LUNTRY

АУДИТ БЕЗОПАСНОСТИ KUBERNETES КЛАСТЕРА БЕЗ KUBERNETES КЛАСТЕРА

Сергей Канибор

Luntry, R&D / Container Security

Обо мне

”
Настоящая защита –
это не запреты, а правильно
расставленные ограничения

R&D / Container Security в Luntry

Более 4 лет опыта
в ИБ

Специализация - Безопасность Контейнеров и Kubernetes

Редактор ТГ-канала
k8s (in) security

Багхантер

- Участник BI.ZONE Bug Bounty и Yandex Bug Bounty
- Автор зарегистрированных CVE и BDU

Спикер

VK Kubernetes
DevOpsConf
OFFZONE
БеКон
PHDays
DevOops

LUNTRY

Agenda

01	Идея Cluster API
02	Безопасность Cluster API
03	Куда смотреть, чтобы было безопасно в будущем

LUNTRY

Предыстория

LUNTRY

Kubernetes Pentest

PEN TESTER

CORE SECURITY
A HelpSystems Company

What my friends think I do

What my mom thinks I do

What society thinks I do

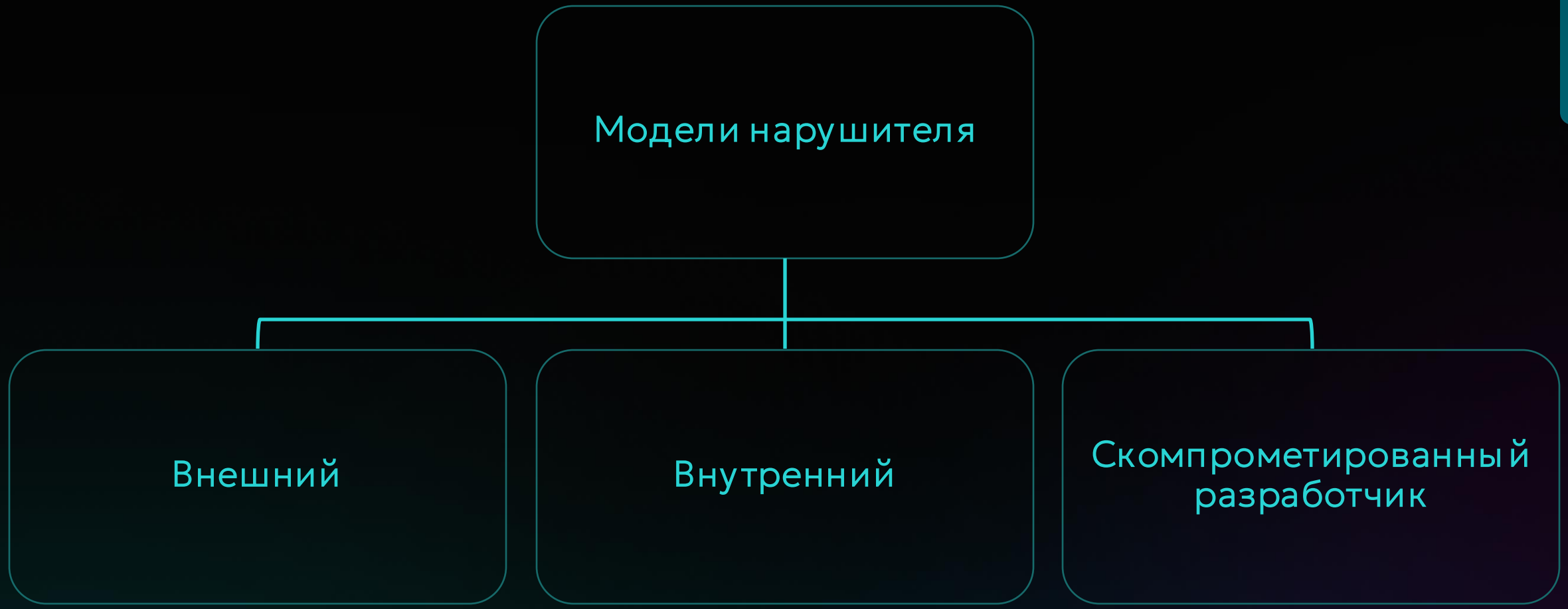
What hackers think I do

What I think I do

What I actually do

LUNTRY

Модели нарушителя



LUNTRY

Что мы проверяем?

```
└─ kubectl auth can-i --list
Resources                                       Non-Resource URLs   Resource Names   Verbs
*.*                                             []                  []              [*]
                                                [*]                 []              [*]
selfsubjectaccessreviews.authorization.k8s.io   []                  []              [create]
selfsubjectrulesreviews.authorization.k8s.io    []                  []              [create]
                                                [/api/*]            []              [get]
                                                [/api]              []              [get]
                                                [/apis/*]           []              [get]
                                                [/apis]             []              [get]
                                                [/healthz]          []              [get]
                                                [/healthz]          []              [get]
                                                [/livez]            []              [get]
                                                [/livez]            []              [get]
                                                [/openapi/*]        []              [get]
                                                [/openapi]          []              [get]
                                                [/readyz]           []              [get]
                                                [/readyz]           []              [get]
                                                [/version/]         []              [get]
                                                [/version/]         []              [get]
                                                [/version]          []              [get]
                                                [/version]          []              [get]
```

- Соответствие CIS K8s
- Политики Policy Engine
- Network Policy
- RBAC
- Эксплуатируем уязвимости рантайм и third-party компонентов

LUNTRY

Что делать с Cluster API?

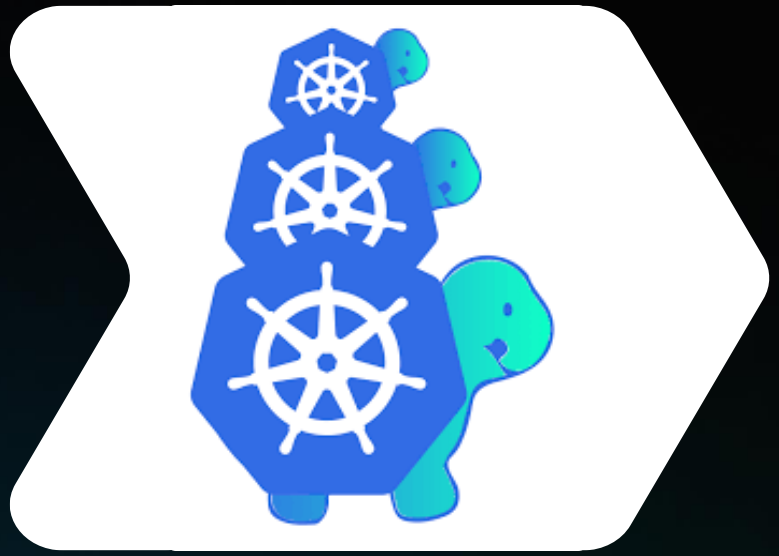
LUNTRY

ИДЕЯ CLUSTER API

Проблемы и их решение

LUNTRY

- Зоопарк кластеров
- Расхождение конфигураций
- Сложность обновлений и масштабирования



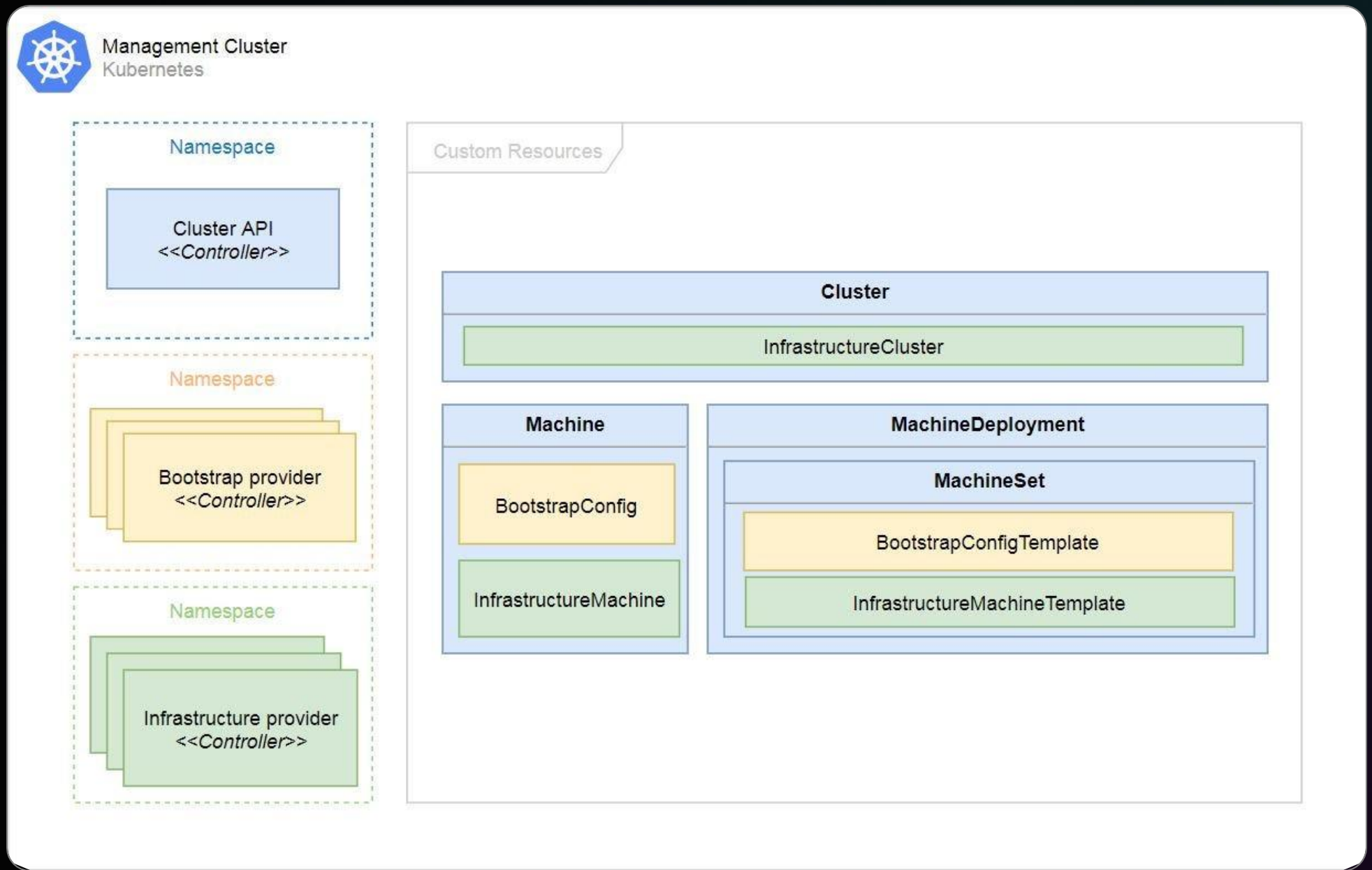
- Интеграции
- Унификация процесса создания k8s кластеров
- Автоматизация развертывания
- Масштабирование и управление
- Единый контроль

CAPI

- Начинался как проект SIG Cluster Lifecycle
- Первый релиз состоялся в июле 2018 года
- 3.7k звезд на GitHub
- Поддерживает десятки различных провайдеров
- Фреймворк для управления жизненным циклом Kubernetes кластеров

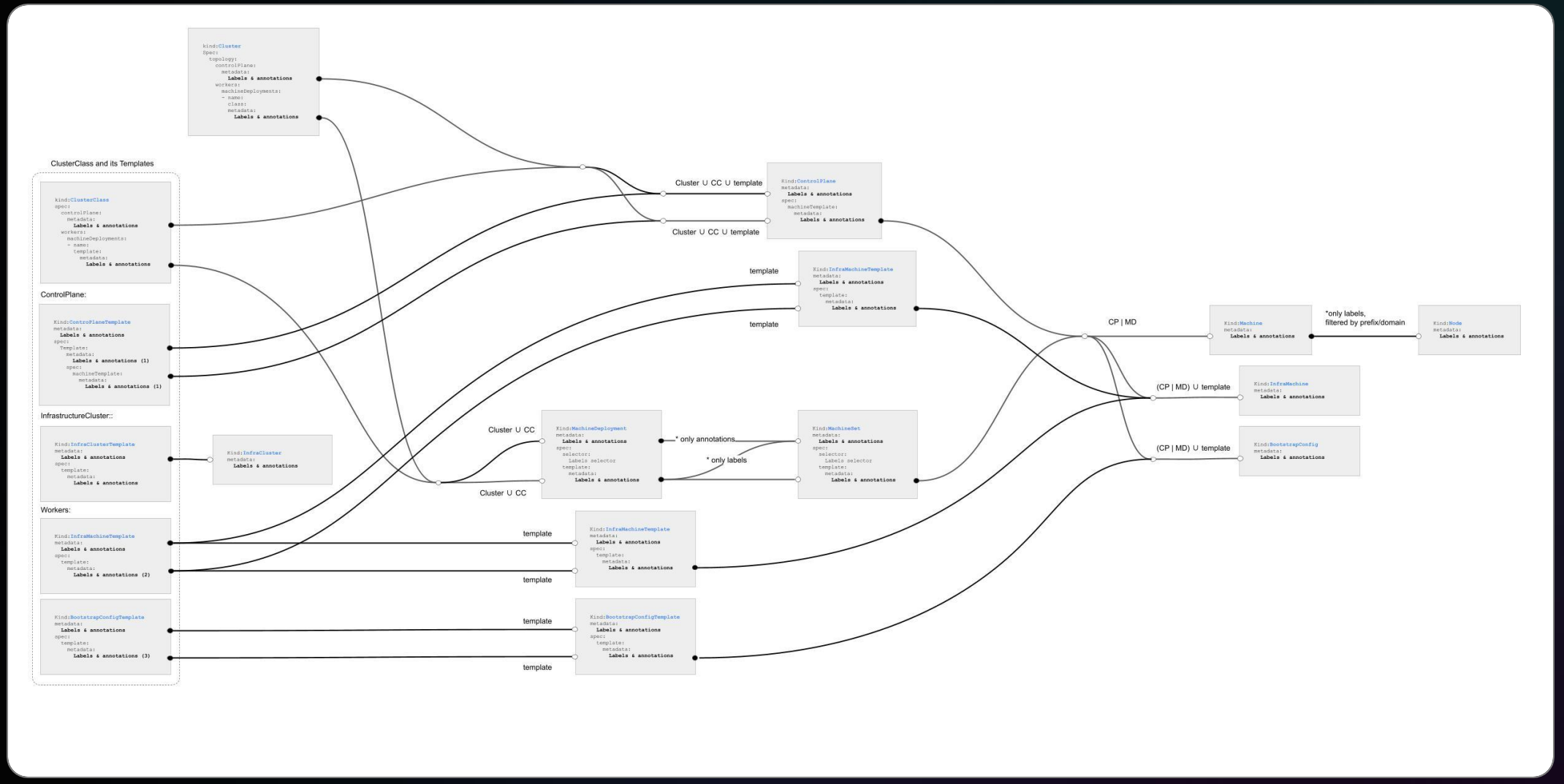
LUNTRY

CAPI манифесты

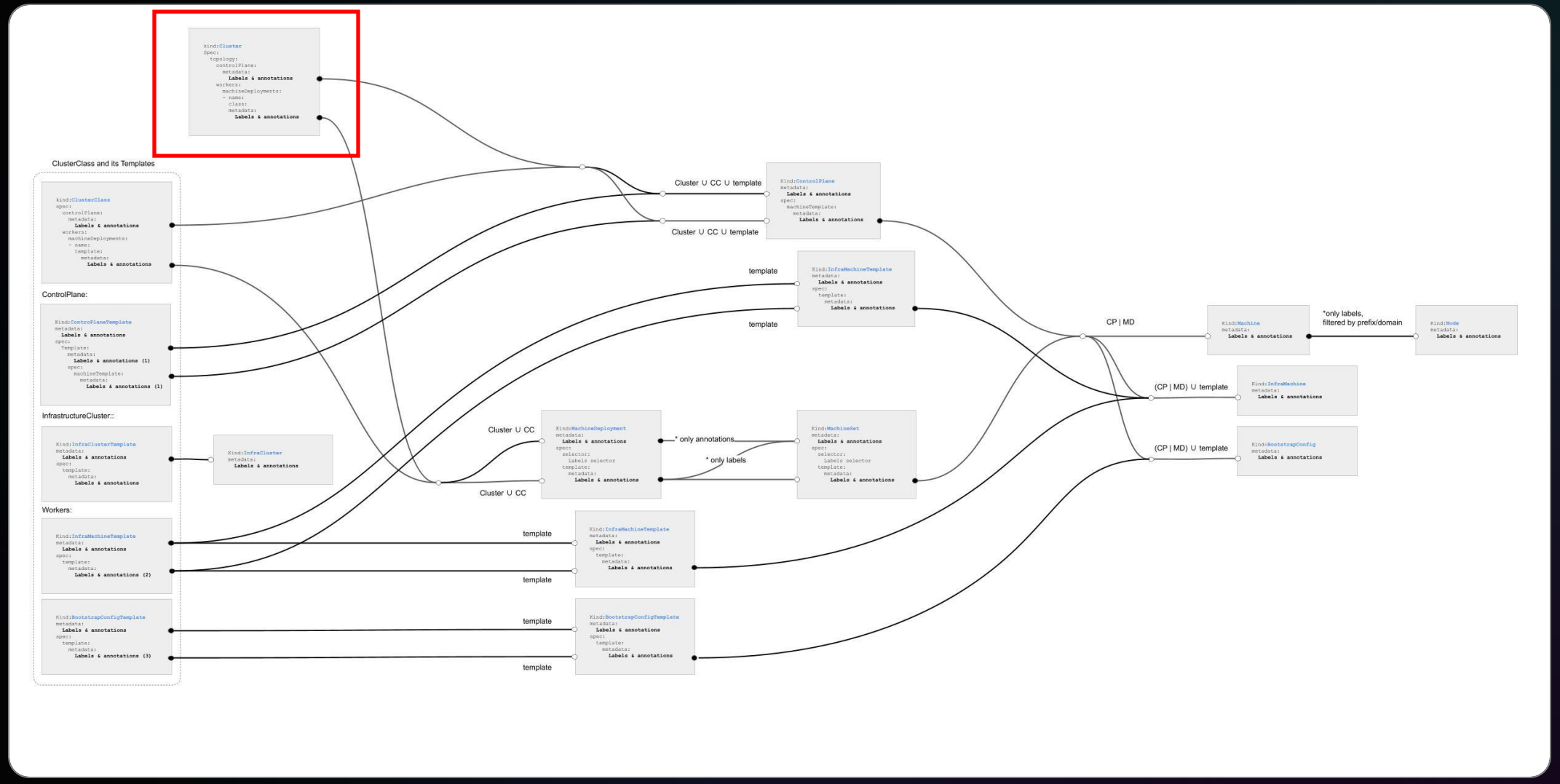


LUNTRY

Metadata propogation



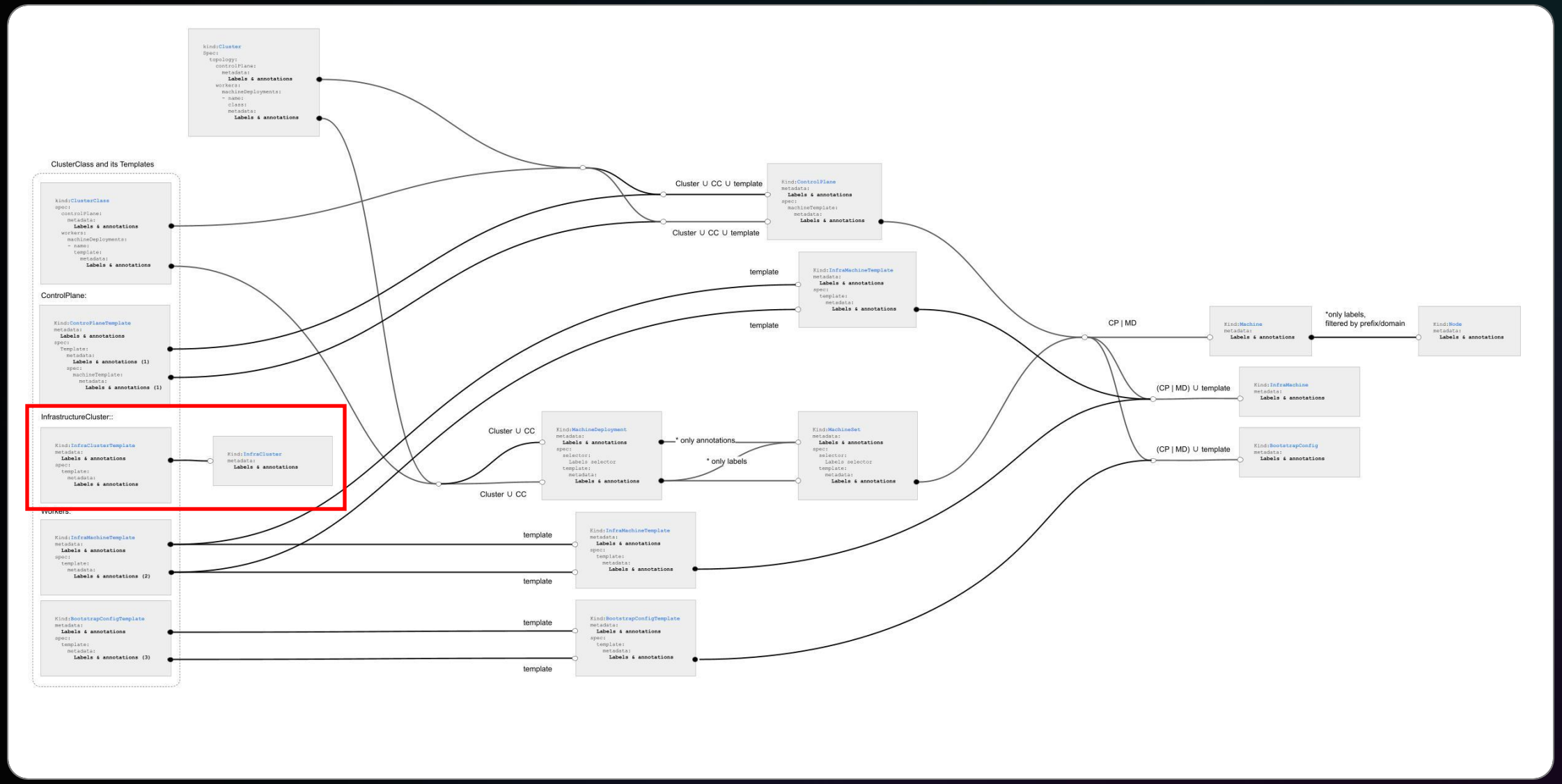
Cluster


ClusterClass and its Templates
ControlPlane:
InfrastructureCluster::
Workers:
Cluster U CC U template
Cluster U CC U template
template
template
Cluster U CC
Cluster U CC
* only annotations
* only labels
template
template
template
template
CP | MD
(CP | MD) U template
(CP | MD) U template
*only labels,
filtered by prefix/domain

Cluster Spec


```
spec:
  clusterNetwork:
    pods:
      cidrBlocks: ["192.168.0.0/16"]
    services:
      cidrBlocks: ["10.96.0.0/12"]
  infrastructureRef:
    apiVersion: infrastructure.cluster.x-k8s.io/v1beta1
    kind: AWSCluster
    name: my-cluster
  controlPlaneRef:
    apiVersion: controlplane.cluster.x-k8s.io/v1beta1
    kind: KubeadmControlPlane
    name: my-cluster-control-plane
```

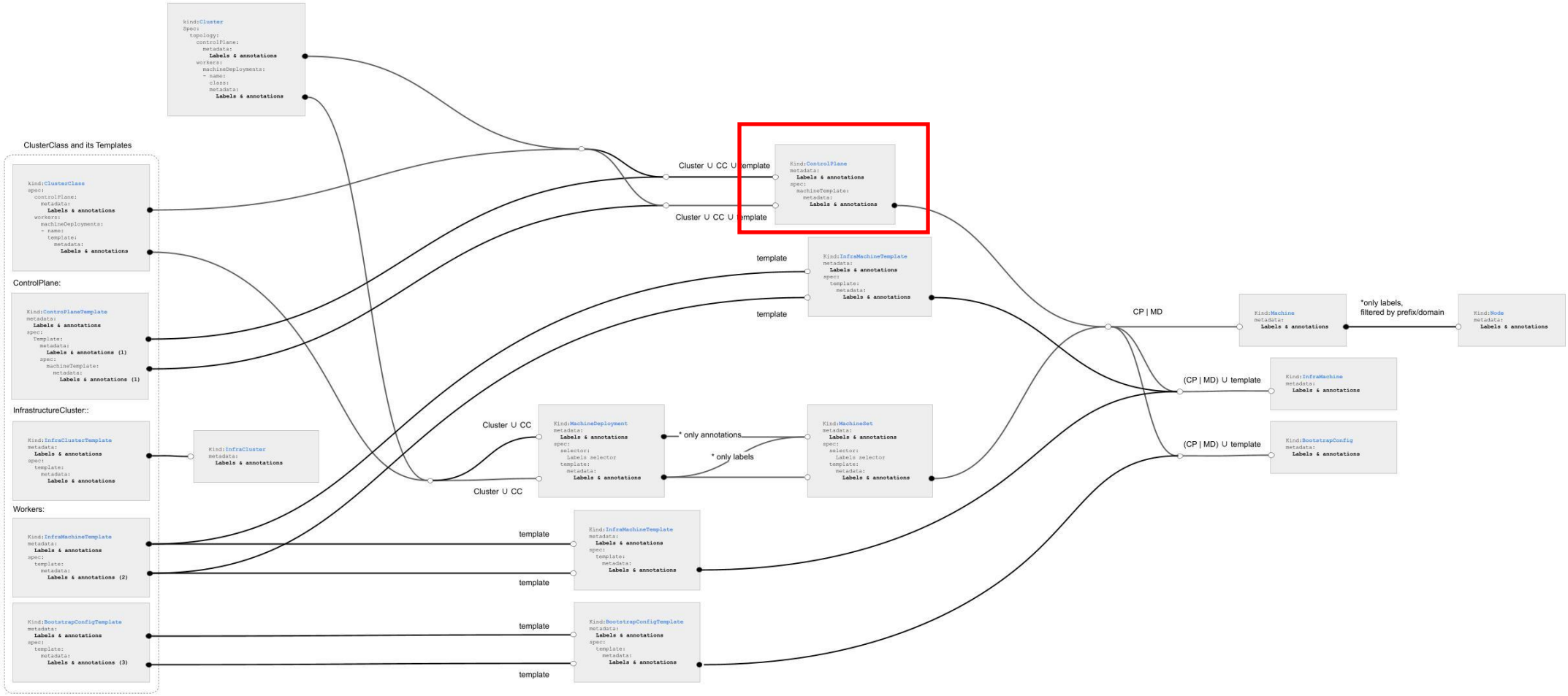
InfraCluster



InfraCluster Spec


```
spec:
  region: us-west-2
  network:
    vpc:
      id: "vpc-0abcd1234efgh5678"
    subnets:
      - id: "subnet-0abcd1234efgh5678"
        role: public
      - id: "subnet-0abcd1234efgh5679"
        role: private
  controlPlaneEndpoint:
    host: "my-cluster-api.example.com"
    port: 6443
```

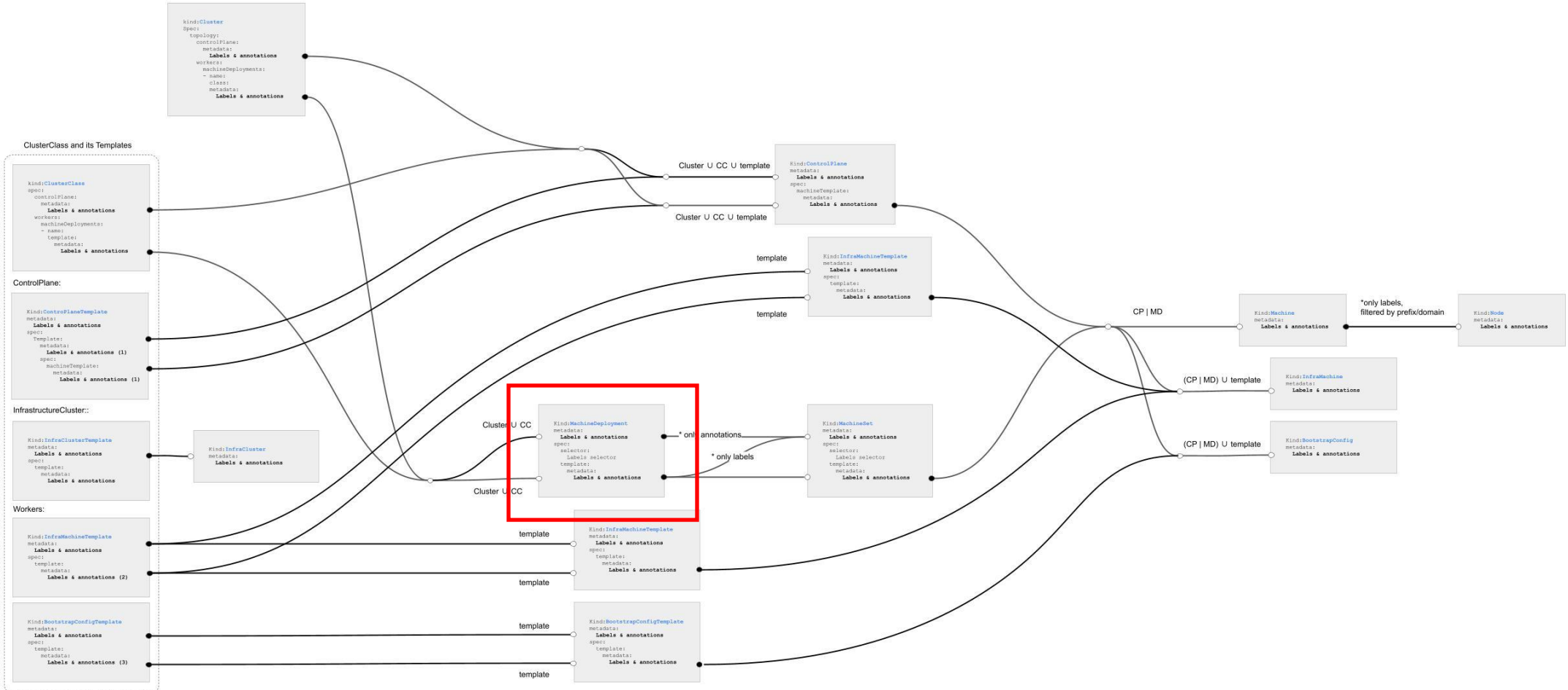
KubeadmControlPlane



KubeadmControlPlane Spec


```
spec:
  replicas: 3
  version: v1.27.3
  kubeadmConfigSpec:
    clusterConfiguration:
      apiServer:
        extraArgs:
          enable-admission-plugins: "NodeRestriction"
infrastructureRef:
      apiVersion: infrastructure.cluster.x-
k8s.io/v1beta1
      kind: AWSMachineTemplate
      name: my-control-plane-machine-template
```

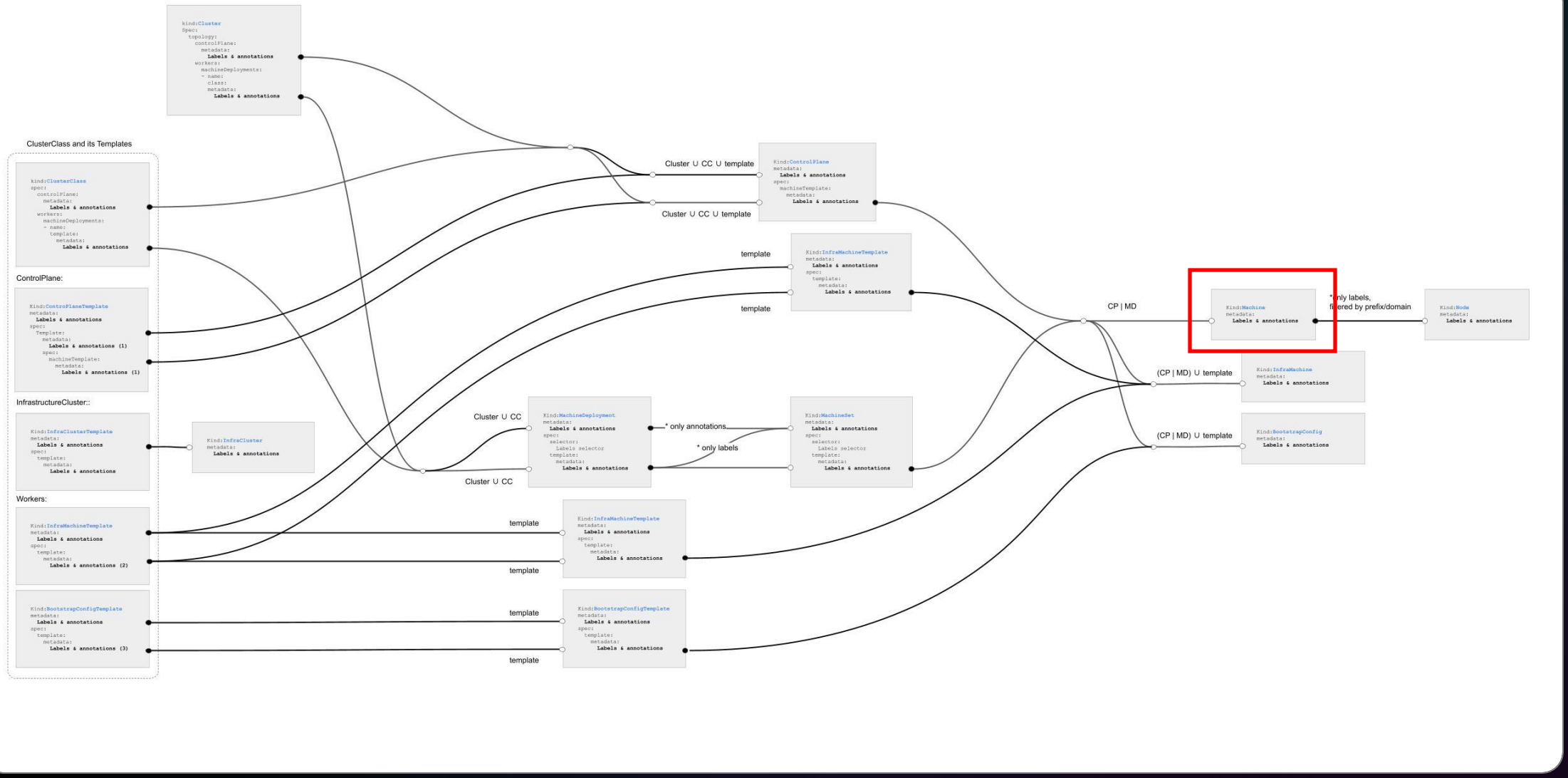
MachineDeployment



MachineDeployment Spec


```
spec:
      version: v1.27.3
      bootstrap:
        configRef:
          apiVersion: bootstrap.cluster.x-
k8s.io/v1beta1
          kind: KubeadmConfigTemplate
          name: my-cluster-md-0-bootstrap
      infrastructureRef:
        apiVersion: infrastructure.cluster.x-
k8s.io/v1beta1
        kind: AWSMachineTemplate
        name: my-cluster-md-0-machine-template
```

Machine


ClusterClass and its Templates
ControlPlane:
InfrastructureCluster::
Workers:
Cluster U CC U template
Cluster U CC U template
template
template
Cluster U CC
Cluster U CC
* only annotations
* only labels
template
template
template
template
CP | MD
(CP | MD) U template
(CP | MD) U template
* only labels,
filtered by prefix/domain

Machine Spec


```
spec:
  clusterName: my-cluster
  version: v1.27.3
  bootstrap:
    configRef:
      apiVersion: bootstrap.cluster.x-k8s.io/v1beta1
      kind: KubeadmConfig
      name: my-cluster-machine-1-bootstrap
  infrastructureRef:
    apiVersion: infrastructure.cluster.x-
k8s.io/v1beta1
    kind: AWSMachine
    name: my-cluster-machine-1
```

БЕЗОПАСНОСТЬ CAPI

Стандарты безопасности

01	CIS Kubernetes Benchmark
02	NSA/CISA Kubernetes Hardening Guide
03	Kubernetes Security Technical Implementation Guide (STIG)
04	PCI Security Standards Council
05	NIST Special Publication 800-190
06	Приказ ФСТЭК России №118

LUNTRY

Стандарты безопасности

01	Компоненты Control Plane
02	ETCD
03	Конфигурации Control Plane
04	Worker Nodes
05	Policies

*порядка 130 проверок

LUNTRY

Компоненты Control Plane

LUNTRY

01 Конфигурационные файлы узлов Control Plane

02 API Server

03 Controller Manager

04 Scheduler

1.2.1 Ensure that the --anonymous-auth argument is set to false (Manual).....
1.2.2 Ensure that the --token-auth-file parameter is not set (Automated).....
1.2.3 Ensure that the DenyServiceExternalIPs is set (Manual) .....
1.2.4 Ensure that the --kubelet-client-certificate and --kubelet-client-key argumer
appropriate (Automated).....

ETCD

LUNTRY

2 etcd.....

2.1 Ensure that the --cert-file and --key-file arguments are set as appropriate (Automatec
2.2 Ensure that the --client-cert-auth argument is set to true (Automated).....
2.3 Ensure that the --auto-tls argument is not set to true (Automated).....
2.4 Ensure that the --peer-cert-file and --peer-key-file arguments are set as appropriate (Automated) .....
2.5 Ensure that the --peer-client-cert-auth argument is set to true (Automated) .....
2.6 Ensure that the --peer-auto-tls argument is not set to true (Automated) .....
2.7 Ensure that a unique Certificate Authority is used for etcd (Manual).....

Конфигурации Control Plane

01 Аутентификация и авторизация

02 Логирование/аудит

LUNTRY

Worker Nodes

01 Конфигурационные файлы узлов Worker Nodes

02 Конфигурация kubelet

4.2.1 Ensure that the --anonymous-auth argument is set to false (Automated) .....
4.2.2 Ensure that the --authorization-mode argument is not set to AlwaysAllow (Automated) .....
4.2.3 Ensure that the --client-ca-file argument is set as appropriate (Automated) .....
4.2.4 Verify that the --read-only-port argument is set to 0 (Manual) .....
4.2.5 Ensure that the --streaming-connection-idle-timeout argument is not set to 0 (Manual) .....
4.2.6 Ensure that the --make-iptables-util-chains argument is set to true (Automated) .....
4.2.7 Ensure that the --hostname-override argument is not set (Manual) .....

Policies

01 RBAC и Service Account

02 Pod Security Standards

03 Network Policies и CNI

04 Управление секретами

05 Сторонний Admission Controller

06 Общие политики

LUNTRY

Чем проверять?

LUNTRY

01	Kube-bench
02	Скрипты
03	Что-то своё

Инструменты

LUNTRY

Минусы

- Внутренний регламент (может быть что угодно)
- Для полного покрытия нужно писать что-то своё

Плюсы

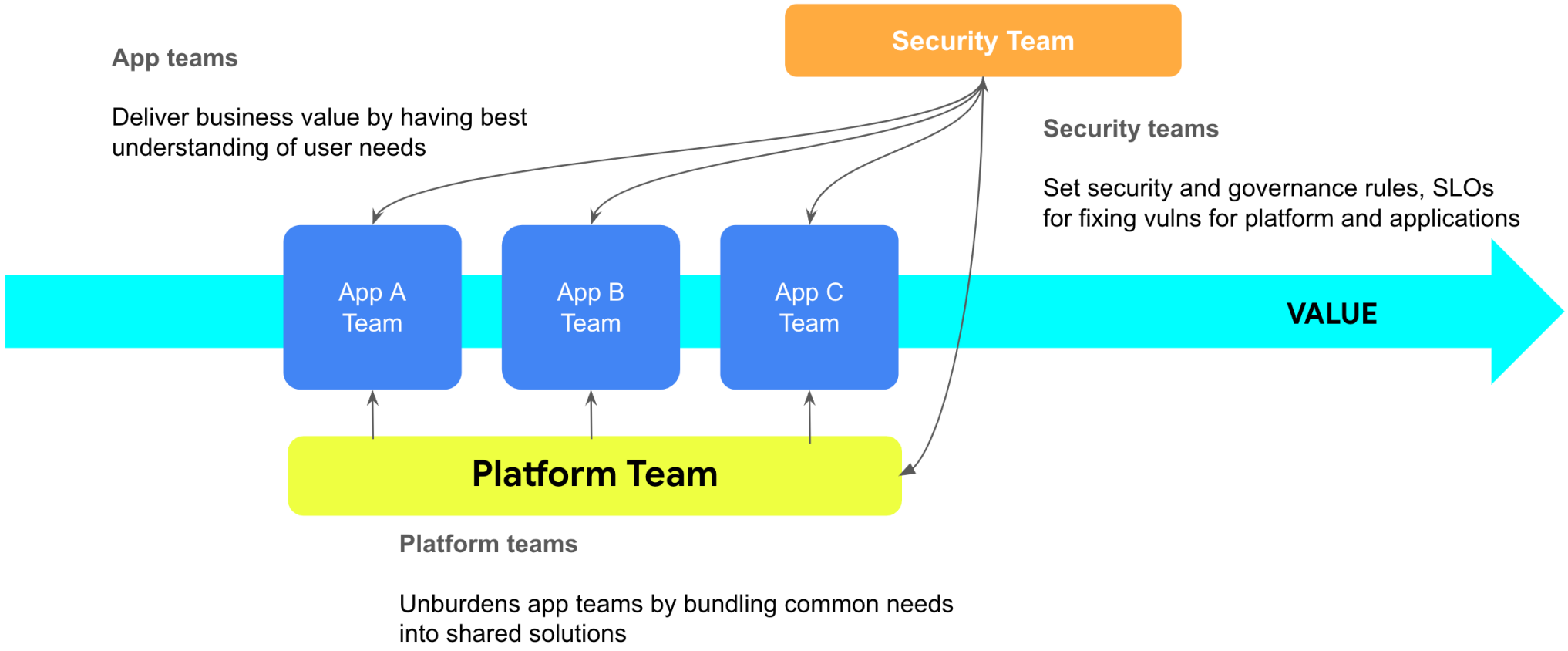
- Open Source

Куда смотреть, чтобы было безопасно в будущем

LUNTRY

КУДА СМОТРЕТЬ, ЧТОБЫ БЫЛО БЕЗОПАСНО В БУДУЩЕМ

Shift Down Security



LUNTRY

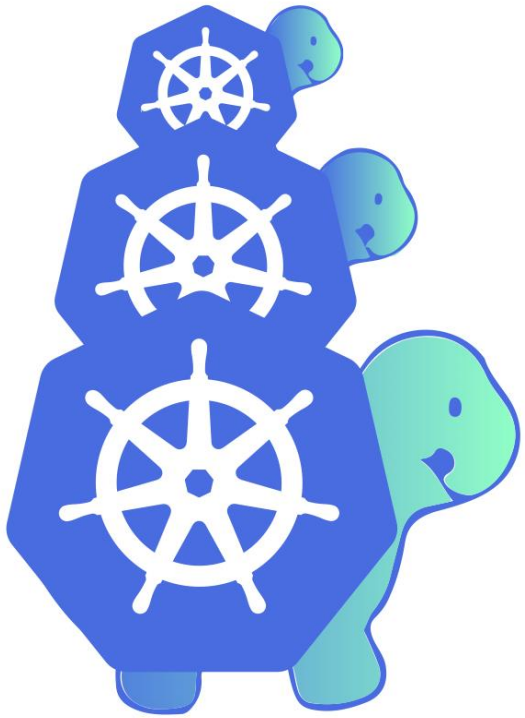
Всё есть YAML

LUNTRY

01	Kubernetes – декларативная система
02	Любой ресурс в Kubernetes – это YAML
03	CAPI манифесты – также набор YAML’ов

Уже понимаете, к чему я?

Используем Policy Engine



Kyverno

LUNTRY

Для чего может быть полезно

- Проверить кластера еще на этапе провижининга
- Удешевить и ускорить исправление мисконфигов

Что можно проверить

- Флаги компонентов Control & Data Plane
- Версии ядра / ОС / container runtime

Что нельзя проверить

- Содержимое YAML манифестов
- securityContext
- Network Policy

Примеры

```
rules:
    - name: validate-kubeadmcontrolplane-secure-params
      match:
        any:
          - resources:
              kinds:
                - KubeadmControlPlane.controlplane.cluster.x-k8s.io
      validate:
        message: "KubeadmControlPlane must follow CIS Kubernetes Benchmark"
        pattern:
          spec:
            kubeadmConfigSpec:
              clusterConfiguration:
                apiServer:
                  extraArgs:
                    audit-policy-file: "/etc/kubernetes/config/audit-policy.yaml",
                    authorization-mode: "RBAC,Node",
                    bind-address: "127.0.0.1",
                    enable-admission-plugins: "NamespaceLifecycle,NodeRestriction"
```

LUNTRY

Примеры

```
rules:
    - name: validate-kubeadmcontrolplane-secure-params
      match:
        any:
          - resources:
              kinds:
                - KubeadmControlPlane.controlplane.cluster.x-k8s.io
      validate:
        message: "KubeadmControlPlane must follow CIS Kubernetes Benchmark"
        pattern:
          spec:
            kubeadmConfigSpec:
              files:
              - content: |
                  authentication:
                    anonymous:
                      enabled: false
                    webhook:
                      cacheTTL: 10s
                      enabled: true
                    x509:
                      clientCAFile: /etc/kubernetes/pki/ca.crt
                  authorization:
                    mode: AlwaysAllow
```

LUNTRY

Итоги

CAPI / CIS K8s Benchmark	1	2	3	4	5
KubeadmControlPlane	59/59	7/7	5/5		0/35
KubeadmConfigTemplate				24/24	0/35

Порядка 73% процентов проверок CIS K8S Benchmark можно выполнить без существования кластера!

Вместо заключения

01 Kubernetes в каждый дом

02 Никогда не забывать про безопасность K8s

03 Можно сделать кластер безопасным, даже когда его не существует

LUNTRY

luntry_official
luntrysolution
luntrysolution

luntry.ru
info@luntry.ru

СЕРГЕЙ КАНИБОР
R&D / Container Security Luntry

sk@luntry.ru
@r0binak

СПАСИБО ЗА ВНИМАНИЕ