

KYLIN-5346

BackGround

因为构建任务卡住导致构建不可用，且当前 ops plan 指标，只能统计任务完成后的数据，无法对正在执行的任务状态进行监控，所以期望增加对 long running 状态的任务进行监控。

Dev Design

定义指标：超时的任务数量：ke_long_running_jobs{project=xx, state="waiting|running",timeout="10m|15m|30m 0.5h|1h|1.5h|2h|3h"}，指标类型：Gauge

指标label说明：

project: 项目名称

state: 状态, waiting: 等待 running: 运行中

timeout: 超时时间, waiting状态任务默认为：5m|10m|15m|30m；running状态任务默认为：0.5h,1h,1.5h,2h,3h

具体代码层面实现逻辑如下：

在原有逻辑基础上（MetricsRegistry.registerProjectPrometheusMetrics(KylinConfig kylinConfig, String project)）增加新增指标的注册逻辑。

大体思路：NDefaultScheduler.getInstance(project).getRunningJobs() 从内存中获取当前节点的所有运行中的任务然后过滤出符合要求（long running）的任务数量并记录到ke_long_running_jobs 指标中

告警规则定义：

新增两条告警规则

1. Too Many Long Waiting Jobs

针对长时间运行等待的构建任务进行告警，默认：等待时间超过15分钟的构建任务数大于5并且持续时间=1分钟则触发告警，告警级别：critical。

2. Too Many Long Running Jobs

针对长时间运行卡住的构建任务进行告警，默认：运行时间超过2小时一直未完成的构建任务数大于3并且持续时间=1分钟则触发告警，告警级别：critical。

详情如下：

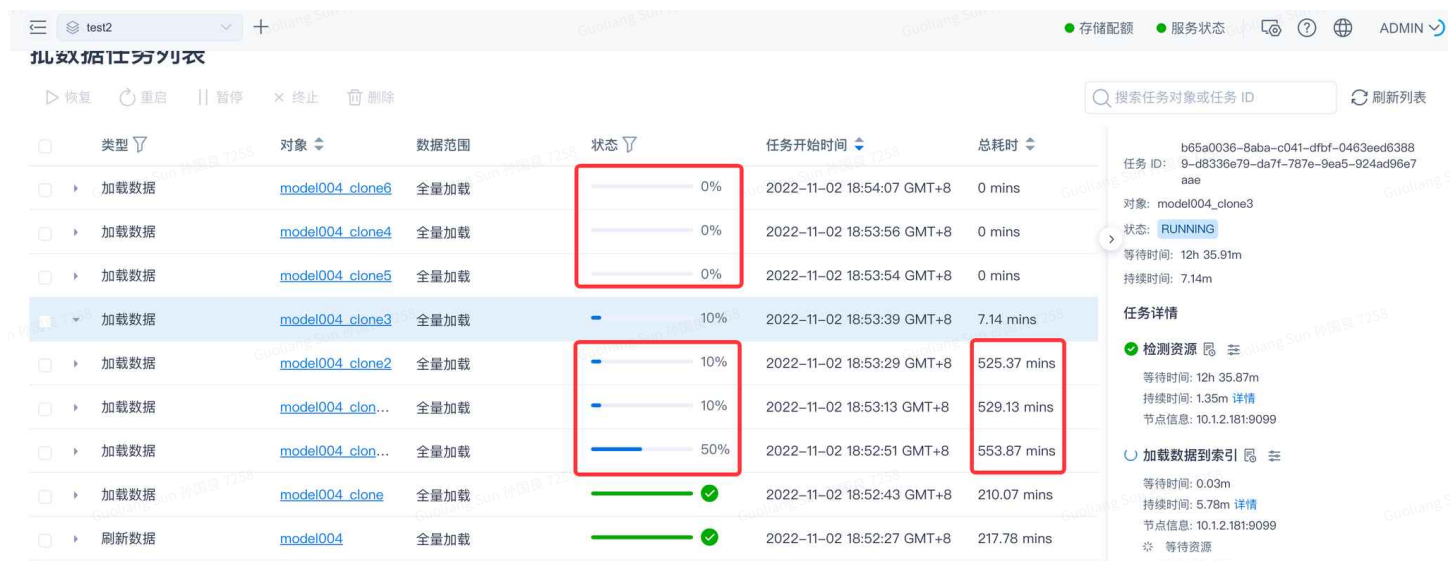
```
1 - alert: Too Many Long Waiting Jobs
2   expr: max by (job) (ke_long_running_jobs{state="waiting",timeout="15m"}) > 5
3   for: 1m
4   labels:
5     severity: critical
6   annotations:
7     summary: "Cluster {{$labels.job}} long waiting jobs is too many"
8     description: "Cluster {{$labels.job}} long waiting jobs is too many, current
9
10 - alert: Too Many Long Running Jobs
11   expr: max by (job) (ke_long_running_jobs{state="running",timeout="2h"}) > 3
12   for: 1m
13   labels:
14     severity: critical
15   annotations:
16     summary: "Cluster {{$labels.job}} long running jobs is too many"
17     description: "Cluster {{$labels.job}} long running jobs is too many, current
```

Test Advice

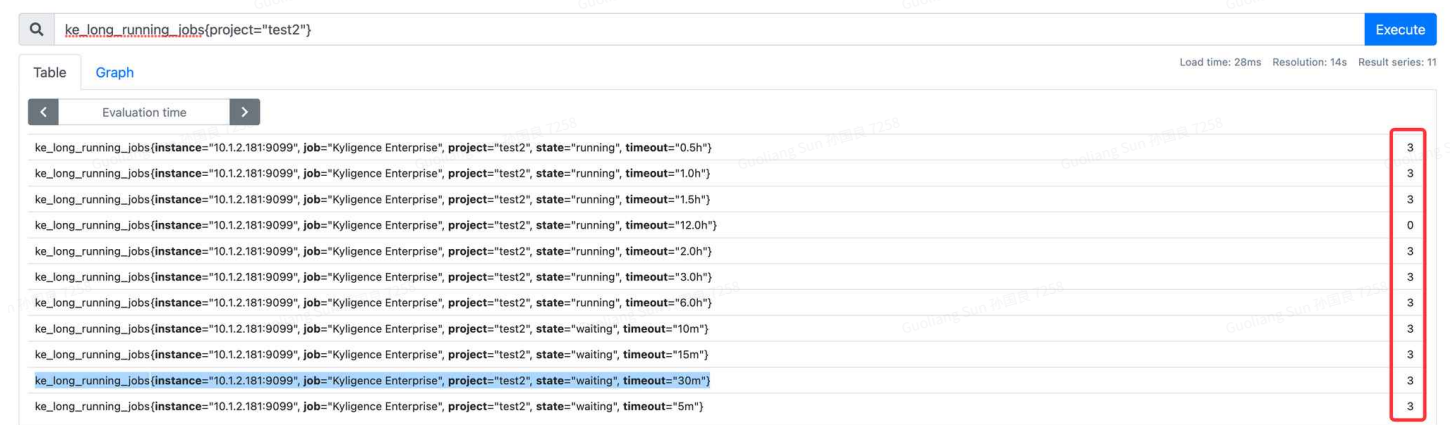
1. 验证指标正确性
2. 验证新增告警规则的正确性

Test Evidence

造3个 pending 状态的任务，3个 long running状态的任务。如下：



查看 prometheus 接口指标输出，符合预期：



验证告警规则配置正确性，可正常触发告警，符合预期，如下图：

