

Disaster recovery Kubernetes

An abstract graphic featuring a complex network of orange nodes and lines, resembling a molecular structure or a data network, set against a dark blue background with bokeh light effects. The nodes are of varying sizes and are interconnected by thin, glowing orange lines. The overall composition is dynamic and tech-oriented.

Привет! Kubernetes не
ломается!



Пока!



Directed by
ROBERT B. WEIDE


Каков наш план?

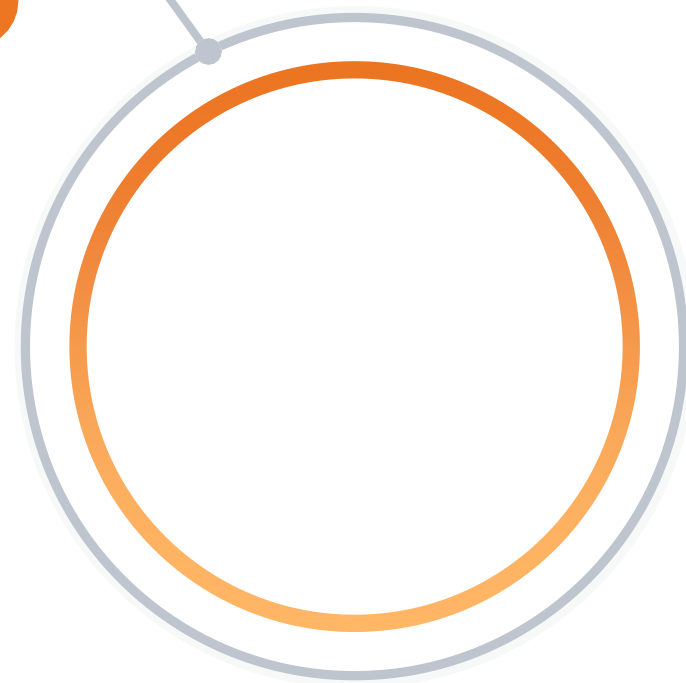
- Выясним как k8s может сломаться
- Структурируем проблемы по нему
- Соберем чек-лист на что требуется обратить внимание
- Разберем каждый из пунктов, разбавив их кейсами
- Сделаем выводы

■ Как можно сломать кubernetes?





Как можно сломать кубернетес?

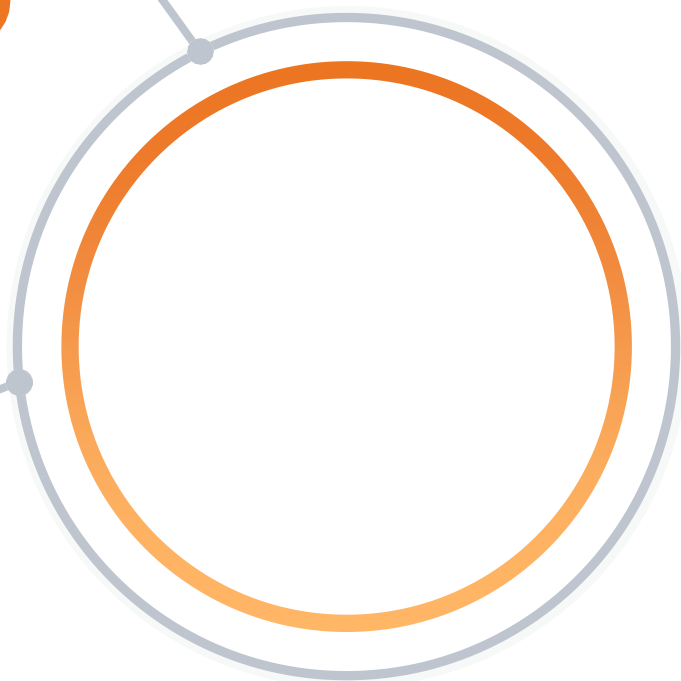
Серверы с k8s сгорели, 
уборщица выдернула
кабель



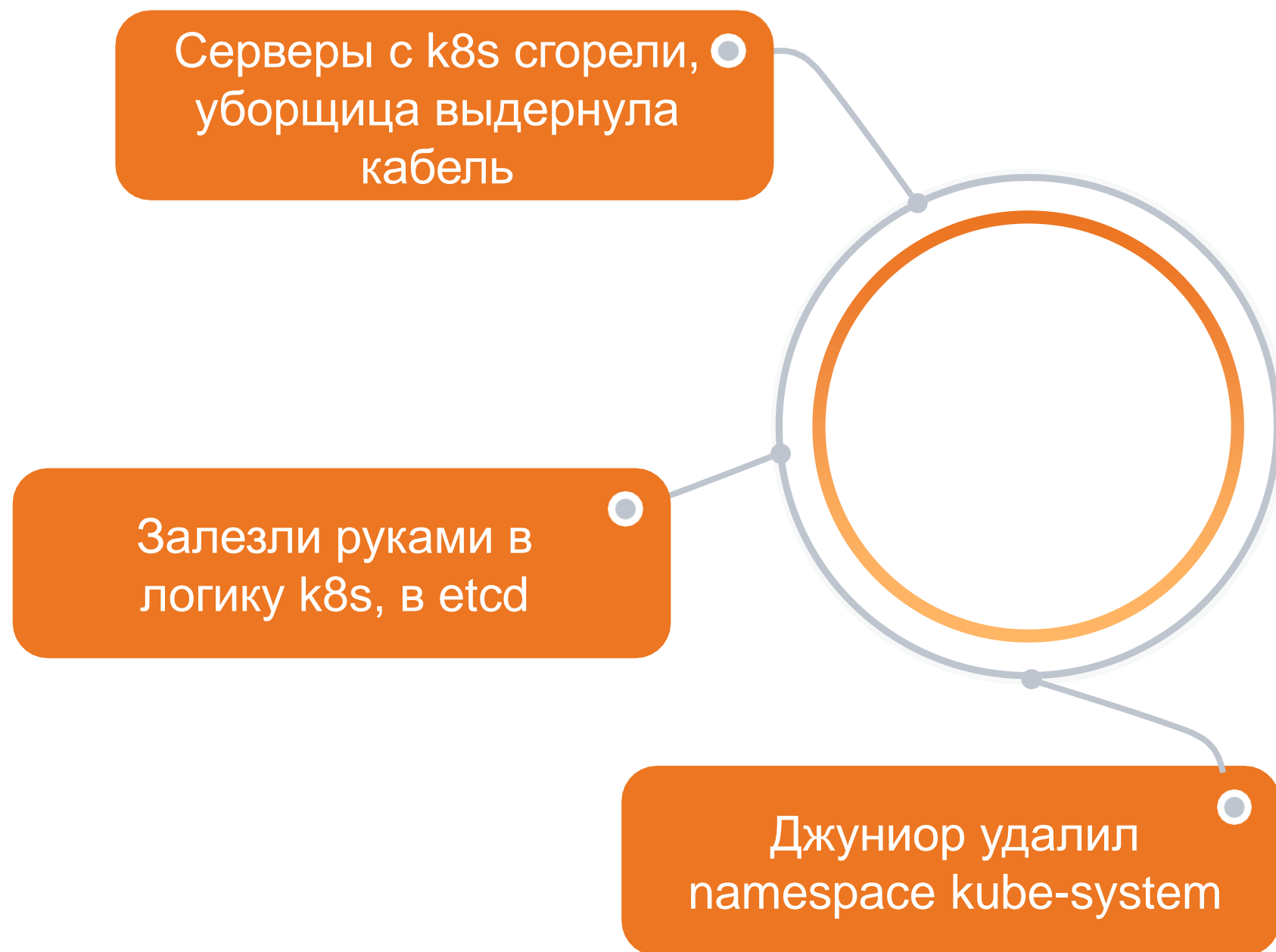
Как можно сломать кубернетес?

Серверы с k8s сгорели, 
уборщица выдернула
кабель

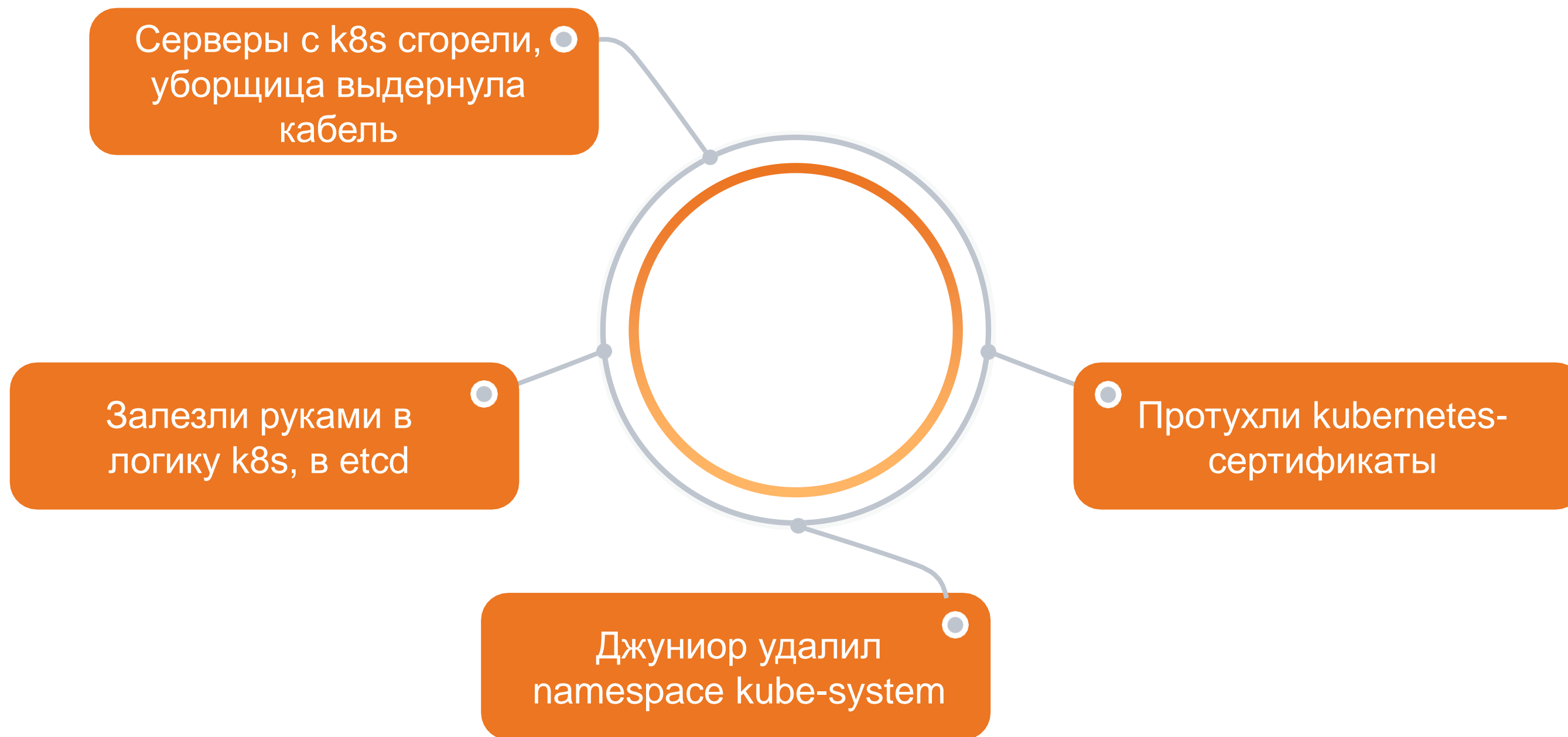
 Залезли руками в
логику k8s, в etcd



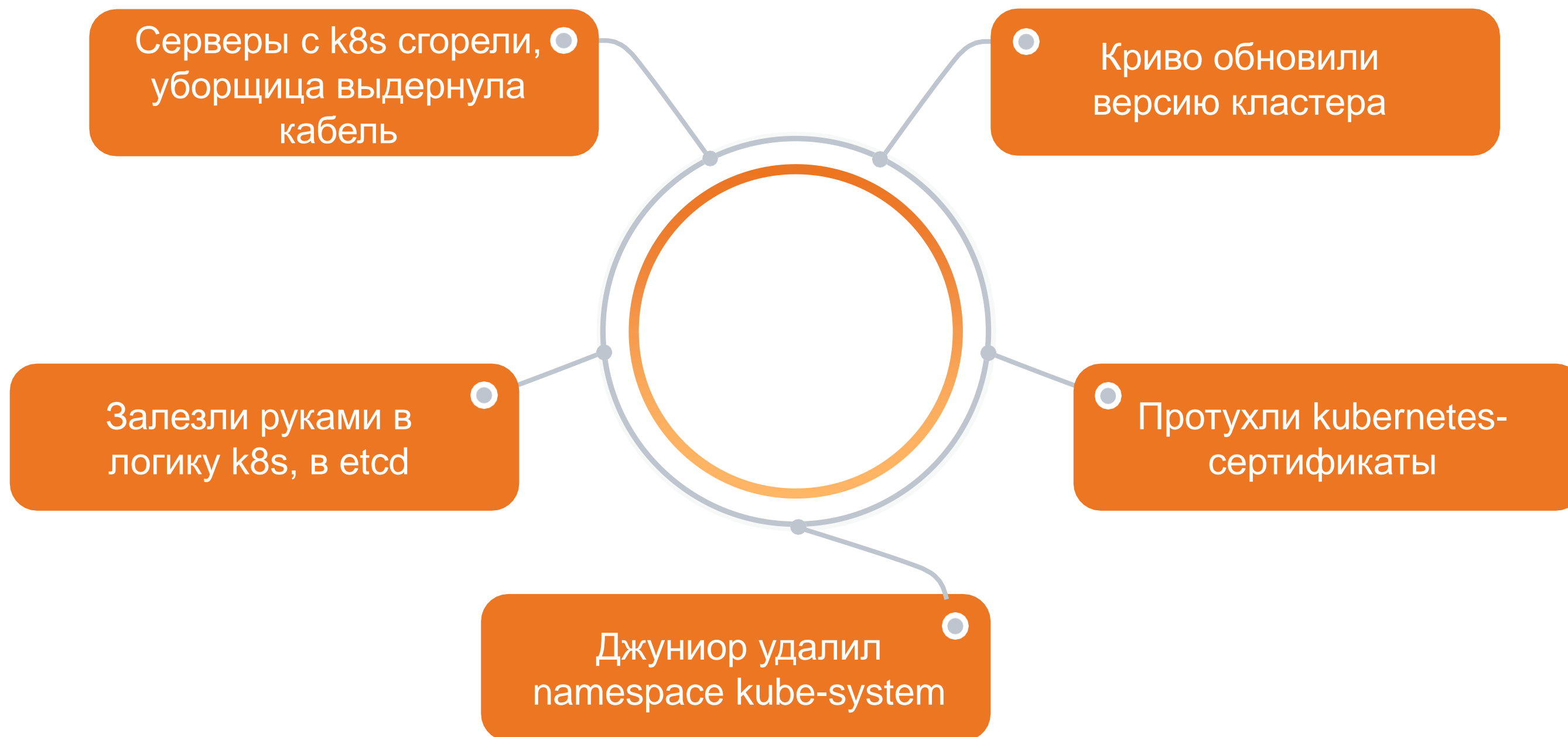
Как можно сломать кубернетес?



Как можно сломать кубернетес?



Как можно сломать кубернетес?



■ Структурируем проблемы:

- «Физический» уровень

Структурируем проблемы:

- «Физический» уровень
 - Качество услуг ДЦ
 - Правильность подбора железа (нагрузка)
 - Отсутствие извращений в архитектуре

CAPTAIN OBVIOUS



■ Структурируем проблемы:

- «Физический» уровень
- Организационный уровень

■ Структурируем проблемы:

- «Физический» уровень
- Организационный уровень
 - Отсутствие описанных процессов
 - Бардак в топологии и в проекте
 - Нет разграничений прав и ответственных
 - Постоянная работа в режиме аврала



■ Структурируем проблемы:

- «Физический» уровень
- Организационный уровень
- Прикладной уровень

■ Структурируем проблемы:

- Прикладной уровень:
 - Недостаточно компетенций

■ Структурируем проблемы:

- Прикладной уровень:
 - Недостаточно компетенций
 - Надежда на русский авось (бекапы, dev-среды)



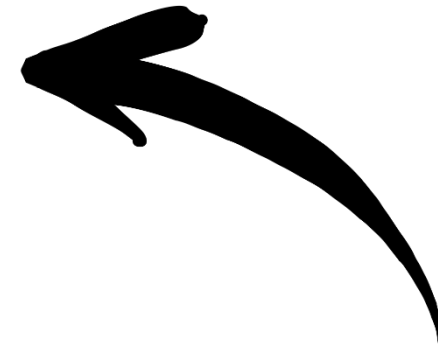
■ Структурируем проблемы:

- Прикладной уровень:
 - Недостаточно компетенций
 - Надежда на русский авось (бекапы, dev-среды)
 - Человеческий фактор



■ Структурируем проблемы:

- «Физический» уровень
- Организационный уровень
- Прикладной уровень



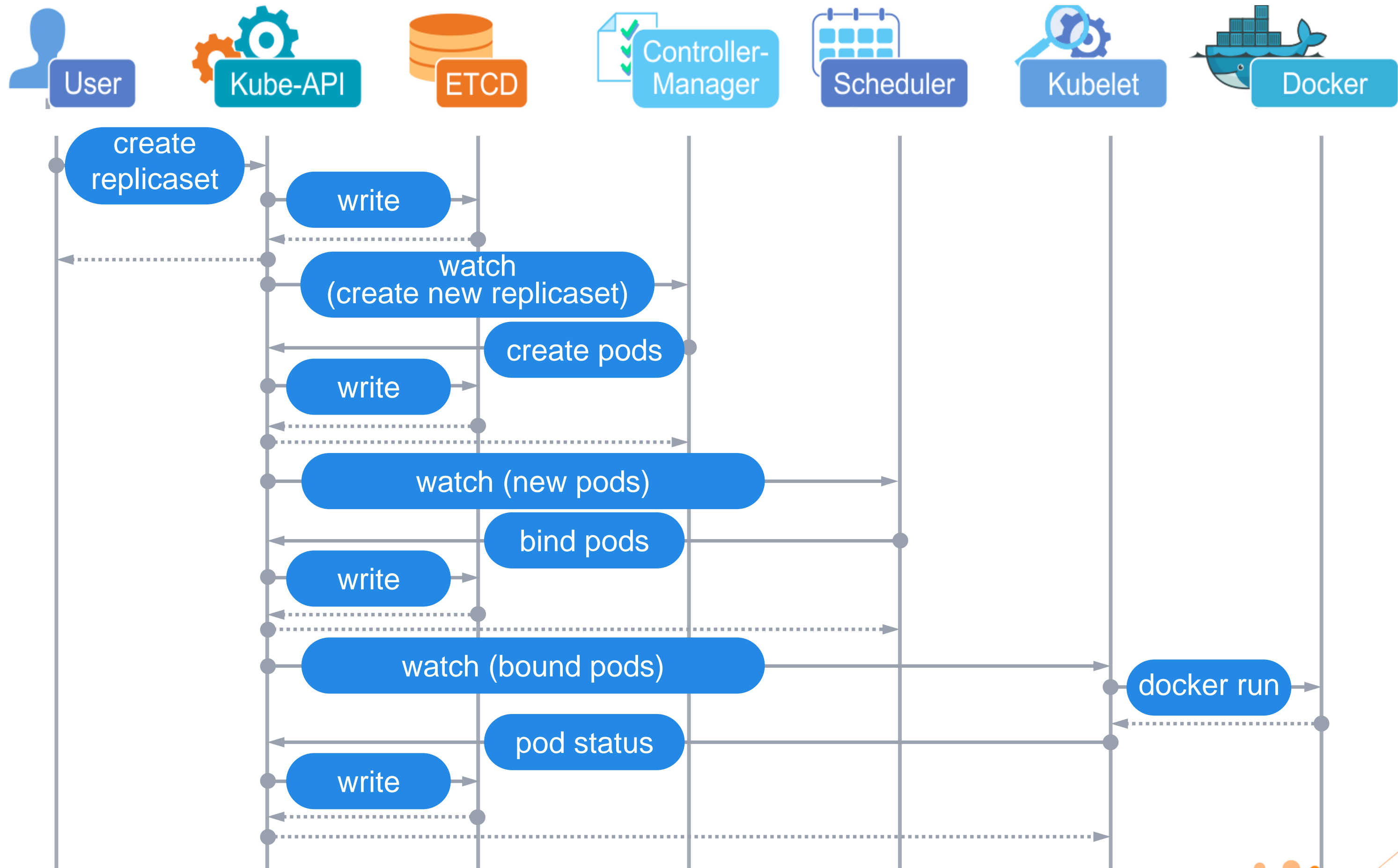
We are here



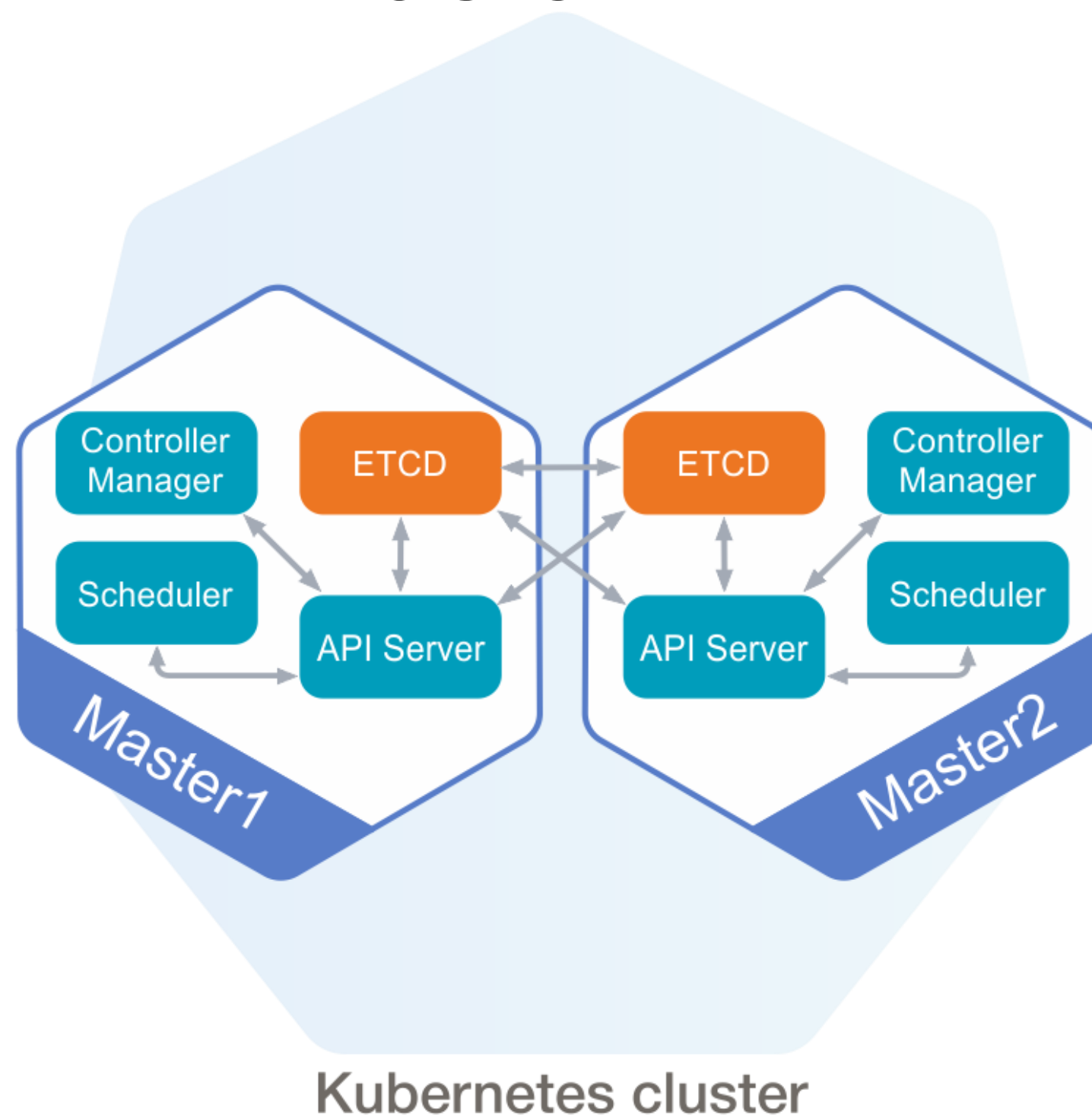
■ На что нужно обратить внимание:

- Отказоустойчивый сетяп Kubernetes
- Резервирование Ingress-controller'a
- Сертификаты кластера
- Регламентные работы (обновление кластера или ПО и т.д.)
- Бэкапы
- Расширенные инструменты k8s (PSP, PDB, RQ...)

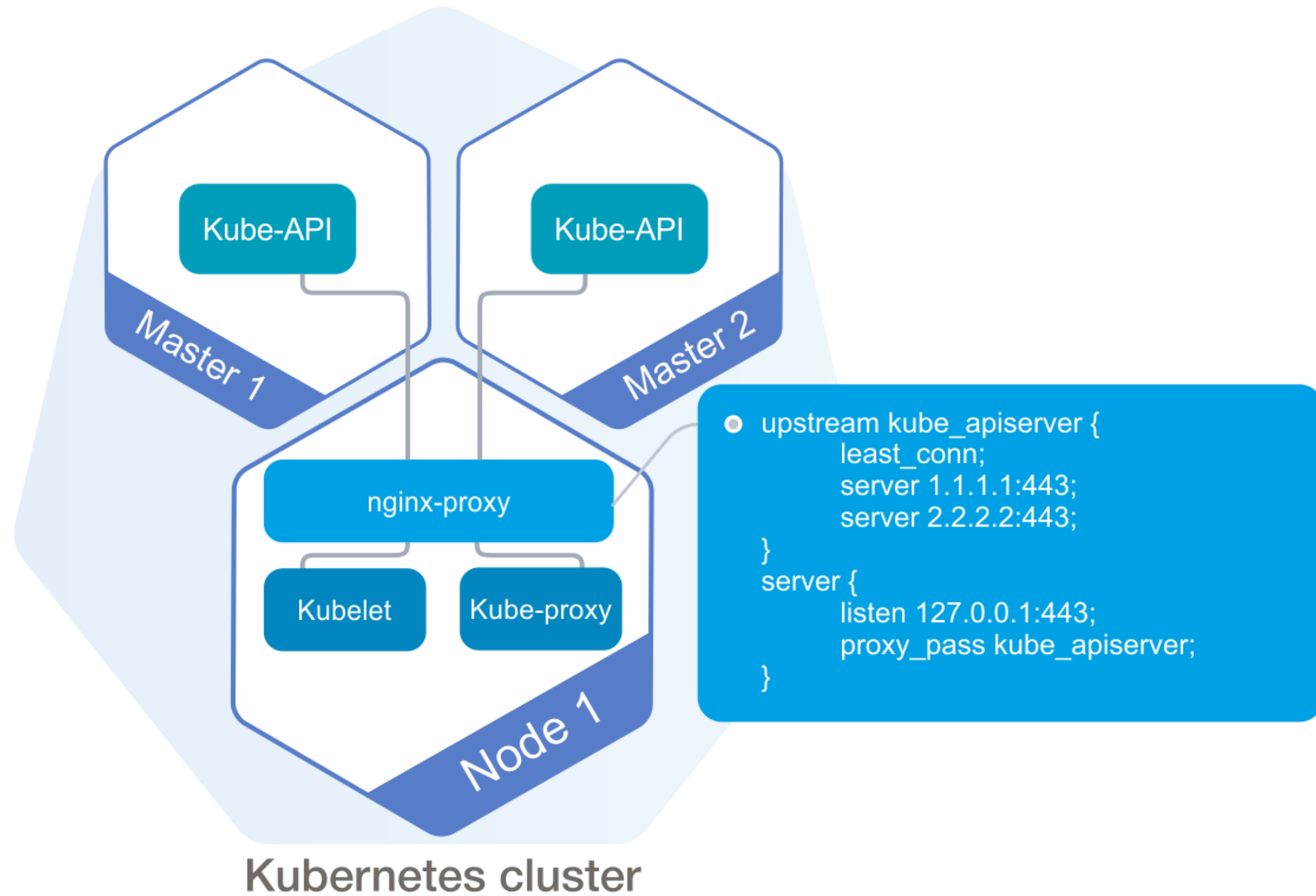




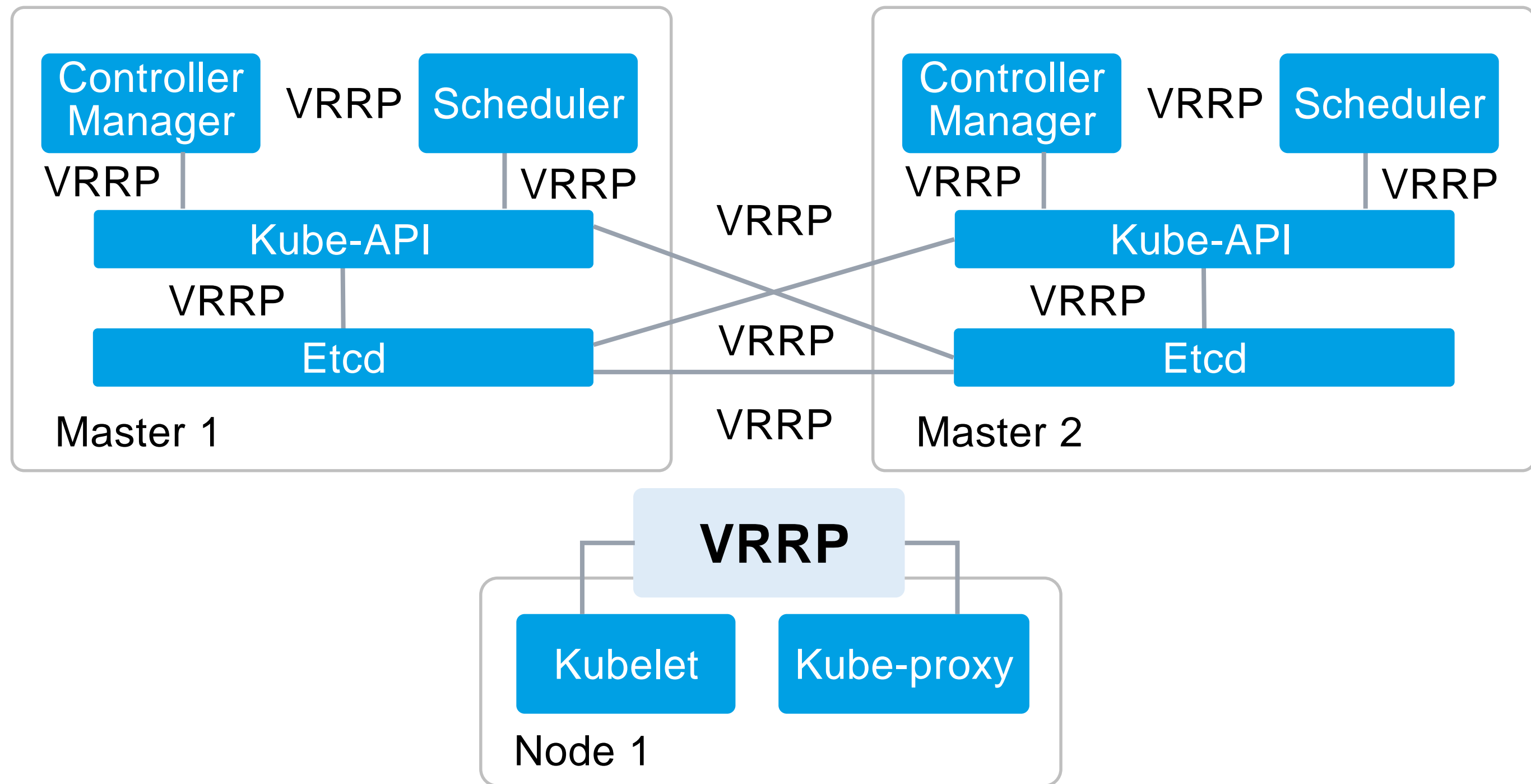
Отказоустойчивый master:



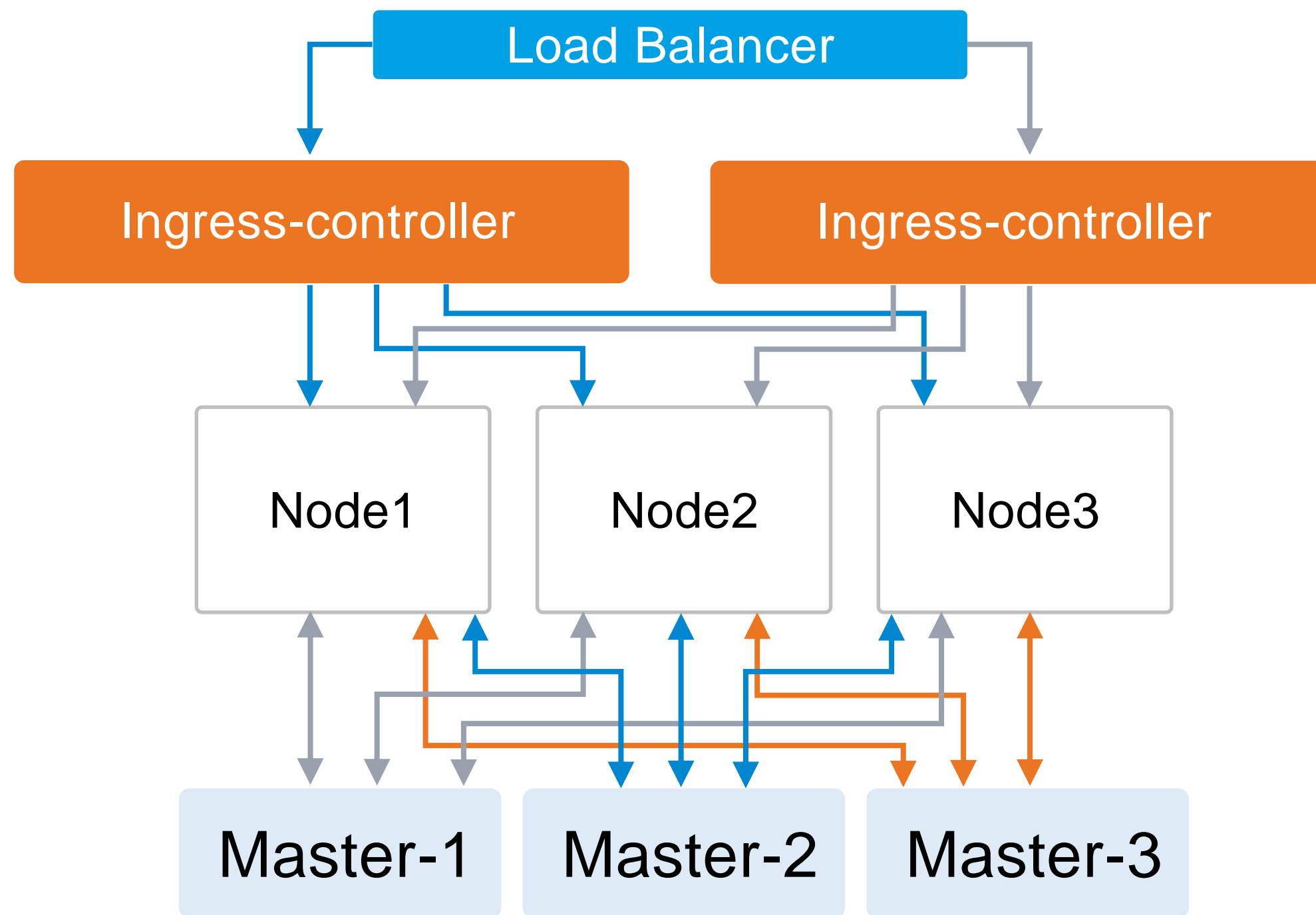
Взаимодействие master<->node:



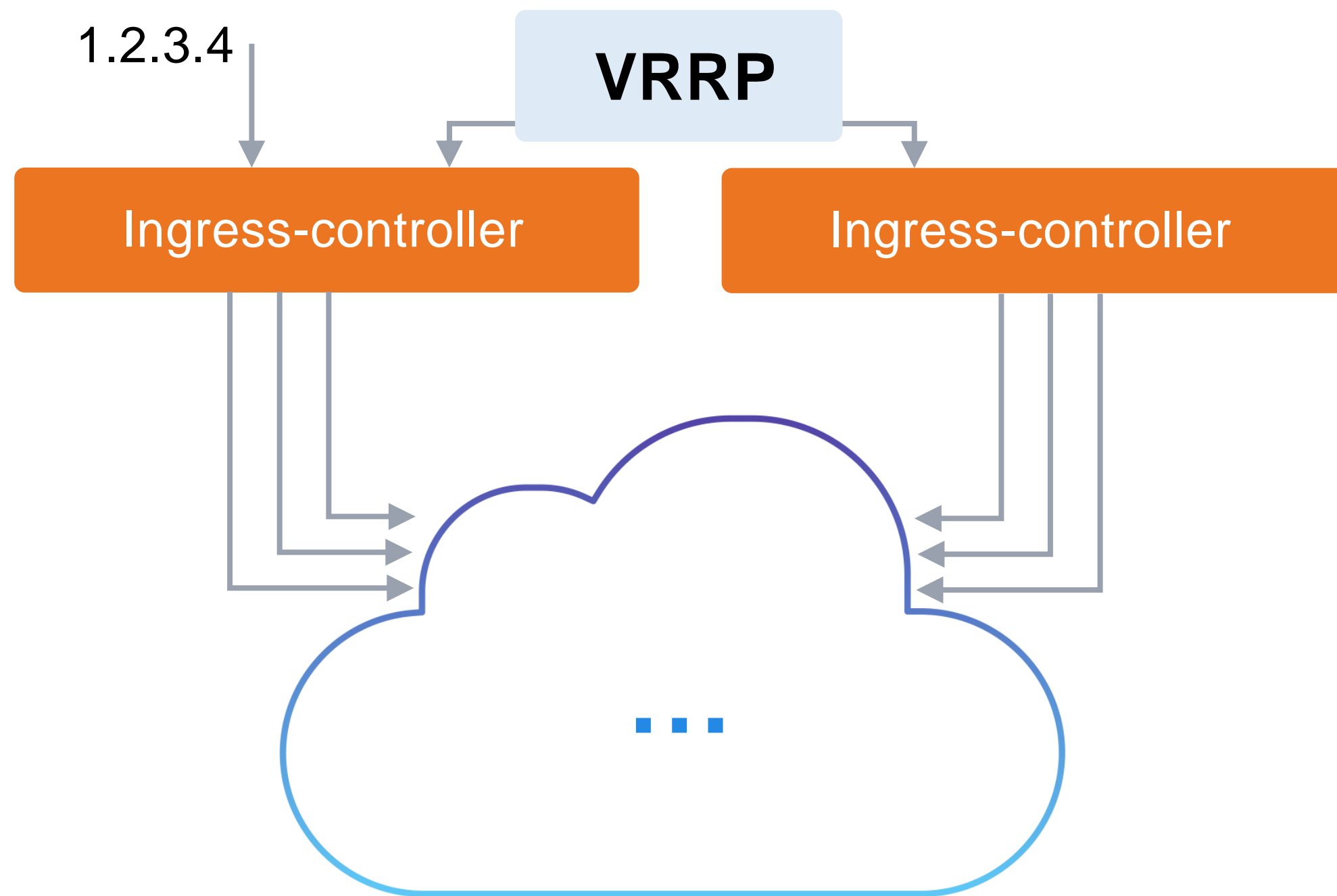
Как вам идея?



Как делаем мы:



Как делаем мы:



Сертификаты kubernetes:

- Это те что лежат в */etc/kuberenetes/pki*
- В старых версиях kubernetes сертификаты протухали
- <https://habr.com/ru/company/southbridge/blog/465733/>
- Пользуйтесь нашим форком kubespray:
<https://github.com/southbridgeio/kubespray>
- С версии 1.15 стало жить легче, а с 1.17 совсем легко

Обновление ПО на ноде кластера:

- Есть идеи как сделать?

Обновление ПО на ноде кластера:

- Убеждаемся, что у нашего приложения >1 реплики
- Эвакуируем поды с нужной ноды (kubectl drain)
- Проводим обновление ПО
- Снимаем ноду с обслуживания (kubectl uncordon)

Обновление версии кластера:

- Пробуем на dev-среде/стенде и читаем Changelog!
- Обновляем основную мастер-ноду на 1 версию
- Обновляем остальные мастер-ноды на 1 версию
- Обновляем воркер-ноды на 1 версию
- <https://kubernetes.io/docs/tasks/administer-cluster/kubeadm/kubeadm-upgrade/>
- В kubespray для этого есть отдельный сценарий

■ Бэкапы кластера

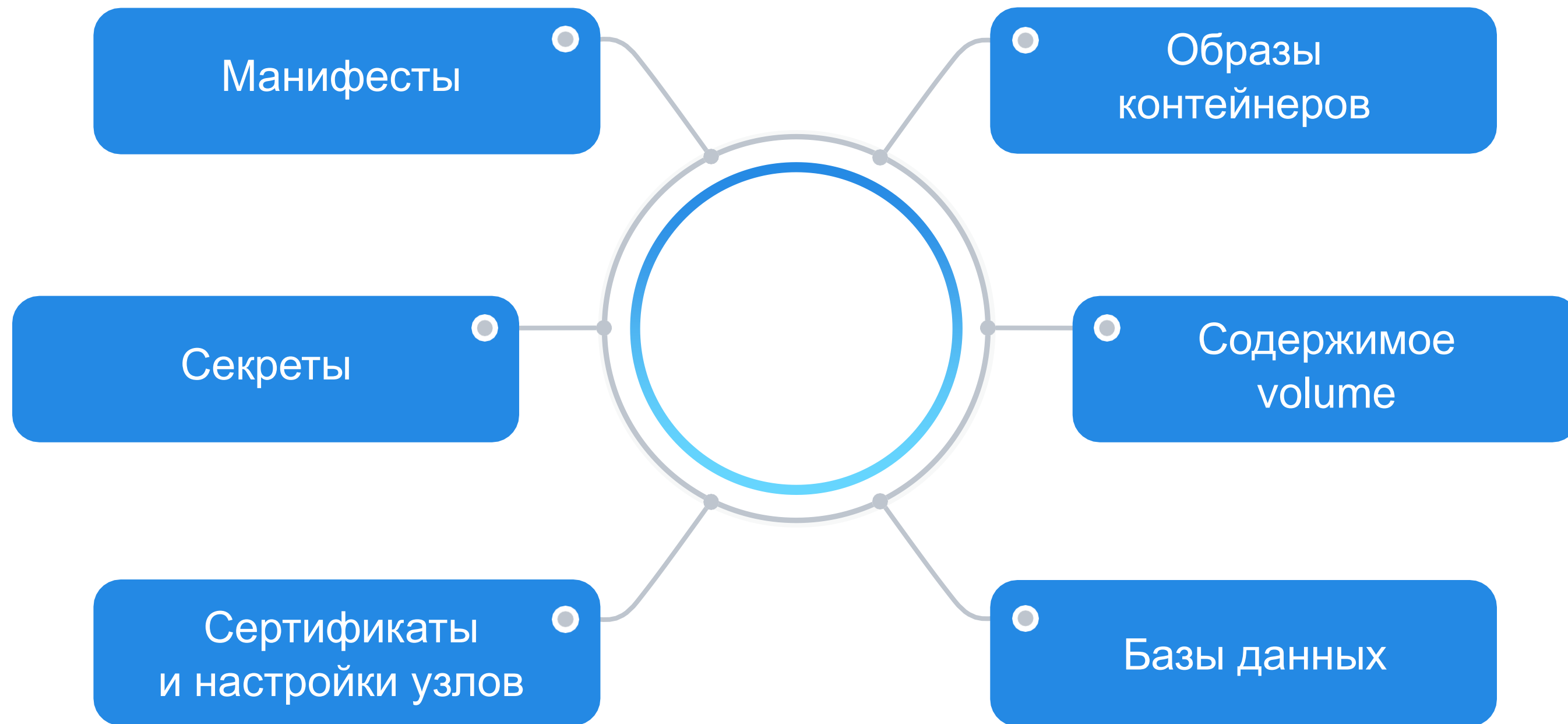
- Нужно ли бэкапить кластер Kubernetes?

■ Бэкапы кластера

- Нужно ли бэкапить кластер Kubernetes?
- Что именно следует бэкапить?



Бекапы кластера



Бекапы кластера

Infrastructure as Code (IaC)

Проще новый создать,
чем старый восстановить

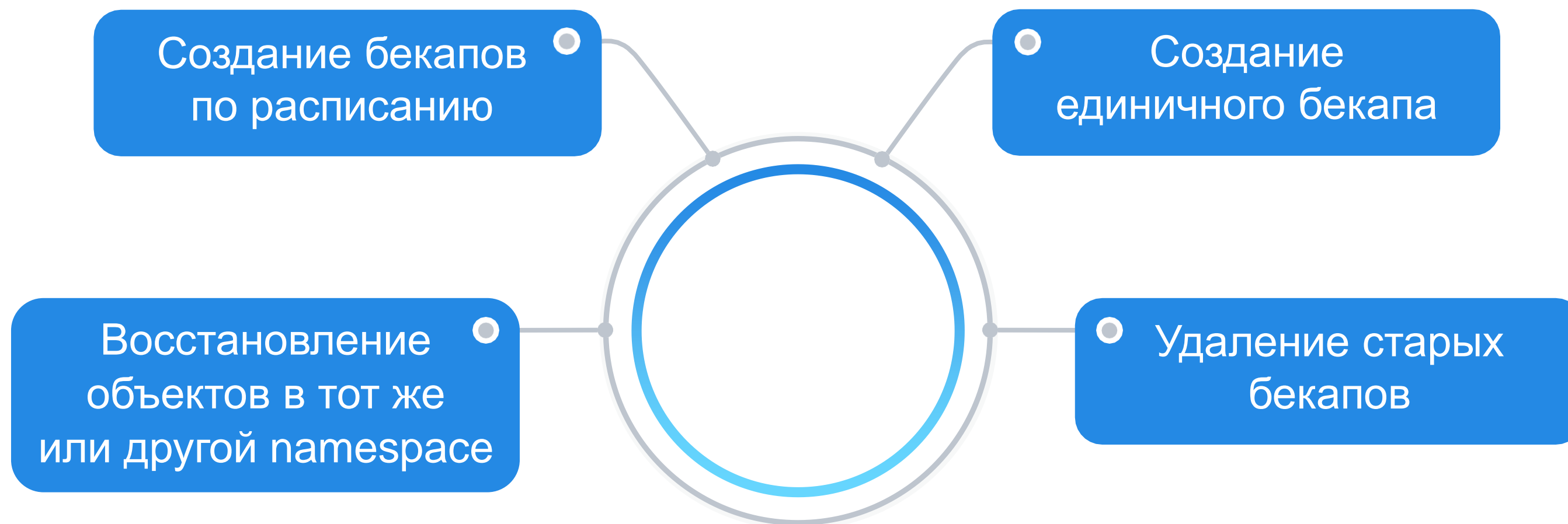
etcd snapshot

Одним махом

Heptio Velero

Придирчиво выбираем

Возможности Neptio Velero



Heptio Velero Backup Storage Provider

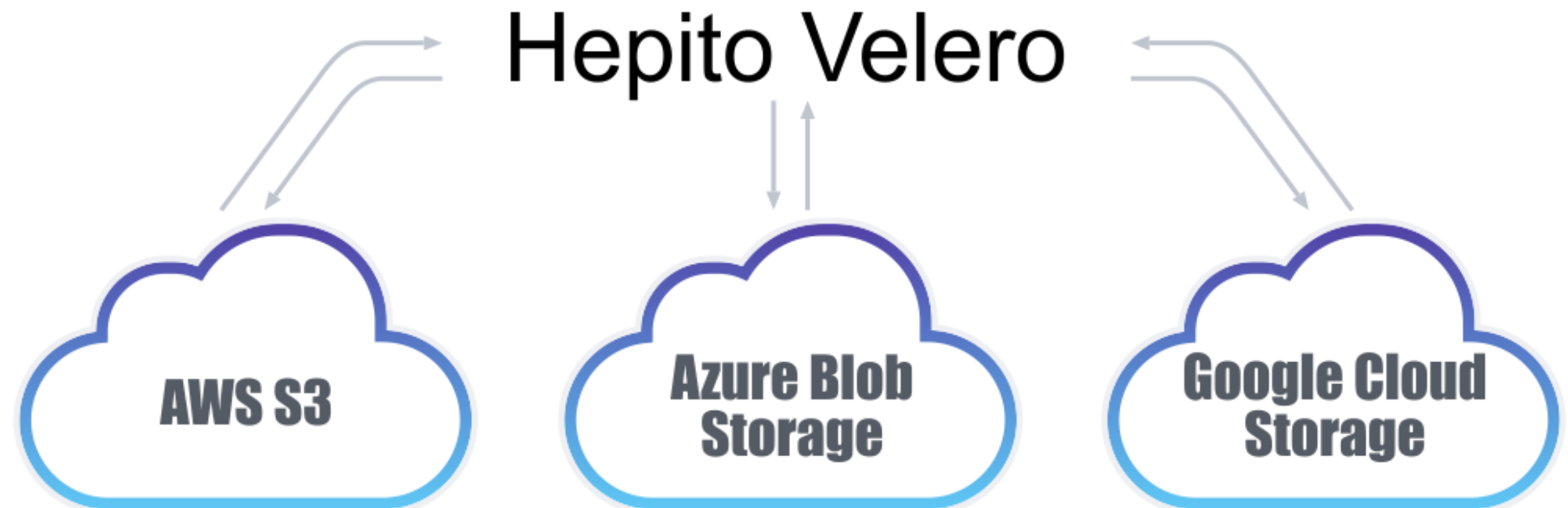
S3-Compatible:

IBM Cloud

Minio

Ceph RADOS

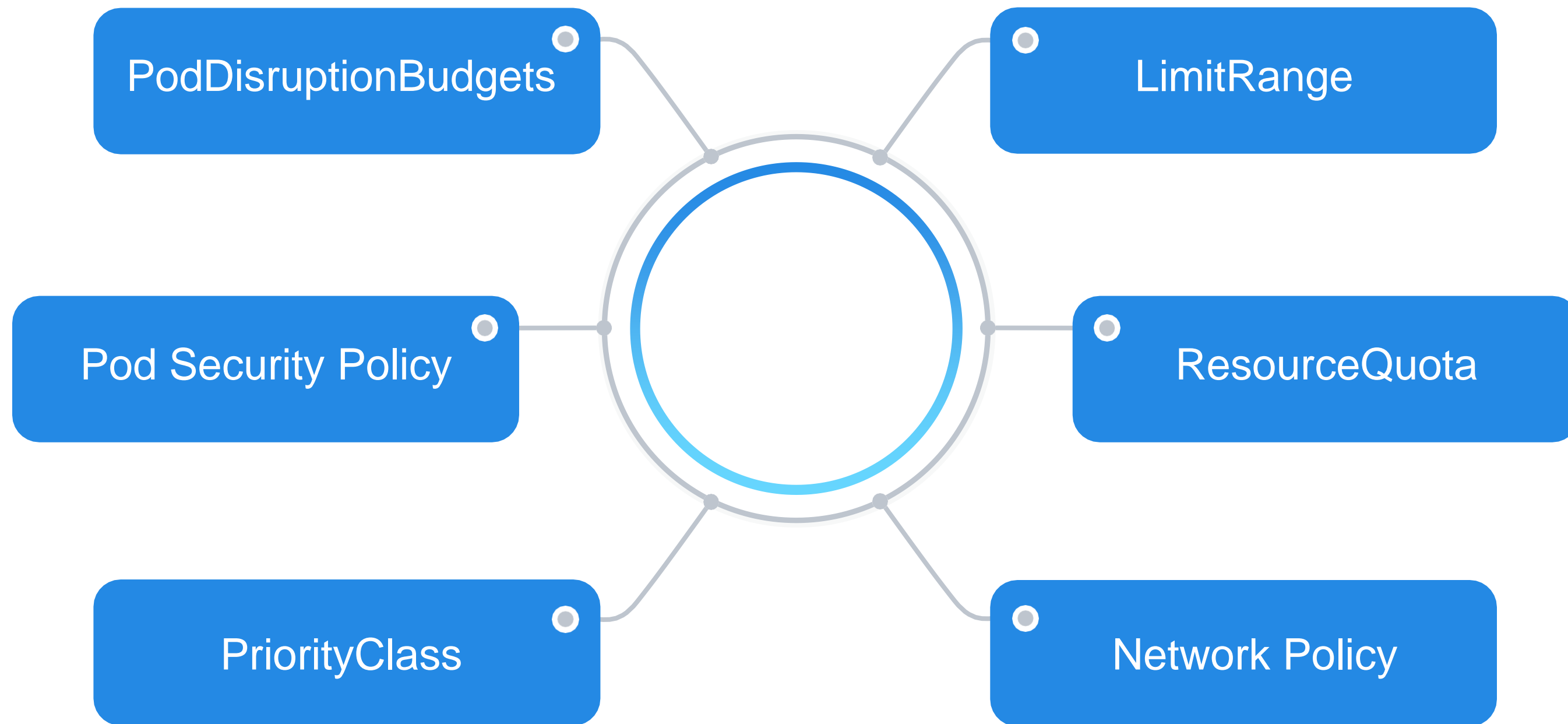
DigitalOcean



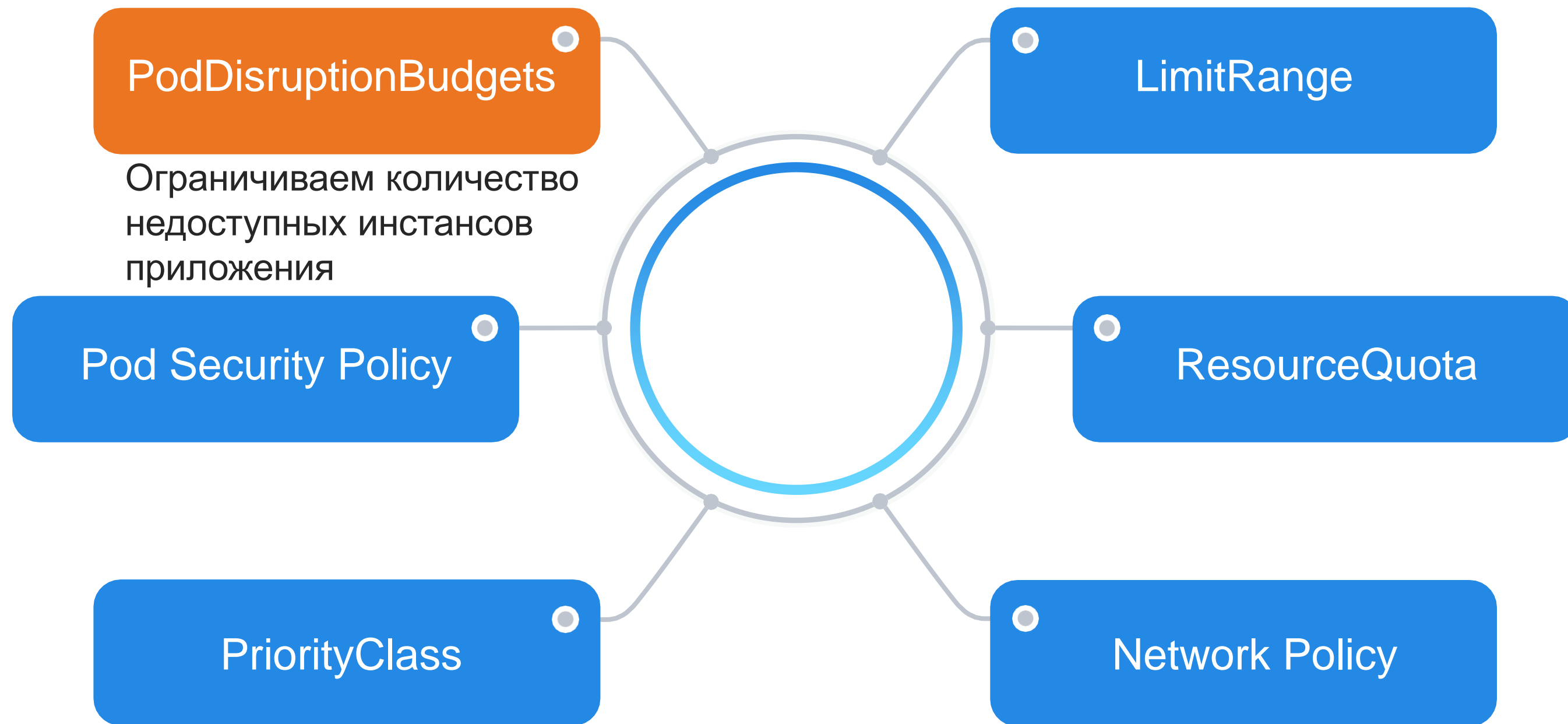
Как бороться с **человеческим фактором**?



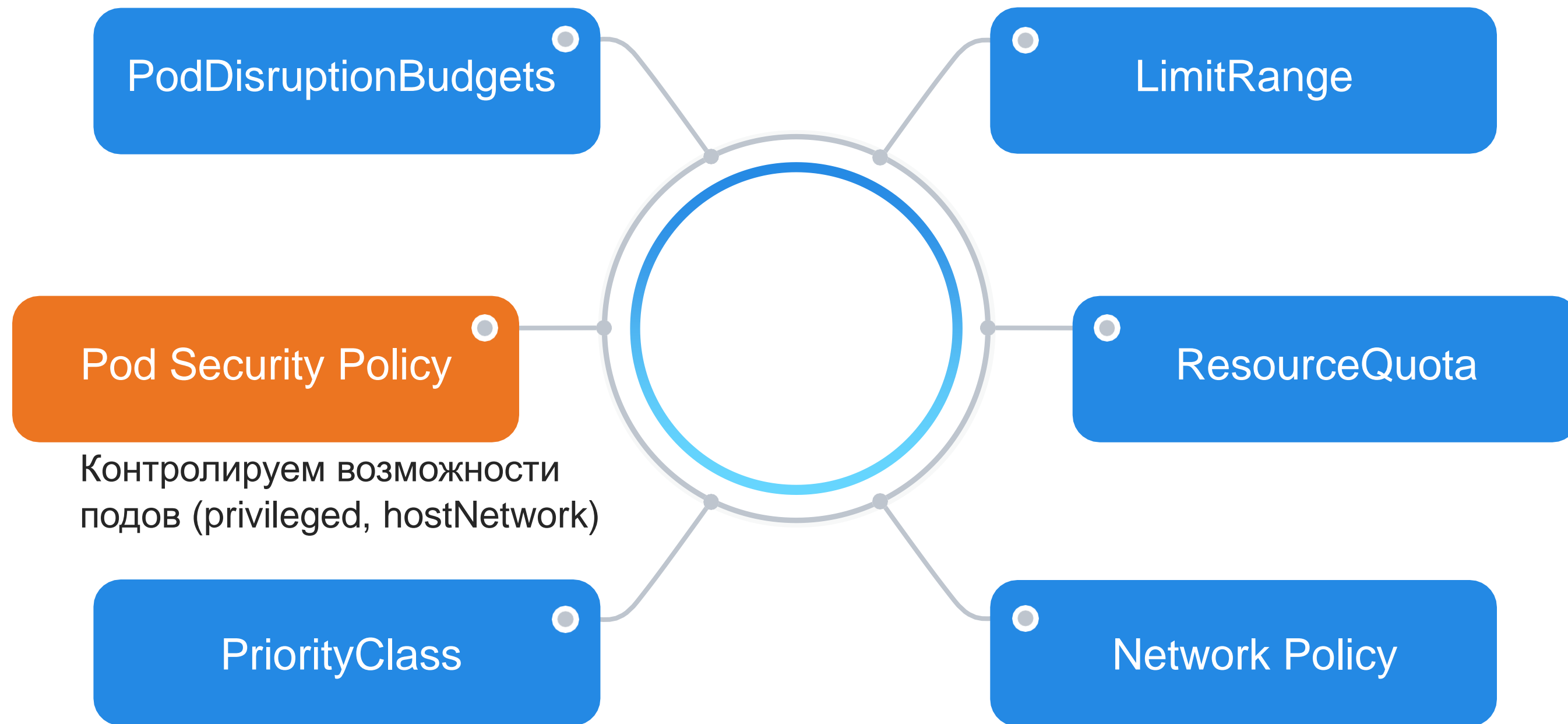
Полезные штуки



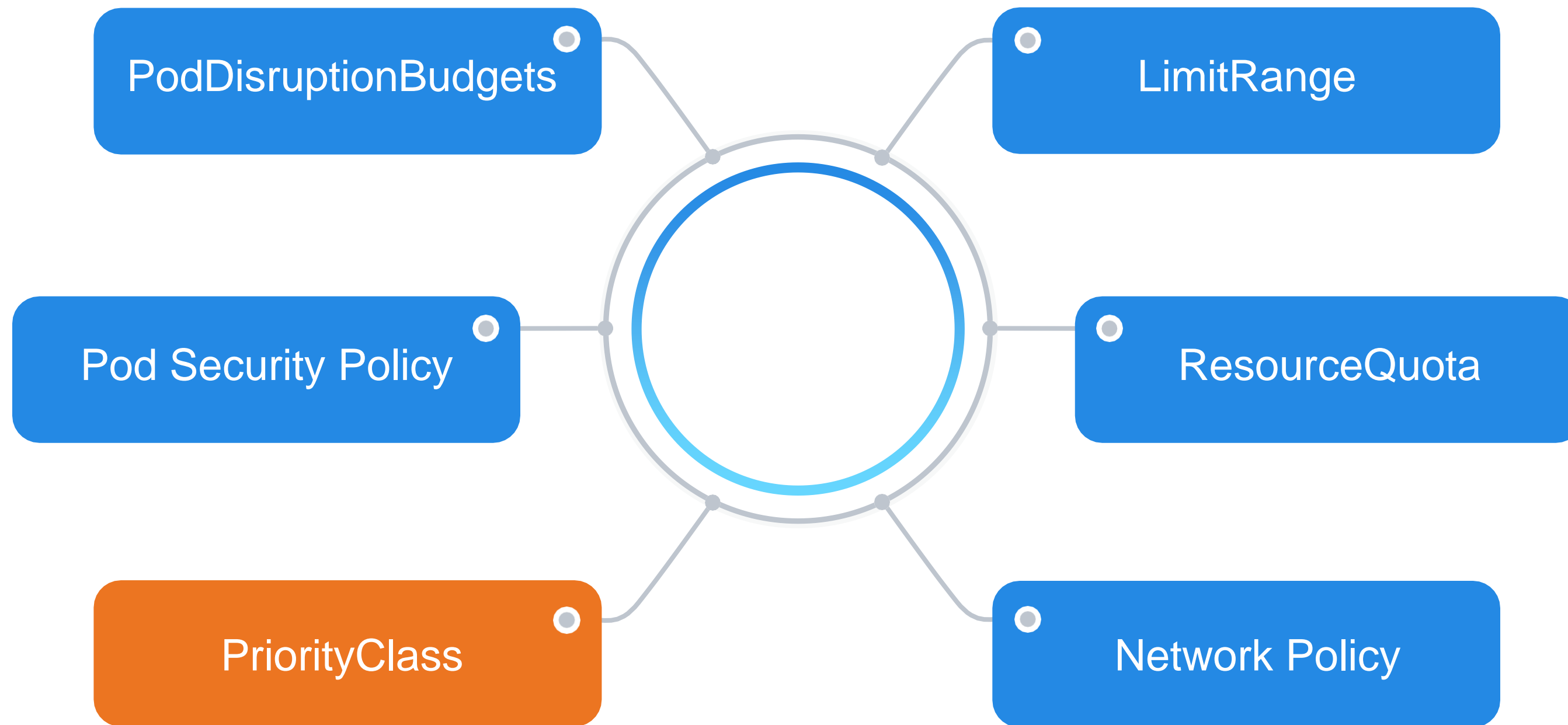
Полезные штуки



Полезные штуки

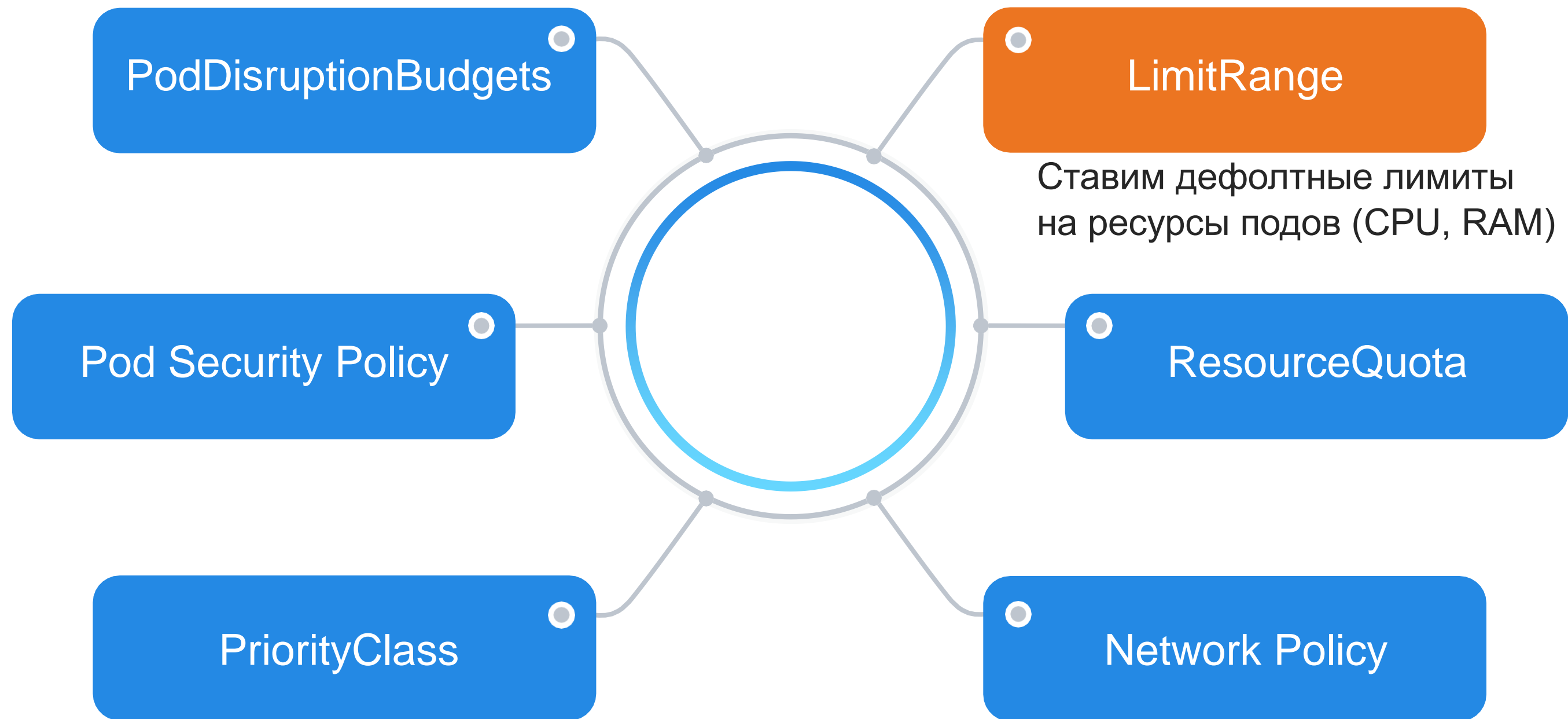


Полезные штуки

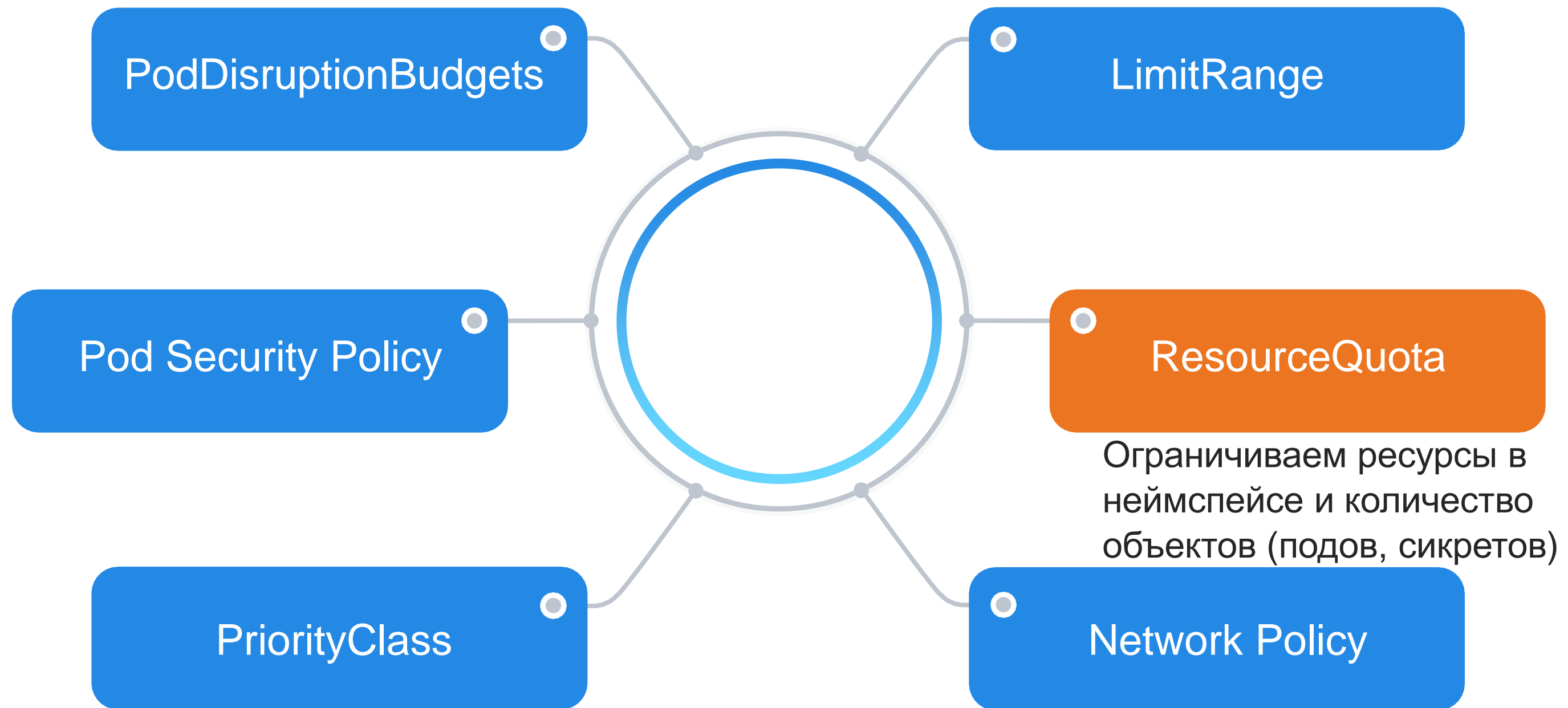


Приоритезируем более важные
поды над менее важными

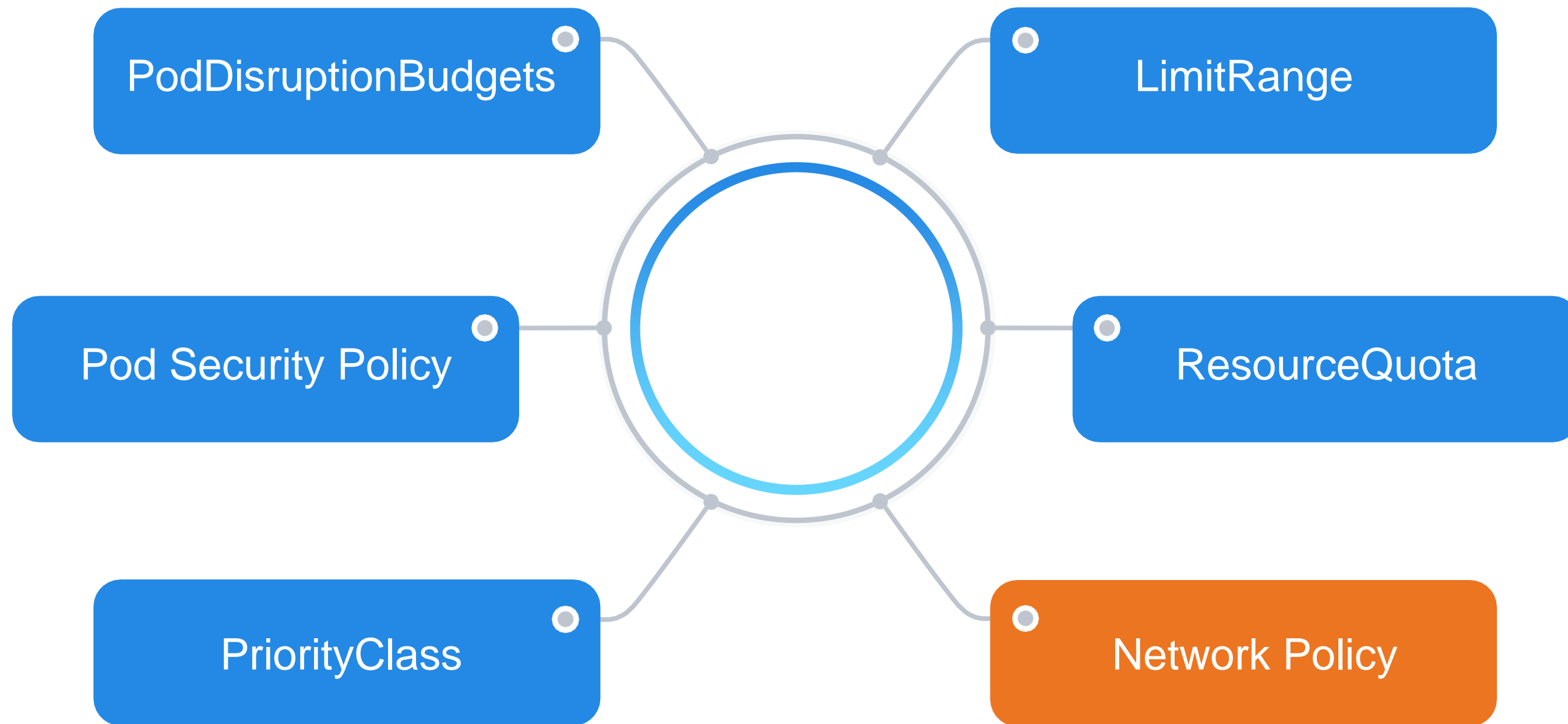
Полезные штуки



Полезные штуки



Полезные штуки



«Файервол» внутри
кубернетес

Где почитать подробнее:

- <https://kubernetes.io/docs/concepts/workloads/pods/disruptions/>
- <https://kubernetes.io/docs/concepts/policy/pod-security-policy/>
- <https://kubernetes.io/docs/concepts/configuration/pod-priority-preemption/>
- <https://kubernetes.io/docs/concepts/policy/limit-range/>
- <https://kubernetes.io/docs/concepts/policy/resource-quotas/>
- <https://kubernetes.io/docs/concepts/services-networking/network-policies/>

А если таки сломалось....:

- <https://learnk8s.io/a/f65ffe9f61de0f4a417f7a05306edd4c.png>

А если таки сломалось....:

- <https://learnk8s.io/a/f65ffe9f61de0f4a417f7a05306edd4c.png>
- kubectl logs || describe || get events || cluster-info dump || kubectl-debug

А если таки сломалось....:

- <https://learnk8s.io/a/f65ffe9f61de0f4a417f7a05306edd4c.png>
- `kubectl logs || describe || get events || cluster-info dump || kubectl-debug`
- `/var/log/containers/ || journalctl -u kubelet`

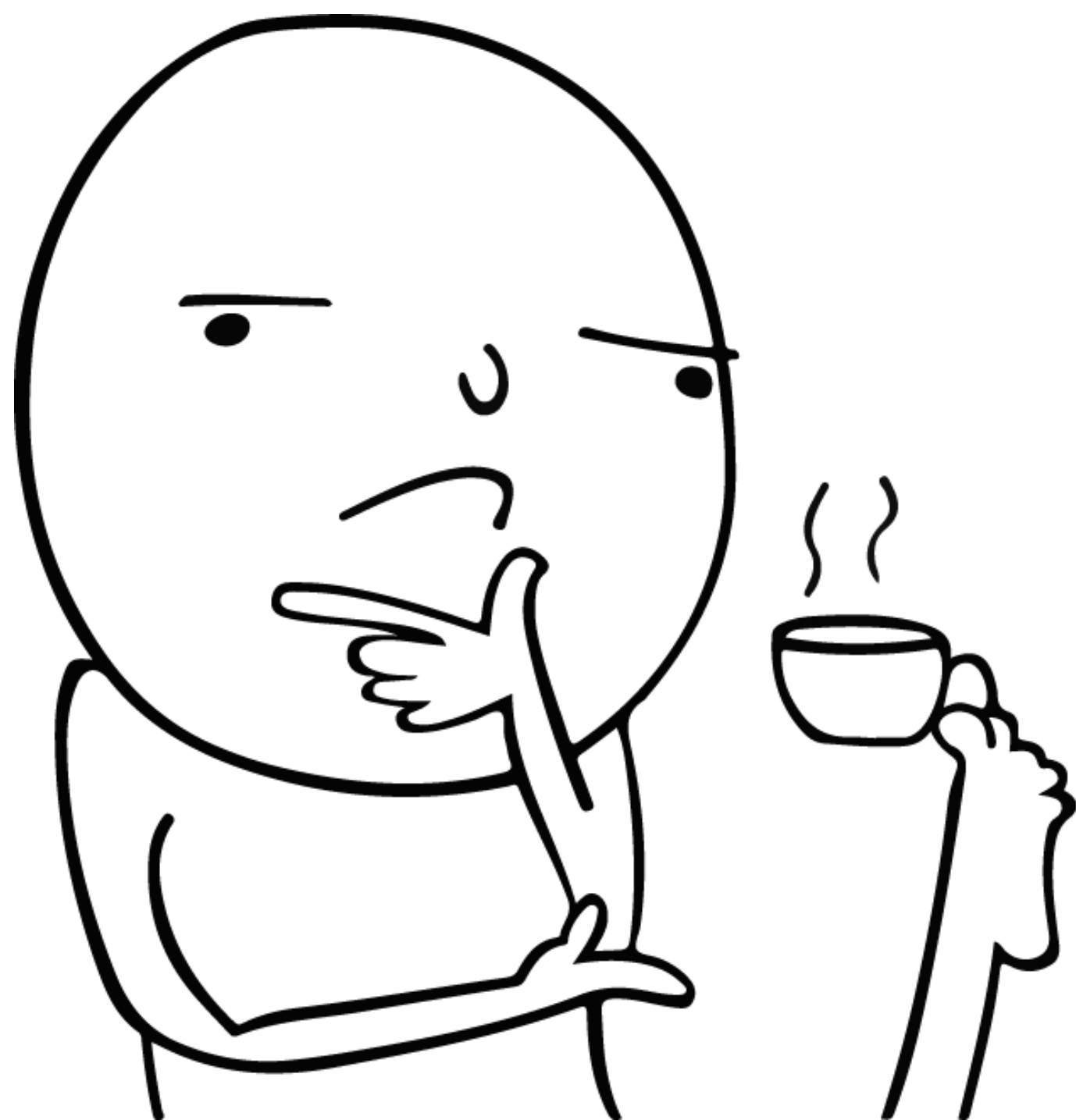
А если таки сломалось....:

- <https://learnk8s.io/a/f65ffe9f61de0f4a417f7a05306edd4c.png>
- kubectl logs || describe || get events || cluster-info dump || kubectl-debug
- /var/log/containers/ || journalctl -u kubelet
- Плясать с бубном || ~~Смириться~~

А если таки сломалось....:

- <https://learnk8s.io/a/f65ffe9f61de0f4a417f7a05306edd4c.png>
- kubectl logs || describe || get events || cluster-info dump || kubectl-debug
- /var/log/containers/ || journalctl -u kubelet
- Плясать с бубном || ~~Смириться~~
- Звонить в Southbridge!





Так и какой вывод?



Как сделать k8s бронейным:

Иметь описанный регламент по типовым работам

Размещаться в хорошем ДЦ

Настроить отказоустойчивый сетеп

Иметь dev-стенд, где экспериментировать

Покрываться мониторингами, делать бекапы

Держать руку на пульсе, иметь актуальные знания

Пользоваться всеми фишками кубернетес

Спасибо!

<https://southbridge.io/>

<https://slurm.io/>

<https://t.me/addstickers/SlurmOfficial>

6-8 июля Слёрм Базовый

13-15 июля Слёрм Мега



Southbridge

СЛЁРМ





ВСЁ!