CLOUD NATIVE COMPUTING FOUNDATION

KubeEdge

THE LINUX FOUNDATION 开源软件学园

云原生边缘计算公开课

Cloud Native Edge Computing

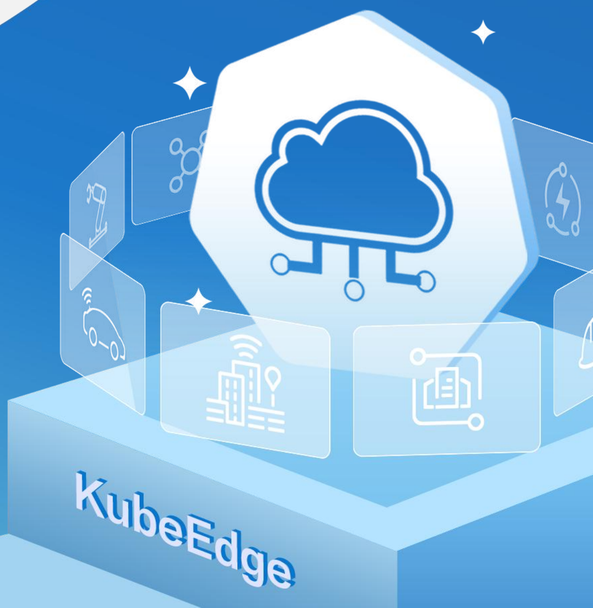
许世威

高级工程师 华为云计算公司

如何安装部署一套 KubeEdge（下）

Cloud Native Edge Computing

- 云原生边缘计算公开课 -



CLOUD NATIVE COMPUTING FOUNDATION

KubeEdge

THE LINUX FOUNDATION 开源软件学园

讲师介绍

许世威

华为云计算公司 高级工程师

硕士毕业于浙江大学，2017年加入华为云云原生团队，参与华为内部云原生技术平台建设与开源社区贡献。主要负责云原生智能边缘平台的设计与开发，KubeEdge社区研发工作，在云原生和边缘容器等领域拥有丰富的开源社区与商业落地实践经验。

目 录

Contents

01

KubeEdge集群运维

基础运维指令

KubeEdge集群常用排错方法：

使用kubectl命令，排查集群资源的当前信息，如kubectl logs、exec、describe等

- 查看KubeEdge集群对象的当前运行时信息，特别是与对象关联的Event事件。这些事件记录了相关主题、发生时间、最近发生时间、发生次数及事件原因等，对排查故障非常有价值。

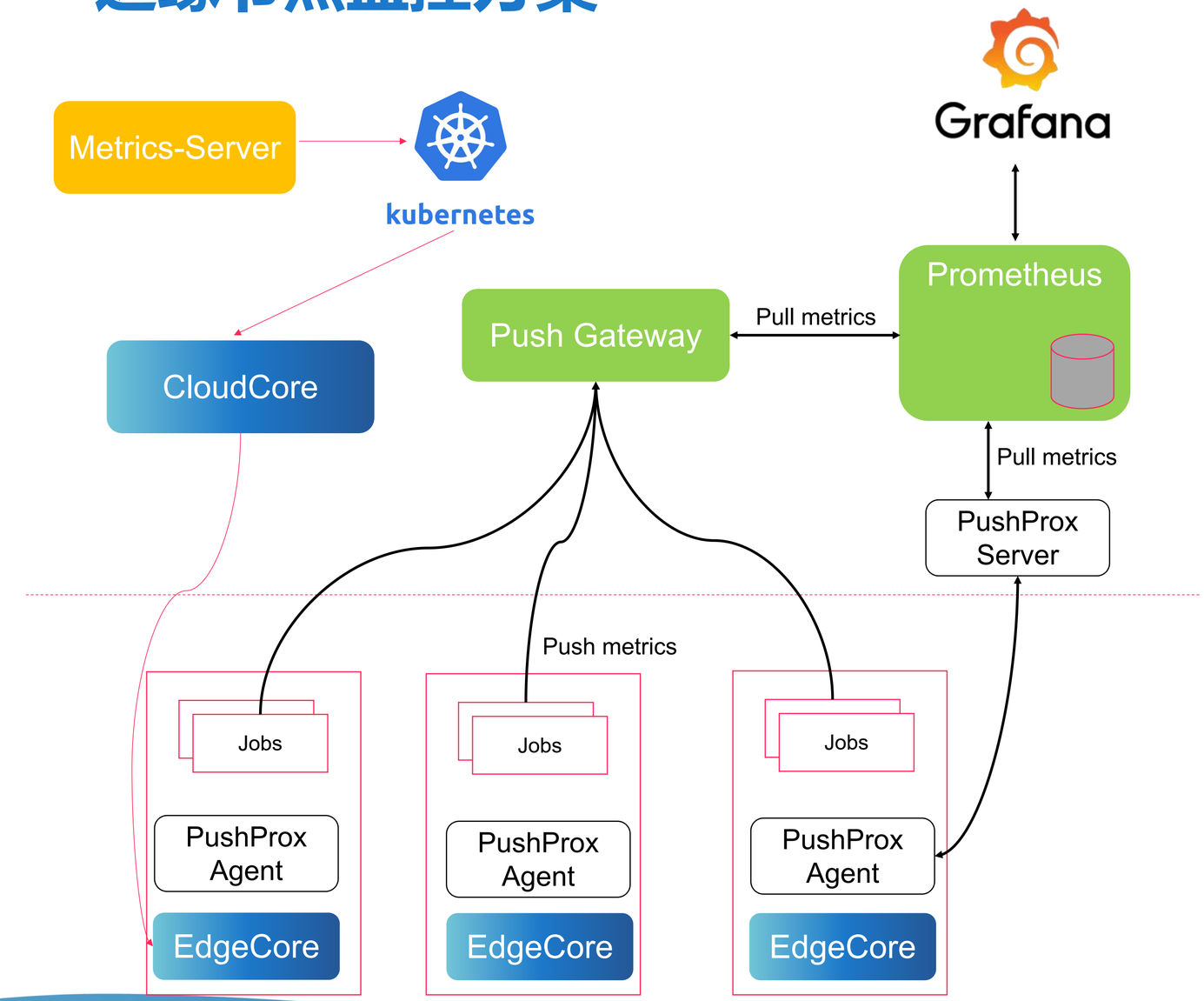
- 对于服务、容器方面的问题，可能需要深入容器内部进行故障诊断，此时可以通过查看容器的运行日志来定位具体问题。

- 对于某些复杂问题，例如Pod调度下发边缘节点这种全局性的问题，可能需要结合集群中管理组件日志来排查。比如搜集云上cloudcore日志，以及各个边缘节点上的edgecore服务日志，通过综合判断各种信息来定位问题。

常见问题举例：

1. 边缘侧无法下载镜像
2. 业务POD持续重启
3. Pod pending无法下发到边缘节点
4. 通过Service无法访问服务

边缘节点监控方案



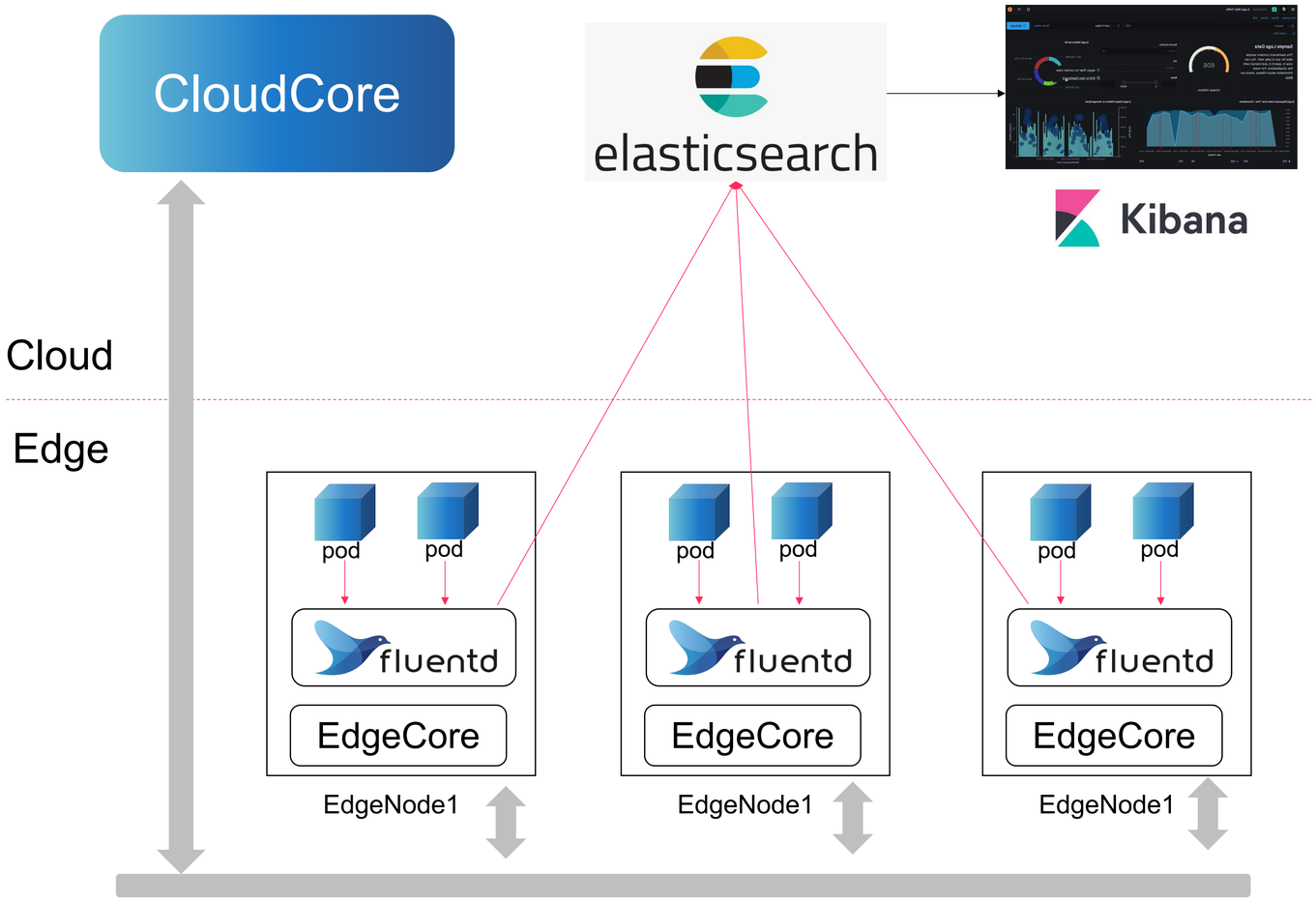
对接Metric-Server，需要开启CloudCore CloudStream和EdgeCore edgeStream 模块

使用Prometheus和Grafana统一监控云端管理组件和边缘节点

针对使用场景，可以选择push或者pull模式

Pull模式基于prometheus-community/PushProx组件，边缘节点允许agent，云端允许proxy server，将Prometheus server访问代理到边缘节点

边缘日志收集分析系统



系统介绍：

fluentd：开源数据收集器，采集容器日志文件、过滤和转换日志数据

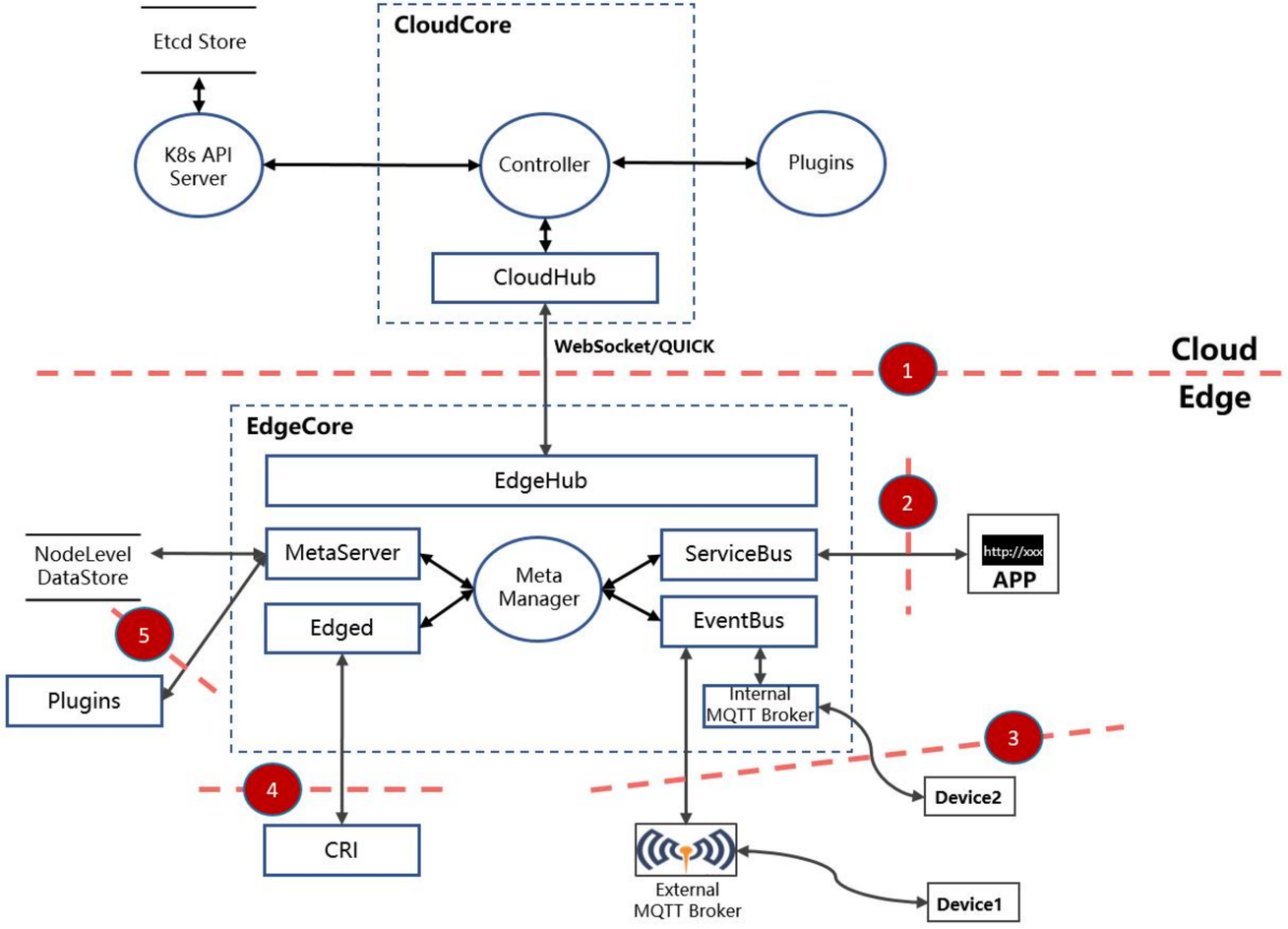
elasticsearch：实时的、分布式的可扩展的搜索引擎，用来存储日志数据

kibana：功能强大的数据可视化 Dashboard

实施要点：

- Elasticsearch和Kibana部署在云端节点，统一运维
- fluentd通过daemonset部署到需要采集日志的边缘节点
- Elasticsearch需要暴漏给边缘节点可访问，默认服务端口9200
- 边缘节点edgecore需要开启list-watch功能，并配置fluentd master地址为边缘server地址

KubeEdge安全生产实践



Threat ID 1：CloudCore与EdgeCore连接

安全加固：

- 证书加密实现双向认证，保障通信安全
- 边缘证书证书定期刷新，防止过期证书被伪造和利用，并建立边缘节点证书轮转机制，保障业务连续性

安全建议：

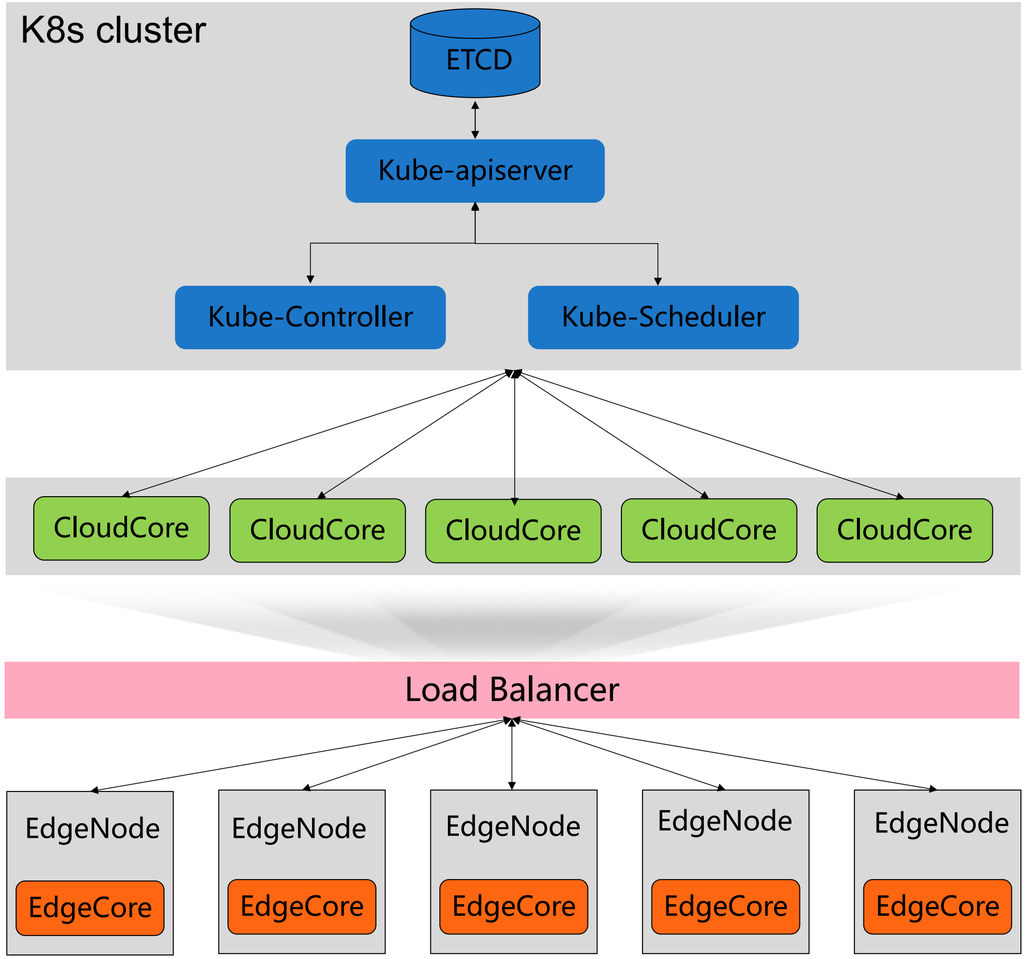
- 部署KubeEdge系统时，使用至少2048位秘钥生成相关证书
- 使用可信安全的CA根证书

详见KubeEdge威胁模型及安全防护分析
https://github.com/kubeedge/community/blob/master/sig-security/sig-security-audit/KubeEdge-threat-model-and-security-protection-analysis.md

02

KubeEdge组件升级

云端CloudCore升级



KubeEdge部署视图

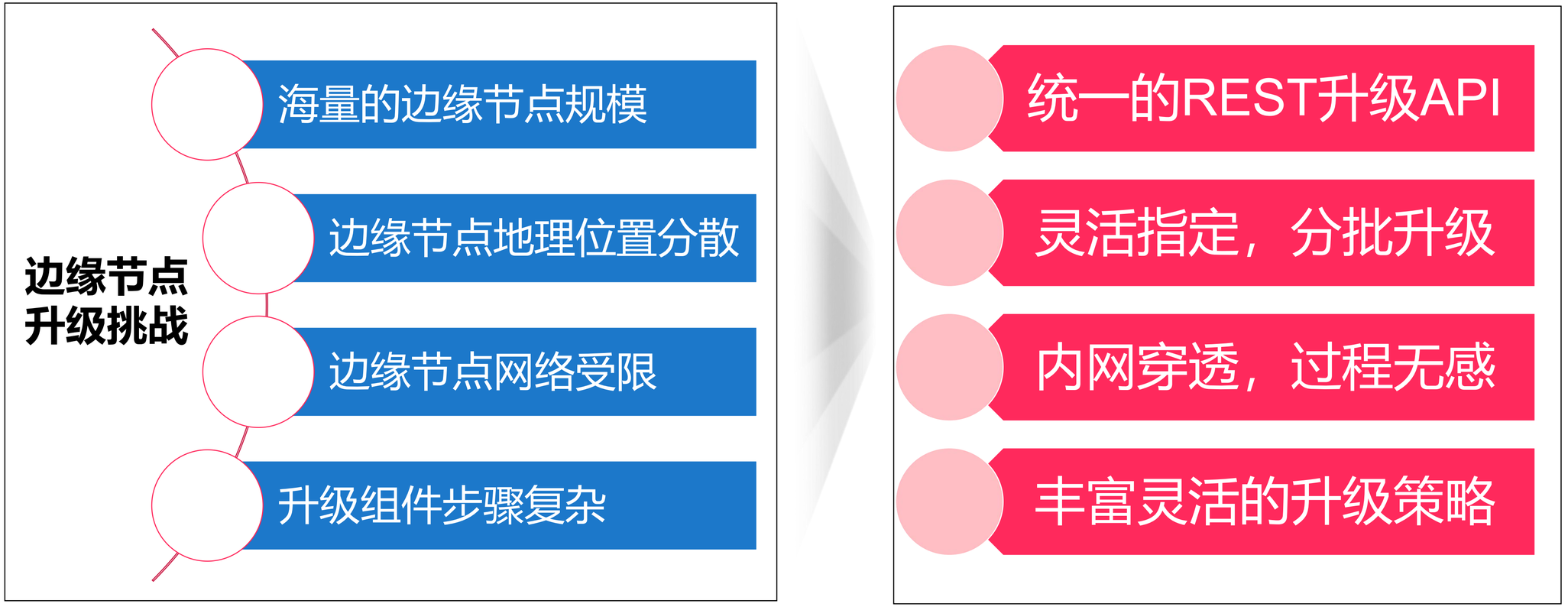
生产部署建议：

- CloudCore采用容器化部署
- CloudCore实例数>=3
- CloudCore通过负载均衡发布服务

生产升级建议：

- 升级过程中，连接到此cloudcore实例的节点会断连，如果使用了router消息路由，会有短暂的中断
- CloudCore更新版本不要跨超过3个版本
- CloudCore滚动更新，每次更新1个实例

边缘EdgeCore升级



边缘EdgeCore升级

```yaml
apiVersion: operations.kubeedge.io/v1alpha1
kind: NodeUpgradeJob
metadata:
  name: upgrade-example
  labels:
    description: upgrade-label
spec:
  version: "v1.12.1"
  timeoutSeconds: 300
  image: kubeedge/installation-package
  labelSelector:
    matchLabels:
      "node-role.kubernetes.io/edge": ""
      node-role.kubernetes.io/agent: ""
  nodeNames:
    - edge-node1
    - edge-node2
```

- **Version**：指定升级的版本号
- **upgradeTool**：支持自定义升级工具，需要在边缘侧注册
- **timeoutSeconds**：升级任务的超时时间，默认是300s，如果超过超时时间，cloudcore没有收到边缘节点的升级结果上报，则认为此次升级超时失败
- **nodeNames和labelSelector**：指定升级节点的范围，可以通过nodeName指定升级节点，或者使用labelSelector来批量选择升级的节点
- **Image**：镜像仓库的名称，用户通过指定该字段以从私有仓库下载kubeedge/installation-package安装包镜像

边缘EdgeCore升级

升级主要核心流程：

1. 用户通过kubectl或者API调用创建NodeUpgradeJob

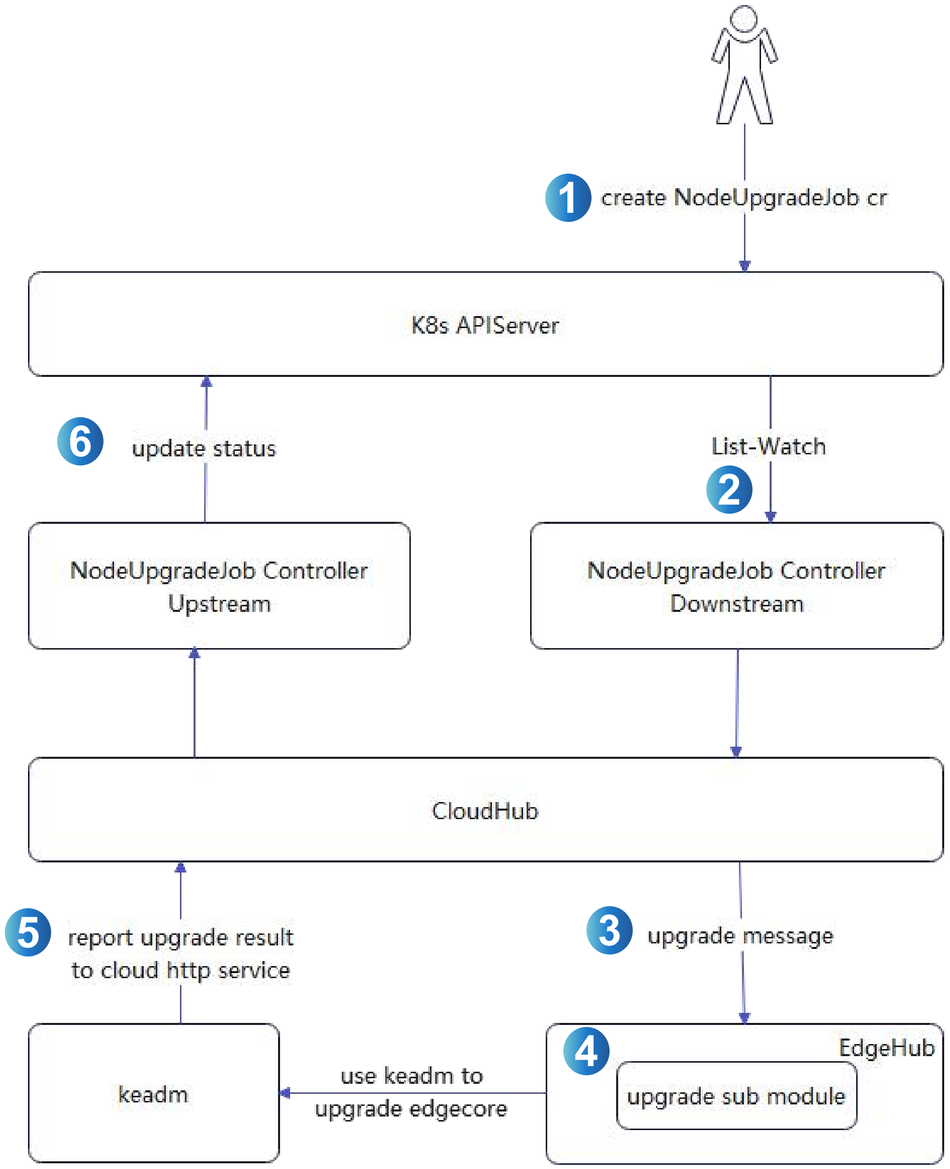
2. CloudCore watch到NodeUpgradeJob事件，并根据其配置，将对应的升级消息下发到边缘节点

3. EdgeCore收到NodeUpgradeJob升级信息，将消息转发给upgrade子模块

4. upgrade子模块子模块调用keadm触发升级动作，包括升级检查、旧版本及数据备份，下载安装镜像，以及执行升级动作

5. Keadm升级结束，将升级结果上报到CloudCore，并更新NodeUpgradeJob状态

6. 用户通过kubectl或者API调用查看NodeUpgradeJob升级结果



03

KubeEdge安装常见问题

安装常见问题-1 CloudCore安装pre-flight check失败

典型问题：

```
error execution phase preflight: [preflight] Some fatal erros occurred
```

排查思路：

1. 之前执行失败，有文件残留执行keadm reset
2. 其余的错误，请根据错误提示进行修复

安装常见问题-2 CloudCore安装超时

典型问题：

```
keadm command failed:  timed out waiting for the condition
```

排查思路：

执行以下命令查看CloudCore相关资源是否创建

kubectl get all –nkubeedge

如已经创建，查看cloudcore pod组件是否正常。

- **CloudCore pod pending状态**

  Pending状态一般是pod调度失败，可以describe pod查看具体的调度失败原因并解决

- **CloudCore pod ImagePullBackOff状态**

  CloudCore安装过程中，会下载安装所用的CloudCore镜像，默认从dockerhub官网拉取，可以提前拉取镜像并预置到节点上。

- **CloudCore pod CrashLoopBackOff状态**

  CrashLoopBackOff表明CloudCore pod运行失败，需要通过kubectl logs查看日志，根据相应的错误解决

安装常见问题-3 EdgeCore获取证书连接拒绝或超时

典型问题：

failed to get CA certificate, err: Get "https://192.168.47.128:10002/ca.crt": dial tcp 192.168.47.128:10002: connect: connection refused

failed to get CA certificate, err: Get "https://192.168.47.128:10002/ca.crt":EOF

Error: failed to get CA certificate, err: Get "https://10.19.28.176:10002/ca.crt": dial tcp 10.19.28.176:10002: connect: no route to host

failed to get CA certificate, err: Get "https://10.96.179.211:10002/ca.crt": dial tcp 10.96.179.211:10002: i/o timeout

排查思路：

1. 边缘节点join时配置的cloudcore地址是否存在于advertise-address地址列表中?
2. 边缘节点和join时配置的cloudcore地址物理网络是否联通，是否有防火墙限制?
3. 云端CloudCore组件是否正常启动？访问的端口是否正常监听?
4. 如以上都正常，使用如下命令查看CloudCore日志是否有错误产生?

   kubectl logs cloudcore-xxxx -n kubeedge

安装常见问题-4 keadm join边缘节点镜像拉取失败

典型问题：

```
edge node join failed: pull Images failed: xxx
```

排查思路：

1. 边缘节点能否访问外网，边缘节点纳管时会下载kubeedge/installation-package镜像，默认从dockerhub官网拉取，如果可以联通外网，则根据具体的报错进一步解决。
2. 如果不能访问外网，则可以通过以下两种方式规避：
   - 从其他可以访问外网的机器手动下载镜像，然后将镜像加载到边缘节点上
   - 从其他可以访问外网的机器手动下载镜像，将镜像上传至内部的镜像仓库，纳管时，通过--image-repository=xxx指定镜像仓库地址

安装常见问题-5 EdgeCore token认证失败

典型问题：

```
Error:failed to get edge certificate from the cloudcore, error: Invalid authorization token
```

```
certmanager.go:94] Error: token credentials are in the wrong format
```

排查思路：

1. Token没有copy完整，报格式错误
2. Token过期失效，重新执行keadm gettoken

安装常见问题-6 EdgeCore获取证书认证失败

典型问题：

```
Error: failed to get edge certificate from the cloudcore, error: Get
"https://10.176.122.3:10002/edge.crt": x509: certificate is valid for 10.176.122.1, not
10.176.122.3
```

```
Error: failed to get edge certificate from the cloudcore, error: Get
"https://10.1.27.1:10002/edge.crt": x509: cannot validate certificate for 10.1.27.1 because it
doesn't contain any IP SANs
```

排查思路：

CloudCore安装时配置的advertise-address，为逗号分隔的多个地址，需要提前规划好，推荐通过负载均衡或者网关地址发布对边缘侧的访问，以保障CloudCore的负载均衡和高可用，目前此地址不允许变更，边缘节点join时配置的cloudcore-ipport，**必须存在于advertise-address列表中。**

```
keadm init --advertise-address=xxx.xxx.xxx.xxx --profile version=v1.12.1 --kube-
config=/root/.kube/config
keadm join --cloudcore-ipport=xxx.xxx.xxx.xxx:10000 --edgenode-name=testing123 --kubeedge-
version=v1.12.1
```

添加社区小助手

课后答疑 | 大咖交流 | 社区礼品

Github: https://github.com/kubeedge

Website: https://kubeedge.io

Slack Channel: https://kubeedge.slack.com

KubeEdge公众号

课程小助手

Thanks


CLOUD NATIVE
COMPUTING FOUNDATION
KubeEdge
THE
LINUX
FOUNDATION
开源软件学园