiVersion: secrets.hashicorp.com/v1beta1
kind: VaultStaticSecret
metadata:
  name: k8s
  namespace:
spec:
  destination:
    create: true
    name: k8s
    labels:
      argocd.argoproj.io/secret-type: cluster
      cluster: k8s
      environment: V
      opal-agent: gatekeeper
      zone: int
    overwrite: true
  mount: secret_v2_p
  path:     argo-cd/clusters/exec-provider-config/k8s
  refreshAfter: 5s
  type: kv-v2
  vaultAuthRef: vault-auth
  rolloutRestartTargets:
    - kind: StatefulSet
      name:     argo-cd-argocd-application-controller
    - kind: Deployment
      name:     argo-cd-argocd-server
```

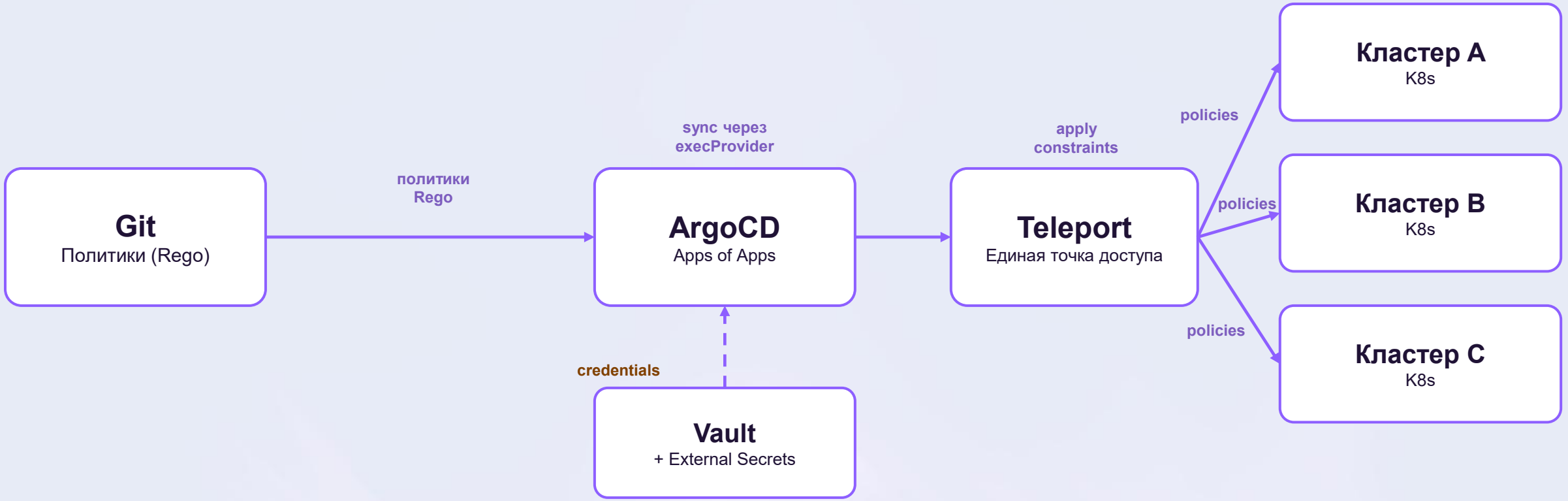
Vault + External Secrets: секреты централизованно

```yaml
apiVersion: secrets.hashicorp.com/v1beta1
kind: VaultStaticSecret
metadata:
  name: k8s
  namespace:
spec:
  destination:
    create: true
    name: k8s
    labels:
      argocd.argoproj.io/secret-type: cluster
      cluster: k8s
      environment: V
      opal-agent: gatekeeper
      zone: int
    overwrite: true
  mount: secret_v2_p
  path:     argo-cd/clusters/exec-provider-config/k8s
  refreshAfter: 5s
  type: kv-v2
  vaultAuthRef: vault-auth
  rolloutRestartTargets:
    - kind: StatefulSet
      name:     argo-cd-argocd-application-controller
    - kind: Deployment
      name:     argo-cd-argocd-server
```

Vault + External Secrets: секреты централизованно

```yaml
apiVersion: secrets.hashicorp.com/v1beta1
kind: VaultStaticSecret
metadata:
  name: k8s
  namespace:
spec:
  destination:
    create: true
    name: k8s
    labels:
      argocd.argoproj.io/secret-type: cluster
      cluster: k8s
      environment: V
      opal-agent: gatekeeper
      zone: int
    overwrite: true
  mount: secret_v2_p
  path:     argo-cd/clusters/exec-provider-config/k8s
  refreshAfter: 5s
  type: kv-v2
  vaultAuthRef: vault-auth
  rolloutRestartTargets:
    - kind: StatefulSet
      name:     argo-cd-argocd-application-controller
    - kind: Deployment
      name:     argo-cd-argocd-server
```

Стек: финальная сборка



Почему Gatekeeper, а не Kyverno?

БЕКОН'26

Kyverno YAML, K8s-native

+ Простой YAML, низкий порог входа
+ Generation policies (создание ресурсов)

— **Сложная data-driven логика**
выразительность YAML ограничена при условиях с итерациями и вычислениями

— **Cross-resource валидация**
правила, опирающиеся на состояние нескольких ресурсов одновременно — отдельная боль

— **Проблемы при масштабе**
известные кейсы: 3.5k политик → OOM компонентов, перегрузка API-сервера при массовом ребилде

Gatekeeper OPA/Rego

+ Полноценный язык: условия, итерации, функции
+ Separation of concerns: Template (логика) + Constraint (параметры)
+ Опыт OPA переиспользуется (Conftest, Terraform)
+ В команде уже были инженеры с опытом Rego — порог входа не был проблемой

— **Порог входа: Rego — отдельный язык**
— **Generation ресурсов отсутствует**

VAP (нативные ValidatingAdmissionPolicy): на момент проекта только начинался — зрелых паттернов и экосистемы ещё не было. Как дела обстоят сейчас — отдельный доклад на этой конференции.

Кастомизация: исключения в Rego

Задача

Политика доставлена на кластер. Нужно исключить конкретный namespace или pod — без релиза новой версии политики.

Решение: метка на namespace

- **Команда:**

```
kubectl label ns kube-system \
  policy.gatekeeper.sh/skip=true
```

- **Что делает Rego:**

  видит метку → пропускает namespace

- **Где живёт правило исключения:**

  в самом Rego, один раз на всю политику

Логика проверки в Rego

1. Берём namespace из проверяемого объекта
2. Достаём его метаданные через data.inventory
3. Если есть метка skip=true — выходим (не нарушение)
4. Иначе применяем основную проверку

Что это даёт

- **Гранулярно** — до конкретного объекта
- **Без релиза** новой версии политики
- **Audit-trail** через Git: кто и когда добавил метку

Кастомизация: исключения в Rego

БЕКОН'26

```yaml
apiVersion: constraints.gatekeeper.sh/v1beta1
kind: K8sPSPPrivilegedContainer
metadata:
  name: psp-privileged-container
  annotations:
    argocd.argoproj.io/sync-wave: "1"
spec:
  enforcementAction: deny
  match:
    kinds:
      - apiGroups: [""]
        kinds: ["Pod"]
  parameters:
    instructionMessage: "Применена запрещающая политика использования privileged = true"
    exemptions:
      - namespace: kube-system
      - namespace: istio-system
      - namespace: monitoring
      - namespace: kyverno
      - namespace: cert-manager
```

```
# Исключение: разрешить всё в namespace
is_exempt {
  ns := input.review.object.metadata.namespace
  exemption := input.parameters.exemptions[_]
  exemption.namespace == ns
  not exemption.podNames
  not exemption.images
}

# Исключение: по имени Pod
is_exempt {
  ns := input.review.object.metadata.namespace
  pod_name := input.review.object.metadata.name
  exemption := input.parameters.exemptions[_]
  exemption.namespace == ns
  exemption.podNames
  pattern := exemption.podNames[_]
  _matches(pod_name, pattern)
}

# Исключение: по образу контейнера
is_exempt {
  ns := input.review.object.metadata.namespace
  exemption := input.parameters.exemptions[_]
  exemption.namespace == ns
  exemption.images
  c := input_containers[_]
  img := c.image
  pattern := exemption.images[_]
  _matches(img, pattern)
}

# Wildcard matcher (suffix *)
```

Кастомизация: исключения в Rego

БЕКОН'26

```yaml
apiVersion: constraints.gatekeeper.sh/v1beta1
kind: K8sPSPPrivilegedContainer
metadata:
  name: psp-privileged-container
  annotations:
    argocd.argoproj.io/sync-wave: "1"
spec:
  enforcementAction: deny
  match:
    kinds:
      - apiGroups: [""]
        kinds: ["Pod"]
  parameters:
    instructionMessage: "Применена запрещающая политика использования privileged = true"
    exemptions:
      - namespace: kube-system
      - namespace: istio-system
      - namespace: monitoring
      - namespace: kyverno
      - namespace: cert-manager
```

```
# Исключение: разрешить всё в namespace
is_exempt {
  ns := input.review.object.metadata.namespace
  exemption := input.parameters.exemptions[_]
  exemption.namespace == ns
  not exemption.podNames
  not exemption.images
}

# Исключение: по имени Pod
is_exempt {
  ns := input.review.object.metadata.namespace
  pod_name := input.review.object.metadata.name
  exemption := input.parameters.exemptions[_]
  exemption.namespace == ns
  exemption.podNames
  pattern := exemption.podNames[_]
  _matches(pod_name, pattern)
}

# Исключение: по образу контейнера
is_exempt {
  ns := input.review.object.metadata.namespace
  exemption := input.parameters.exemptions[_]
  exemption.namespace == ns
  exemption.images
  c := input_containers[_]
  img := c.image
  pattern := exemption.images[_]
  _matches(img, pattern)
}

# Wildcard matcher (suffix *)
```

Кастомизация: исключения в Rego

БЕКОН'26

```yaml
apiVersion: constraints.gatekeeper.sh/v1beta1
kind: K8sPSPPrivilegedContainer
metadata:
  name: psp-privileged-container
  annotations:
    argocd.argoproj.io/sync-wave: "1"
spec:
  enforcementAction: deny
  match:
    kinds:
      - apiGroups: [""]
        kinds: ["Pod"]
  parameters:
    instructionMessage: "Применена запрещающая политика использования privileged = true"
    exemptions:
      - namespace: kube-system
      - namespace: istio-system
      - namespace: monitoring
      - namespace: kyverno
      - namespace: cert-manager
```

```
# Исключение: разрешить всё в namespace
is_exempt {
  ns := input.review.object.metadata.namespace
  exemption := input.parameters.exemptions[_]
  exemption.namespace == ns
  not exemption.podNames
  not exemption.images
}

# Исключение: по имени Pod
is_exempt {
  ns := input.review.object.metadata.namespace
  pod_name := input.review.object.metadata.name
  exemption := input.parameters.exemptions[_]
  exemption.namespace == ns
  exemption.podNames
  pattern := exemption.podNames[_]
  _matches(pod_name, pattern)
}

# Исключение: по образу контейнера
is_exempt {
  ns := input.review.object.metadata.namespace
  exemption := input.parameters.exemptions[_]
  exemption.namespace == ns
  exemption.images
  c := input_containers[_]
  img := c.image
  pattern := exemption.images[_]
  _matches(img, pattern)
}

# Wildcard matcher (suffix *)
```

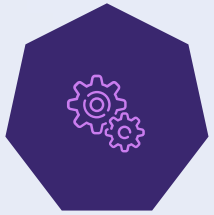
БЕКОН'26

ПРОЦЕСС

02

Как подключить новый кластер

ШАГ 1
VAULT + SECRET

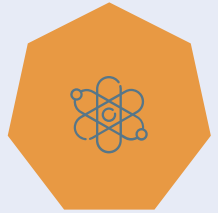


Создать секрет в Vault
Создать VaultStaticSecret
ESO создаёт k8s Secret
ArgoCD видит кластер

ШАГ 2
ЛЕЙБЛЫ

Лейблы кластера указаны у VaultStaticSecret в git
zone=prod / zone=test
cluster=имя-кластера
ArgoCD подхватывает автоматически

ШАГ 3
АВТОМАТИКА



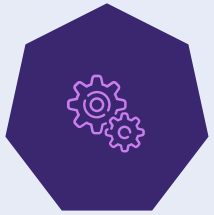
ApplicationSet выбирает репозиторий
Gatekeeper получает политики
Кластер сразу compliant

Жизненный цикл политики

БЕКОН'26

ЭТАП 1
NEW POLICY

Git commit



Пишем ConstraintTemplate
(логика Rego) +
Constraint (параметры)
Коммит в нужный репо
PR Review + аппрув команды кластера

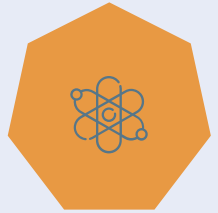
ЭТАП 2
DELIVERY

ArgoCD sync

ArgoCD синхронизирует
по лейблам: wide / zone / cluster
Все нужные кластеры
получают политику

ЭТАП 3
ENFORCE

Gatekeeper



warn: логируем, не блокируем
deny: блокируем ресурс
Metrics → Victoria Metrics
→ Grafana дашборд

БЕКОН'26

МОНИТОРИНГ И OBSERVABILITY

03

КОНФЕРЕНЦИЯ ПО БЕЗОПАСНОСТИ КОНТЕЙНЕРОВ И КОНТЕЙНЕРНЫХ СРЕД

Мониторинг: Victoria Metrics + OPA Scorecard

- **Victoria Metrics (push-модель)**
- Агент в каждом кластере собирает /metrics сервиса в namespace
- → отправляет централизованно в единый VM
- → данные из всех кластеров в одном месте

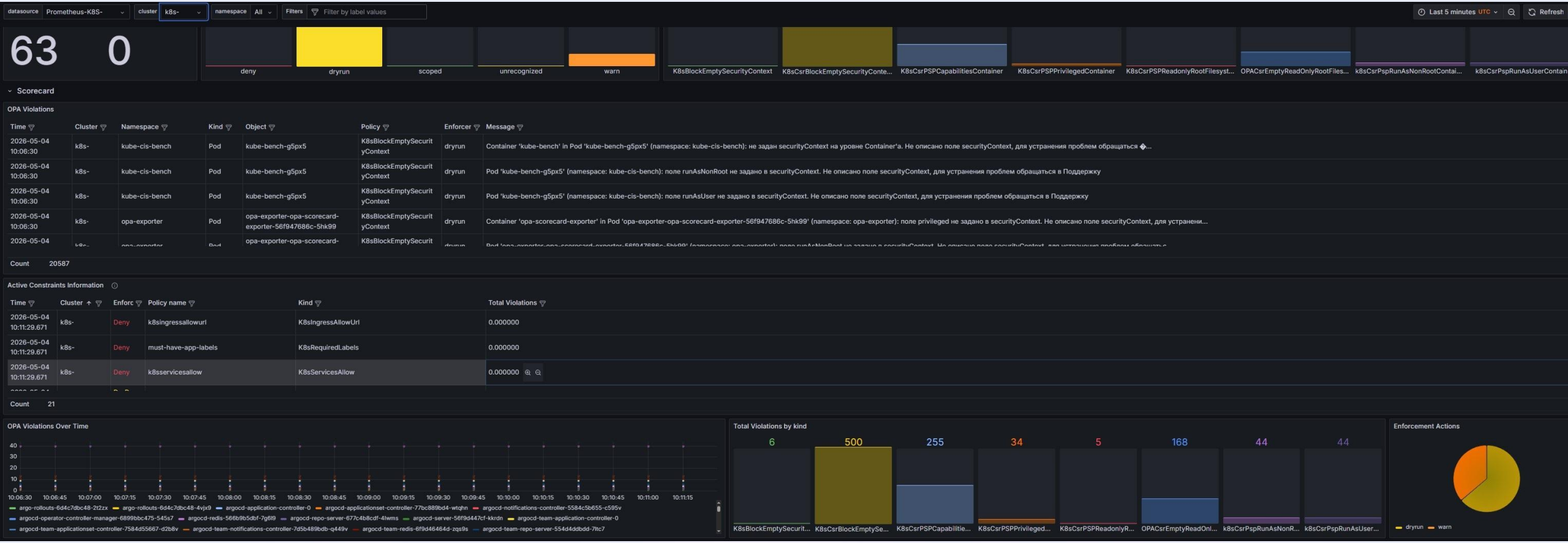
- **OPA Scorecard**
- Расширение Gatekeeper — экспортирует violations как Prometheus-метрики
- gatekeeper_violations{constraint_kind, enforcement_action, ...}

- Алерты: недоступность ArgoCD, зависшая синхронизация, сработки политик → письмо на почту / webhook

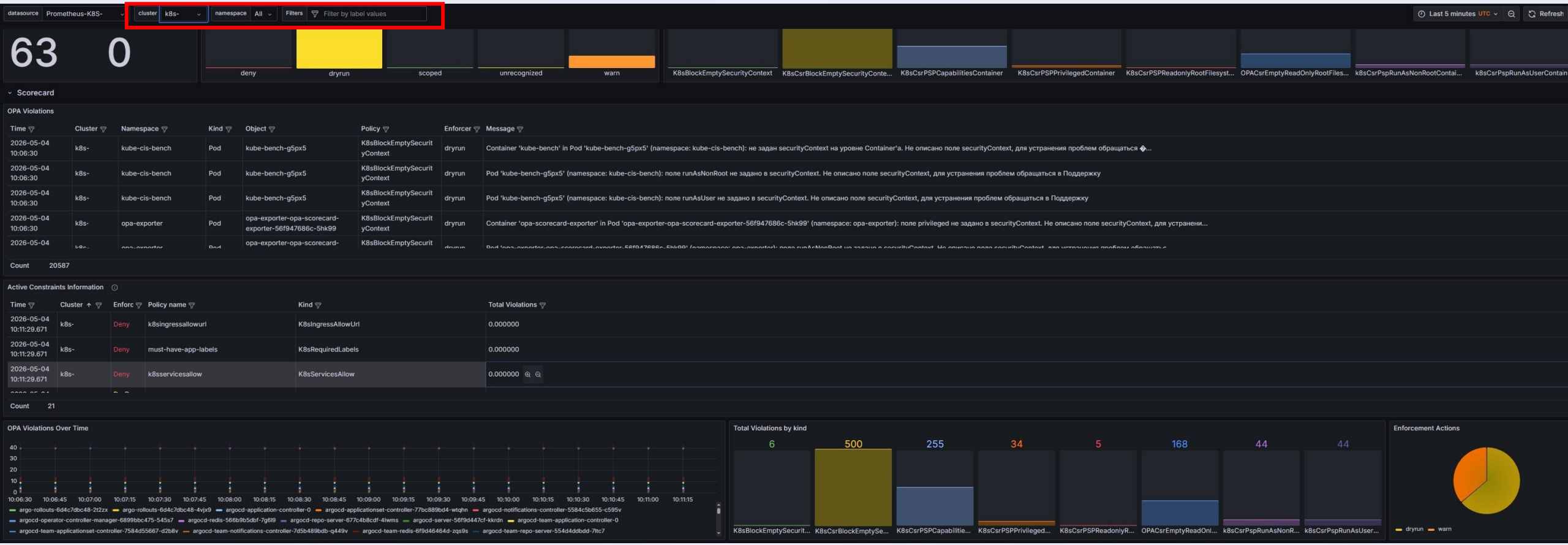
- **Что умеет этот стек**
- \+ Violations в реальном времени по кластерам, политикам, enforcement_action
- \+ Фильтрация и группировка в Grafana — по кластеру, зоне, constraint_kind

Мониторинг: Victoria Metrics + OPA Scorecard

БЕКОН'26

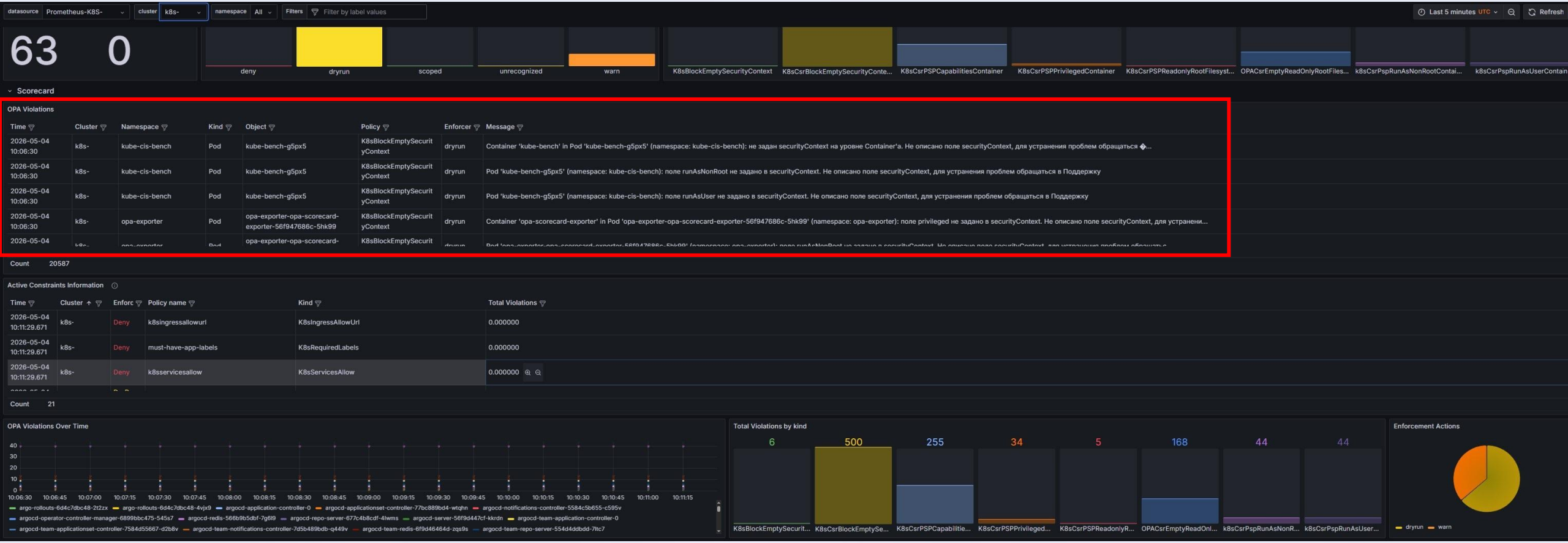


Мониторинг: Victoria Metrics + OPA Scorecard



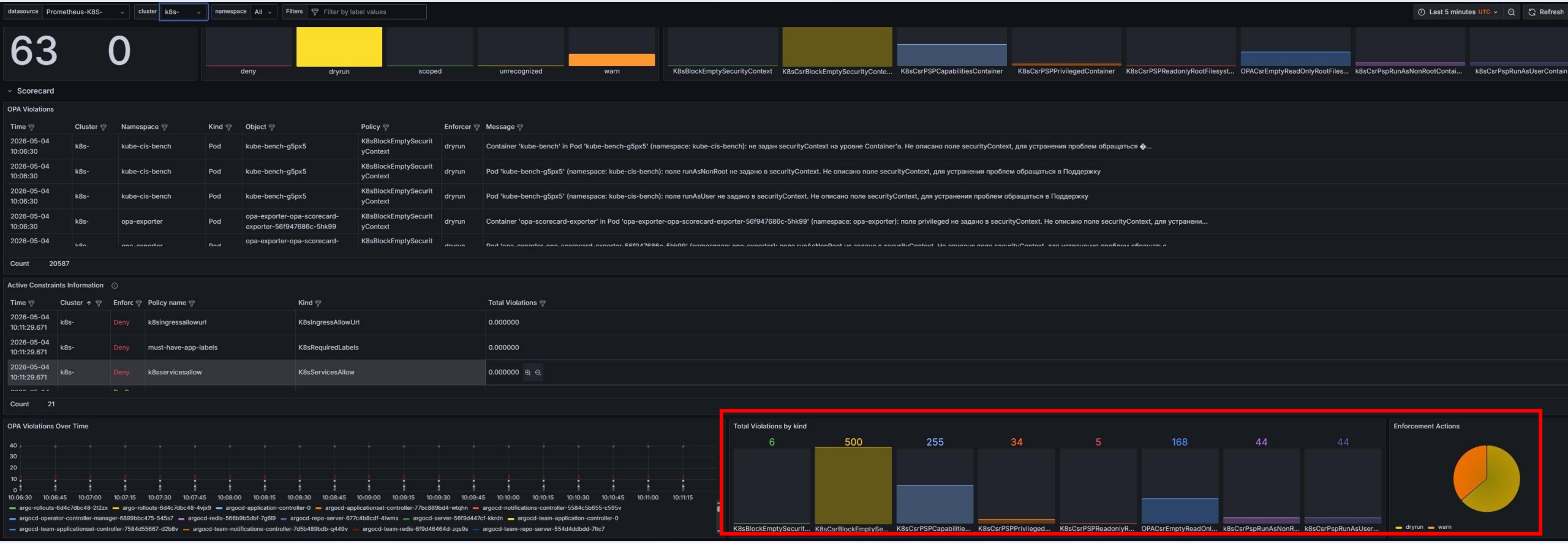
Мониторинг: Victoria Metrics + OPA Scorecard

БЕКОН'26



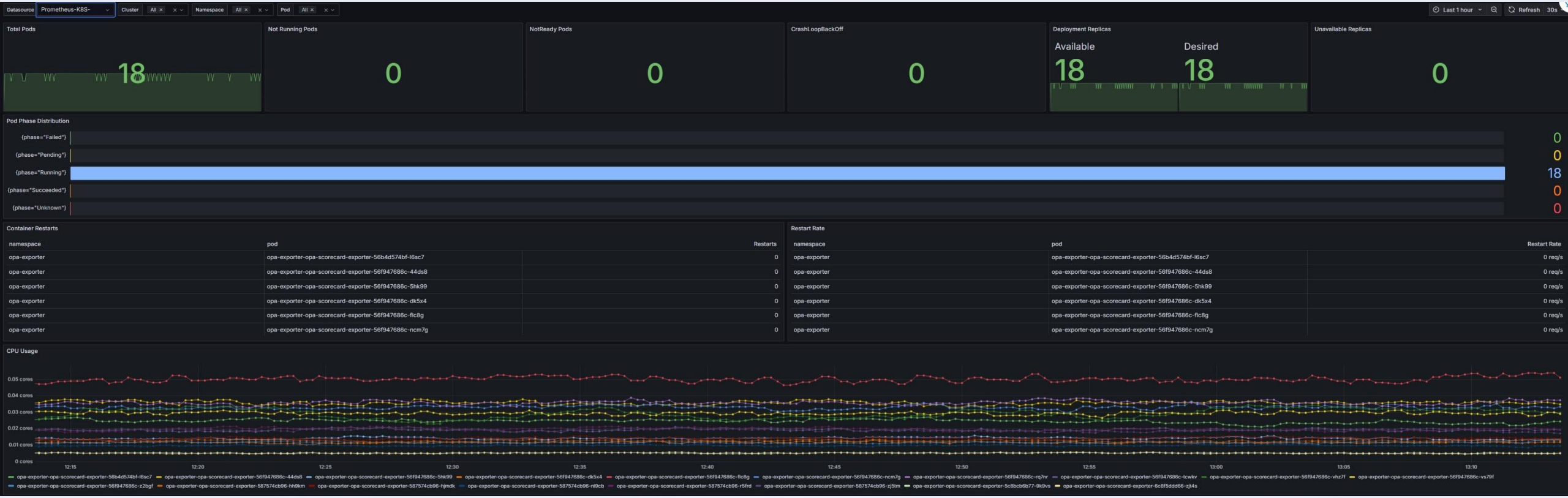
Мониторинг: Victoria Metrics + OPA Scorecard

БЕКОН'26



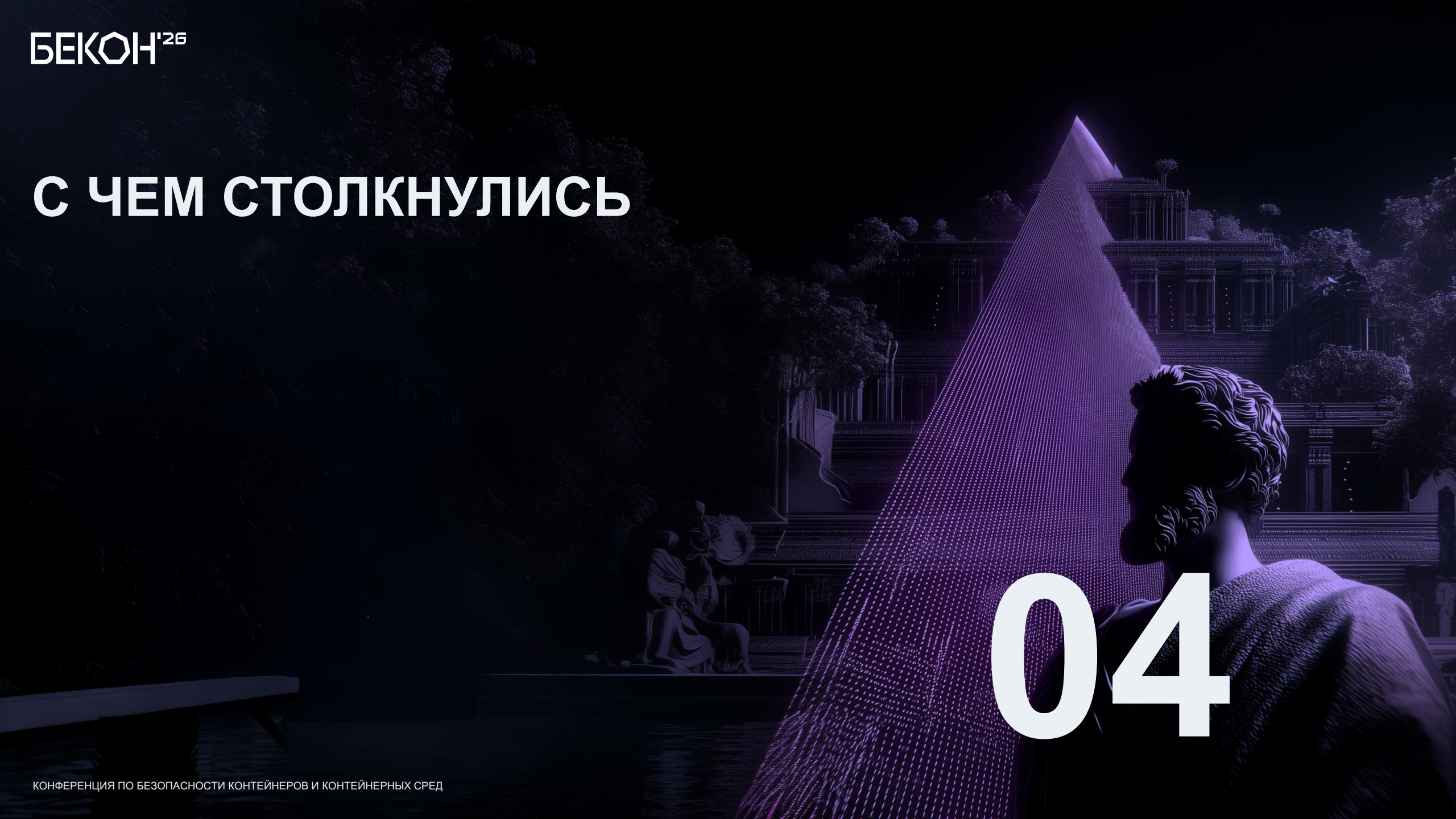
Мониторинг: Victoria Metrics + OPA Scorecard

БЕКОН'26



БЕКОН'26

С ЧЕМ СТОЛКНУЛИСЬ



04

Teleport: проблема ротации tbot токена

- **Проблема ротации tbot токена**

- ArgoCD вызывает execProviderConfig параллельно —
  - до 5+ горутин одновременно для одного кластера
  - → 5 × 150MB = 750MB только на tbot

- **Решение: flock + кэш credentials**

- Первый вызов: tbot start → пишет credentials
- Остальные: ждут lock → читают готовое
- TTL кэша: 5 мин, одно обновление в час

Teleport: ротация токенов и один proxy — много кластеров

БЕКОН'26

```
[2026-03-27 08:49:47] [3204] [] --- INVOKED --- args: auth_server=k8s-xxx cluster=k8s-dev cert_provided=yes
[2026-03-27 08:49:47] [3204] [] MEMORY: MemTotal:=49321716 MemFree:=10107420 MemAvailable:=37628232 Buffers:=2368 Cached:=21985276 Slab:=8080896 SReclaimable:=6117124 | cgroup: usage=344236032 limit=536870912 anon(RSS)=28696985
[2026-03-27 08:49:47] [3204] [] MEMORY-TOP: argocd-applicat=367292kB script.sh=3876kB script.sh=2544kB script.sh=2212kB awk=1344kB
[2026-03-27 08:49:47] [3204] [] CACHE EXPIRED: age=3600s >= ttl=300s — need renewal
[2026-03-27 08:49:47] [3204] [] CLEANUP: removing stale credential files
[2026-03-27 08:49:47] [3204] [] MEMORY: MemTotal:=49321716 MemFree:=10102384 MemAvailable:=37623196 Buffers:=2368 Cached:=21985284 Slab:=8080832 SReclaimable:=6117124 | cgroup: usage=343609344 limit=536870912 anon(RSS)=28700262
[2026-03-27 08:49:47] [3204] [] MEMORY-TOP: argocd-applicat=367292kB script.sh=3880kB script.sh=2552kB script.sh=2220kB awk=1516kB
[2026-03-27 08:49:47] [3204] [] DISK: /tmp=51M cluster_dir=52K files=4
[2026-03-27 08:49:47] [3204] [] RENEW: starting tbot start...
[2026-03-27 08:49:49] [3204] [] RENEW: tbot start finished in 1985ms (rc=0)
[2026-03-27 08:49:49] [3204] [] MEMORY: MemTotal:=49321716 MemFree:=10076856 MemAvailable:=37598236 Buffers:=2368 Cached:=21985864 Slab:=8079008 SReclaimable:=6117124 | cgroup: usage=344178688 limit=536870912 anon(RSS)=28702720
[2026-03-27 08:49:49] [3204] [] MEMORY-TOP: argocd-applicat=367292kB script.sh=3900kB script.sh=2568kB script.sh=2236kB awk=1436kB
[2026-03-27 08:49:49] [3204] [] DROP_CACHES: no permission to write /proc/sys/vm/drop_caches (need SYS_ADMIN cap or sysctl vm.drop_caches_userns=1)
[2026-03-27 08:49:49] [3204] [] DROP_CACHES: consider adding to securityContext: capabilities: add: ['SYS_ADMIN'] or use sysctl
[2026-03-27 08:49:49] [3204] [] MEMORY: MemTotal:=49321716 MemFree:=10073228 MemAvailable:=37594628 Buffers:=2368 Cached:=21985864 Slab:=8079008 SReclaimable:=6117124 | cgroup: usage=343822336 limit=536870912 anon(RSS)=28704768
[2026-03-27 08:49:49] [3204] [] MEMORY-TOP: argocd-applicat=367292kB script.sh=3964kB script.sh=2568kB script.sh=2236kB awk=1400kB
[2026-03-27 08:49:49] [3204] [] RENEW: SUCCESS — certificates renewed
[2026-03-27 08:49:49] [3204] [] RENEW: tbot kube credentials OK
[2026-03-27 08:49:50] [3204] [] DISK: /tmp=51M cluster_dir=108K files=18
[2026-03-27 09:49:50] [3401] [] --- INVOKED --- args: auth_server=k8s-xxx cluster=k8s-dev cert_provided=yes
[2026-03-27 09:49:50] [3401] [] MEMORY: MemTotal:=49321716 MemFree:=10585400 MemAvailable:=38240376 Buffers:=2368 Cached:=22116360 Slab:=8087976 SReclaimable:=6120132 | cgroup: usage=343261184 limit=536870912 anon(RSS)=28676096
[2026-03-27 09:49:50] [3401] [] MEMORY-TOP: argocd-applicat=366896kB script.sh=3728kB script.sh=2460kB script.sh=2264kB awk=1440kB
[2026-03-27 09:49:50] [3401] [] CACHE EXPIRED: age=3601s >= ttl=300s — need renewal
[2026-03-27 09:49:50] [3401] [] CLEANUP: removing stale credential files
[2026-03-27 09:49:50] [3401] [] MEMORY: MemTotal:=49321716 MemFree:=10588244 MemAvailable:=38243220 Buffers:=2368 Cached:=22116368 Slab:=8087976 SReclaimable:=6120132 | cgroup: usage=344535040 limit=536870912 anon(RSS)=28680192
[2026-03-27 09:49:50] [3401] [] MEMORY-TOP: argocd-applicat=366896kB script.sh=3732kB script.sh=2468kB script.sh=2272kB awk=1368kB
[2026-03-27 09:49:50] [3401] [] DISK: /tmp=51M cluster_dir=56K files=4
[2026-03-27 09:49:50] [3401] [] RENEW: starting tbot start...
[2026-03-27 09:49:51] [3401] [] RENEW: tbot start finished in 1375ms (rc=0)
[2026-03-27 09:49:51] [3401] [] MEMORY: MemTotal:=49321716 MemFree:=10589800 MemAvailable:=38245140 Buffers:=2368 Cached:=22116736 Slab:=8088000 SReclaimable:=6120132 | cgroup: usage=343638016 limit=536870912 anon(RSS)=28682240
[2026-03-27 09:49:52] [3401] [] MEMORY-TOP: argocd-applicat=366896kB script.sh=3752kB script.sh=2484kB script.sh=2288kB awk=1476kB
[2026-03-27 09:49:52] [3401] [] DROP_CACHES: no permission to write /proc/sys/vm/drop_caches (need SYS_ADMIN cap or sysctl vm.drop_caches_userns=1)
[2026-03-27 09:49:52] [3401] [] DROP_CACHES: consider adding to securityContext: capabilities: add: ['SYS_ADMIN'] or use sysctl
[2026-03-27 09:49:52] [3401] [] MEMORY: MemTotal:=49321716 MemFree:=10585868 MemAvailable:=38241232 Buffers:=2368 Cached:=22116816 Slab:=8087880 SReclaimable:=6120132 | cgroup: usage=343179264 limit=536870912 anon(RSS)=28684288
[2026-03-27 09:49:52] [3401] [] MEMORY-TOP: argocd-applicat=366896kB script.sh=3752kB script.sh=2484kB script.sh=2288kB awk=1364kB
[2026-03-27 09:49:52] [3401] [] RENEW: SUCCESS — certificates renewed
[2026-03-27 09:49:52] [3401] [] RENEW: tbot kube credentials OK
[2026-03-27 09:49:52] [3401] [] DISK: /tmp=51M cluster_dir=112K fil
```

Teleport: ротация токенов и один proxy — много кластеров

БЕКОН'26

```
[2026-03-27 08:49:47] [3204] [] --- INVOKED --- args: auth_server=k8s-xxx cluster=k8s-dev cert_provided=yes
[2026-03-27 08:49:47] [3204] [] MEMORY: MemTotal:=49321716 MemFree:=10107420 MemAvailable:=37628232 Buffers:=2368 Cached:=21985276 Slab:=8080896 SReclaimable:=6117124 | cgroup: usage=344236032 limit=536870912 anon(RSS)=28696985
[2026-03-27 08:49:47] [3204] [] MEMORY-TOP: argocd-applicat=367292kB script.sh=3876kB script.sh=2544kB script.sh=2212kB awk=1344kB
[2026-03-27 08:49:47] [3204] [] CACHE EXPIRED: age=3600s >= ttl=300s — need renewal
[2026-03-27 08:49:47] [3204] [] CLEANUP: removing stale credential files
[2026-03-27 08:49:47] [3204] [] MEMORY: MemTotal:=49321716 MemFree:=10102384 MemAvailable:=37623196 Buffers:=2368 Cached:=21985284 Slab:=8080832 SReclaimable:=6117124 | cgroup: usage=343609344 limit=536870912 anon(RSS)=28700262
[2026-03-27 08:49:47] [3204] [] MEMORY-TOP: argocd-applicat=367292kB script.sh=3880kB script.sh=2552kB script.sh=2220kB awk=1516kB
[2026-03-27 08:49:47] [3204] [] DISK: /tmp=51M cluster_dir=52K files=4
[2026-03-27 08:49:47] [3204] [] RENEW: starting tbot start...
[2026-03-27 08:49:49] [3204] [] RENEW: tbot start finished in 1985ms (rc=0)
[2026-03-27 08:49:49] [3204] [] MEMORY: MemTotal:=49321716 MemFree:=10076856 MemAvailable:=37598236 Buffers:=2368 Cached:=21985864 Slab:=8079008 SReclaimable:=6117124 | cgroup: usage=344178688 limit=536870912 anon(RSS)=28702720
[2026-03-27 08:49:49] [3204] [] MEMORY-TOP: argocd-applicat=367292kB script.sh=3900kB script.sh=2568kB script.sh=2236kB awk=1436kB
[2026-03-27 08:49:49] [3204] [] DROP_CACHES: no permission to write /proc/sys/vm/drop_caches (need SYS_ADMIN cap or sysctl vm.drop_caches_userns=1)
[2026-03-27 08:49:49] [3204] [] DROP_CACHES: consider adding to securityContext: capabilities: add: ['SYS_ADMIN'] or use sysctl
[2026-03-27 08:49:49] [3204] [] MEMORY: MemTotal:=49321716 MemFree:=10073228 MemAvailable:=37594628 Buffers:=2368 Cached:=21985864 Slab:=8079008 SReclaimable:=6117124 | cgroup: usage=343822336 limit=536870912 anon(RSS)=28704768
[2026-03-27 08:49:49] [3204] [] MEMORY-TOP: argocd-applicat=367292kB script.sh=3964kB script.sh=2568kB script.sh=2236kB awk=1400kB
[2026-03-27 08:49:49] [3204] [] RENEW: SUCCESS — certificates renewed
[2026-03-27 08:49:49] [3204] [] RENEW: tbot kube credentials OK
[2026-03-27 08:49:50] [3204] [] DISK: /tmp=51M cluster_dir=108K files=18
[2026-03-27 09:49:50] [3401] [] --- INVOKED --- args: auth_server=k8s-xxx cluster=k8s-dev cert_provided=yes
[2026-03-27 09:49:50] [3401] [] MEMORY: MemTotal:=49321716 MemFree:=10585400 MemAvailable:=38240376 Buffers:=2368 Cached:=22116360 Slab:=8087976 SReclaimable:=6120132 | cgroup: usage=343261184 limit=536870912 anon(RSS)=28676096
[2026-03-27 09:49:50] [3401] [] MEMORY-TOP: argocd-applicat=366896kB script.sh=3728kB script.sh=2460kB script.sh=2264kB awk=1440kB
[2026-03-27 09:49:50] [3401] [] CACHE EXPIRED: age=3601s >= ttl=300s — need renewal
[2026-03-27 09:49:50] [3401] [] CLEANUP: removing stale credential files
[2026-03-27 09:49:50] [3401] [] MEMORY: MemTotal:=49321716 MemFree:=10588244 MemAvailable:=38243220 Buffers:=2368 Cached:=22116368 Slab:=8087976 SReclaimable:=6120132 | cgroup: usage=344535040 limit=536870912 anon(RSS)=28680192
[2026-03-27 09:49:50] [3401] [] MEMORY-TOP: argocd-applicat=366896kB script.sh=3732kB script.sh=2468kB script.sh=2272kB awk=1368kB
[2026-03-27 09:49:50] [3401] [] DISK: /tmp=51M cluster_dir=56K files=4
[2026-03-27 09:49:50] [3401] [] RENEW: starting tbot start...
[2026-03-27 09:49:51] [3401] [] RENEW: tbot start finished in 1375ms (rc=0)
[2026-03-27 09:49:51] [3401] [] MEMORY: MemTotal:=49321716 MemFree:=10589800 MemAvailable:=38245140 Buffers:=2368 Cached:=22116736 Slab:=8088000 SReclaimable:=6120132 | cgroup: usage=343638016 limit=536870912 anon(RSS)=28682240
[2026-03-27 09:49:52] [3401] [] MEMORY-TOP: argocd-applicat=366896kB script.sh=3752kB script.sh=2484kB script.sh=2288kB awk=1476kB
[2026-03-27 09:49:52] [3401] [] DROP_CACHES: no permission to write /proc/sys/vm/drop_caches (need SYS_ADMIN cap or sysctl vm.drop_caches_userns=1)
[2026-03-27 09:49:52] [3401] [] DROP_CACHES: consider adding to securityContext: capabilities: add: ['SYS_ADMIN'] or use sysctl
[2026-03-27 09:49:52] [3401] [] MEMORY: MemTotal:=49321716 MemFree:=10585868 MemAvailable:=38241232 Buffers:=2368 Cached:=22116816 Slab:=8087880 SReclaimable:=6120132 | cgroup: usage=343179264 limit=536870912 anon(RSS)=28684288
[2026-03-27 09:49:52] [3401] [] MEMORY-TOP: argocd-applicat=366896kB script.sh=3752kB script.sh=2484kB script.sh=2288kB awk=1364kB
[2026-03-27 09:49:52] [3401] [] RENEW: SUCCESS — certificates renewed
[2026-03-27 09:49:52] [3401] [] RENEW: tbot kube credentials OK
[2026-03-27 09:49:52] [3401] [] DISK: /tmp=51M cluster_dir=112K fil
```

Teleport: один proxy — много кластеров

- **Один Teleport proxy — много кластеров**

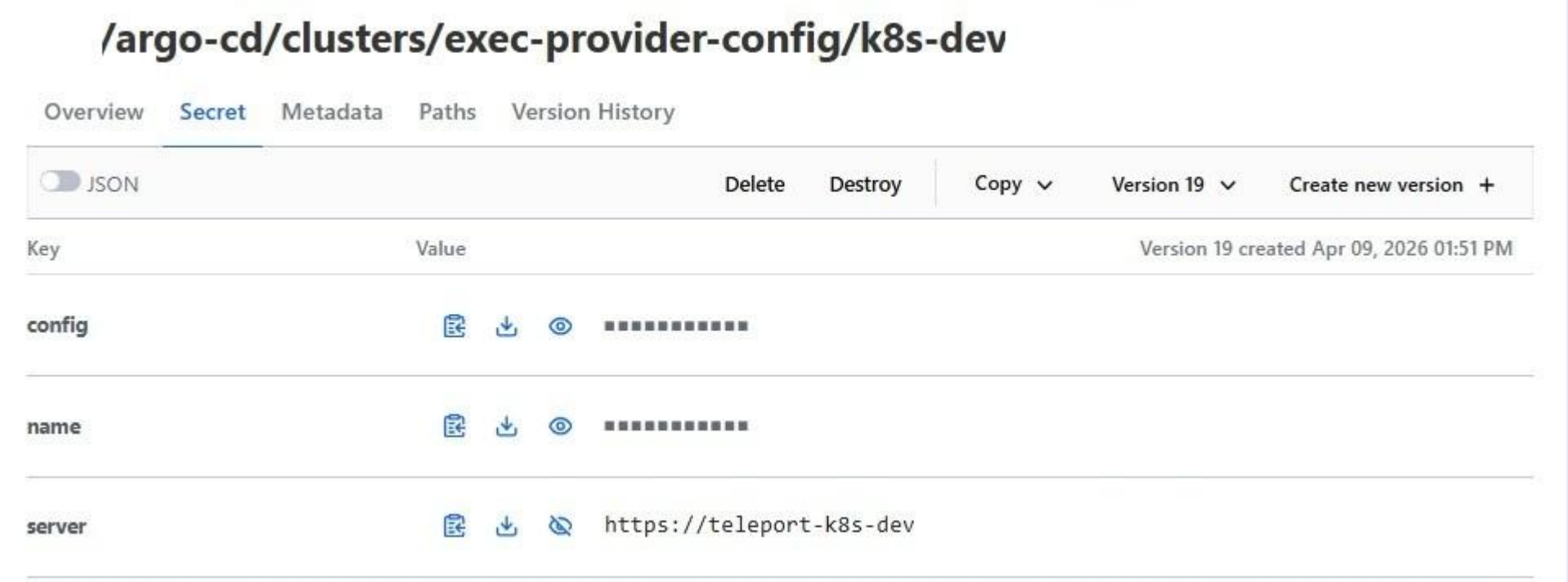
- ArgoCD кэширует кластеры по server URL
  - Один proxy → одинаковый URL → конфликт
  - Кластеры "мигают" и меняются местами

- **Решение: ExternalName Service**

- Один Service на кластер в namespace ArgoCD:
  - teleport-cluster1.argo.svc → teleport.domain
  - teleport-cluster2.argo.svc → teleport.domain
  - Уникальный URL, DNS через k8s, без изменений

Teleport: ротация токенов и один proxy — много кластеров

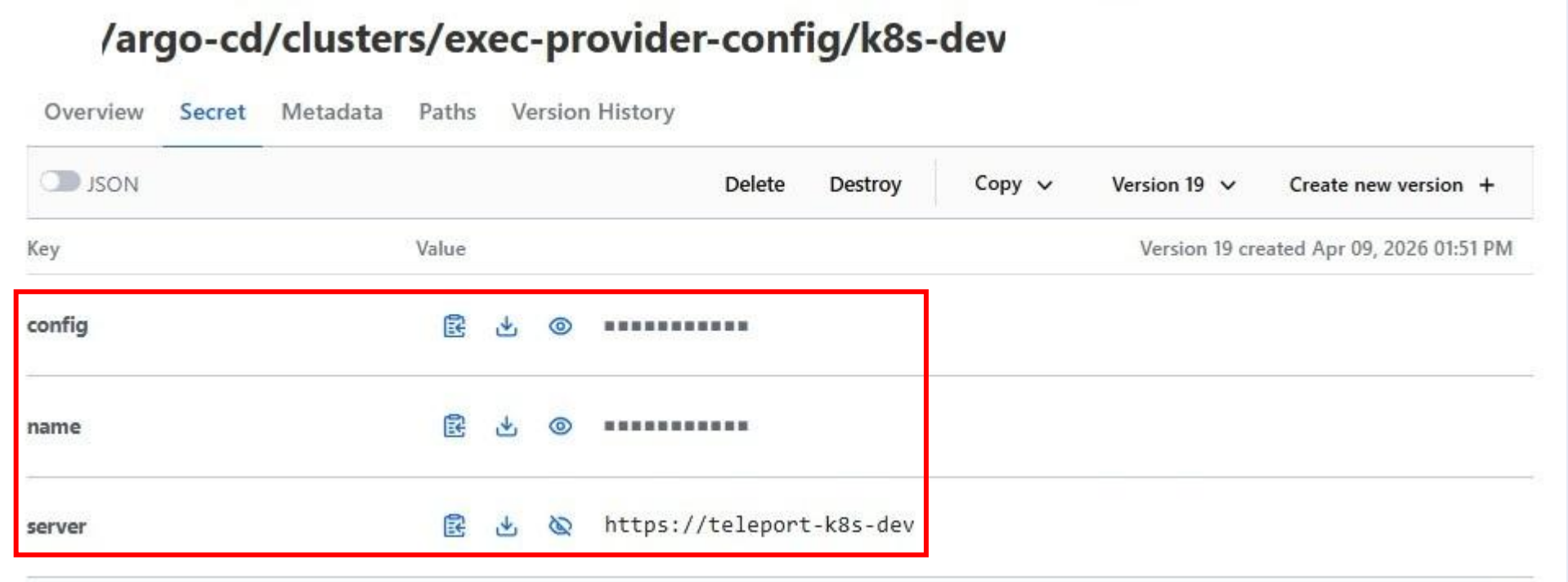
БЕКОН'26




```yaml
apiVersion: v1
kind: Service
metadata:
  name: teleport-k8s-dev
  namespace: k-
spec:
  type: ExternalName
  externalName: k8s

  ports:
    - port: 443
```

Teleport: ротация токенов и один proxy — много кластеров

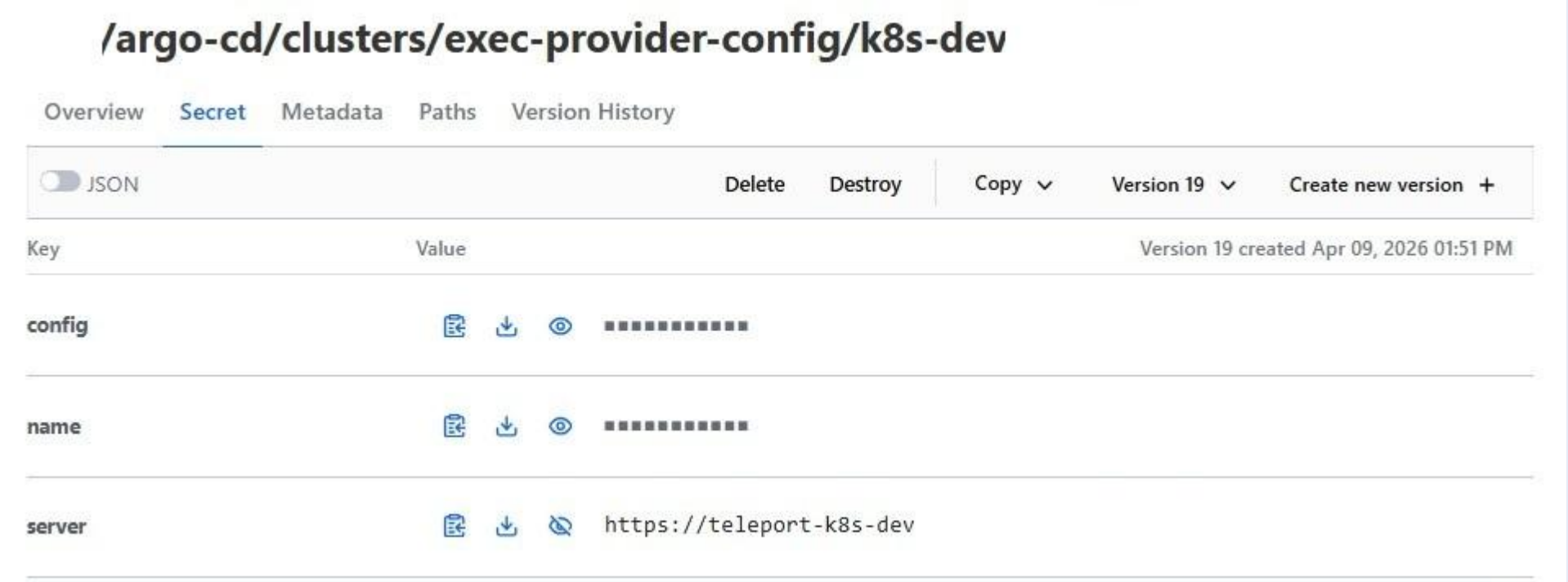
БЕКОН'26




```yaml
apiVersion: v1
kind: Service
metadata:
  name: teleport-k8s-dev
  namespace: k-
spec:
  type: ExternalName
  externalName: k8s

  ports:
    - port: 443
```

Teleport: ротация токенов и один proxy — много кластеров

БЕКОН'26




```yaml
apiVersion: v1
kind: Service
metadata:
  name: teleport-k8s-dev
  namespace: k-
spec:
  type: ExternalName
  externalName: k8s

  ports:
    - port: 443
```

ArgoCD под нагрузкой: hardening values

При большом числе кластеров стандартные настройки не держат:

controller.status.processors — параллельная обработка статусов
controller.operation.processors — параллельные операции sync
controller.repo.server.timeout.seconds — таймаут на получение манифестов
GOMEMLIMIT — жёсткий лимит памяти Go runtime, агрессивный GC
sharding — разбивка кластеров по репликам application-controller

На одном кластере с Gatekeeper: **~350MB RSS** (стабильно, без утечек)
Прогноз на 10 кластеров: **~2.6GB** → с sharding 2 реплики = **1.3GB** каждая

Подробнее — отличный пост по hardening ArgoCD для больших инсталляций по QR

https://akuity.io/blog/argo-cd-application-controller-performance-tuning

А есть ли проблемы?

Ставишь Gatekeeper. Политики применяются. Violations есть.
Хочешь посмотреть их в Grafana — и тут сюрприз:

- **Нет встроенного Prometheus экспортера для нарушений**
- **--log-denies=true пишет в stdout, но в audit логи не попадает**
- **gatekeeper_violations метрика есть, но без constraint_name лейбла**

Решение: opa-scorecard на GitHub — 49 звезд…
Последний коммит — 4 года назад…

Допилили, развернули, работает — опрашивает K8s API каждые 10 сек,
читает .status violations из Constraint объектов, экспортирует в Prometheus

Profit: violations по кластерам и политикам в Grafana в реальном времени

https://github.com/mcelep/opa-scorecard

Принципы, которые работают

БЕКОН'26

- **Единая точка правды**
  Все политики живут в Git: один репозиторий, один источник истины

Принципы, которые работают

БЕКОН'26

- **Единая точка правды**
  Все политики живут в Git: один репозиторий, один источник истины
- **Автоматическая доставка**
  ArgoCD доставляет политики без ручного труда — гарантированно и воспроизводимо

Принципы, которые работают

Единая точка правды

Все политики живут в Git: один репозиторий, один источник истины

Автоматическая доставка

ArgoCD доставляет политики без ручного труда — гарантированно и воспроизводимо

Гранулярность зон

wide / zone / cluster — политика едет туда, куда нужно

Принципы, которые работают

БЕКОН'26

Единая точка правды

Все политики живут в Git: один репозиторий, один источник истины

Автоматическая доставка

ArgoCD доставляет политики без ручного труда — гарантированно и воспроизводимо

Гранулярность зон

wide / zone / cluster — политика едет туда, куда нужно

Секреты централизованы

Vault + External Secrets: не размазаны по кластерам, всё под контролем

Принципы, которые работают

Единая точка правды

Все политики живут в Git: один репозиторий, один источник истины

Автоматическая доставка

ArgoCD доставляет политики без ручного труда — гарантированно и воспроизводимо

Гранулярность зон

wide / zone / cluster — политика едет туда, куда нужно

Секреты централизованы

Vault + External Secrets: не размазаны по кластерам, всё под контролем

Видимость в реальном времени

Victoria Metrics + Grafana — compliance прямо сейчас, а не на квартальном отчёте

Принципы, которые работают

БЕКОН'26

- **Единая точка правды**
  Все политики живут в Git: один репозиторий, один источник истины
- **Автоматическая доставка**
  ArgoCD доставляет политики без ручного труда — гарантированно и воспроизводимо
- **Гранулярность зон**
  wide / zone / cluster — политика едет туда, куда нужно
- **Секреты централизованы**
  Vault + External Secrets: не размазаны по кластерам, всё под контролем
- **Видимость в реальном времени**
  Victoria Metrics + Grafana — compliance прямо сейчас, а не на квартальном отчёте
- **Минимальные права**
  Argo видит только Constraints и Templates — principle of least privilege

БЕКОН'26

КОНФЕРЕНЦИЯ ПО БЕЗОПАСНОСТИ КОНТЕЙНЕРОВ И КОНТЕЙНЕРНЫХ СРЕД

@truebad0ur

aorekhov@swordfishsecurity.ru