

## 背景

在《Prometheus监控实战》这本书中，我们知道一个好的监控系统应该满足/提供以下内容：

- 全局视角，从最高层(业务)依次展开；
- 协助故障诊断；
- 作为基础设施、应用程序开发和业务人员的信息源；
- 内置于应用程序设计、开发和部署的生命周期中；
- 尽可能的自动化，并提供自服务；

拥有了一个好的监控系统之后，这还不够。我们还需要一个告警消息，告警消息可以为我们体统一些指示，表明我们环境中的某些状态已经发生了变化，而且通常是一些更糟糕的情况，一个好的告警消息的关键应该是能够在正确的时间，以正确的理由和正确的速度发送，并且其中包含了有用的信息；

那么告警消息的发送的目的地，也成为了其中关键的一个环节，从传统的E-mail、电话、短信通知到现在层出不穷的企业办公聊天软件，如钉钉、企业微信、飞书等，选择合适的工具来接收通知，以保证处理人员能够及时的收到消息尤为重要；

以SUSE Rancher企业版为例，其中集成了除E-mail、slack等常用的通知平台外，还特别针对中国企业用户集成了钉钉、企业微信、阿里云短信等通知平台，但是即使是这样，在某些企业用户的角度仍然无法满足其需求，比如说飞书用户，就没办法直接进行对接。在本文中，使用SUSE Rancher告警通知中的Webhook方式结合另一个优秀的开源项目PrometheusAlert实现多种通知平台的告警发送；

## PrometheusAlert全家桶

\*PrometheusAlert是开源的运维告警中心消息转发系统，支持主流的监控系统Prometheus、Zabbix，日志系统Graylog2，Graylog3、数据可视化系统Grafana、SonarQube。阿里云-云监控，以及所有支持WebHook接口的系统发出的预警消息，支持将收到的这些消息发送到钉钉，微信，email，飞书，腾讯短信，腾讯电话，阿里云短信，阿里云电话，华为短信，百度云短信，容联云电话，七陌短信，七陌语音，TeleGram，百度Hi(如流)等。

以上内容引用PrometheusAlert项目简介，可以看到这个项目可以对接多种的主流监控系统，并且支持通过webhook接口发送的告警消息，推送到多种接受平台。并且项目处于活跃状态，在进行不断更新，未来会有更多的平台接入，感兴趣的同学可以查看并参与其中。

- 项目地址：<https://github.com/feiyu563/PrometheusAlert>

## 让告警飞起来

本文将使用SUSE Rancher和PrometheusAlert实现SUSE Rancher产生的告警通知发送到飞书中进行演示，更多配置可以查看PrometheusAlert项目文档；

### 创建飞书机器人

- 创建一个飞书群组，并添加一个自定义机器人；



- 记录生成的webhook地址；这一步可以根据需求定义安全相关配置；



- 更多关于飞书机器人请查看链接：

<https://open.feishu.cn/document/ukTMukTMukTM/ucTM5YjL3ETO24yNxkjN>

本文将PrometheusAlert部署在SUSE Rancher创建的自定kubernetes集群中，使用yaml文件进行部署；

## 创建Configmap

这一步骤中创建PrometheusAlert的Configmap配置，PrometheusAlert也提供了通过环境变量的方式注入配置，但是为了方便管理，这里使用Configmap的方式；

```
apiVersion: v1
data:
  app.conf: |-
    #-----↓全局配置-----
    appname = PrometheusAlert
    #登录用户名
    login_user=prometheusalert
    #登录密码
    login_password=prometheusalert
    #监听地址
    httpaddr = "0.0.0.0"
    #监听端口
    httpport = 8080
    runmode = dev
    #设置代理 proxy = http://123.123.123.123:8080
    proxy =
    #开启JSON请求
    copyrequestbody = true
    #告警消息标题
    title=PrometheusAlert
    #链接到告警平台地址
    GraylogAlerturl=http://graylog.org
    #钉钉告警 告警logo图标地址

logourl=https://raw.githubusercontent.com/feiyu563/PrometheusAlert/master/
doc/alert-center.png
    #钉钉告警 恢复logo图标地址

rlogourl=https://raw.githubusercontent.com/feiyu563/PrometheusAlert/master
/doc/alert-center.png
    #短信告警级别(等于3就进行短信告警) 告警级别定义 0 信息,1 警告,2 一般严重,3 严重,4
灾难
    messagelevel=3
    #电话告警级别(等于4就进行语音告警) 告警级别定义 0 信息,1 警告,2 一般严重,3 严重,4
灾难
    phonecalllevel=4
    #默认拨打号码(页面测试短信和电话功能需要配置此项)
    defaultphone=xxxxxxx
    #故障恢复是否启用电话通知0为关闭,1为开启
    phonecallresolved=0
    #是否前台输出file or console
    logtype=file
    #日志文件路径
    logpath=logs/prometheusalertcenter.log
```

```
#转换Prometheus,graylog告警消息的时区为CST时区(如默认已经是CST时区,请勿开启)
prometheus_cst_time=0
#数据库驱动,支持sqlite3,mysql,postgres如使用mysql或postgres,请开启
db_host,db_port,db_user,db_password,db_name的注释
db_driver=sqlite3
#db_host=127.0.0.1
#db_port=3306
#db_user=root
#db_password=root
#db_name=prometheusalert
#是否开启告警记录 0为关闭,1为开启
AlertRecord=0
#是否开启告警记录定时删除 0为关闭,1为开启
RecordLive=0
#告警记录定时删除周期,单位天
RecordLiveDay=7
# 是否将告警记录写入es7, 0为关闭, 1为开启
alert_to_es=0
# es地址,是[]string
# beego.Appconfig.Strings读取配置为[]string,使用";"而不是","
to_es_url=http://localhost:9200
# to_es_url=http://es1:9200;http://es2:9200;http://es3:9200
# es用户和密码
# to_es_user=username
# to_es_pwd=password

#-----↓webhook-----
#是否开启钉钉告警通道,可同时开始多个通道0为关闭,1为开启
open-dingding=1
#默认钉钉机器人地址
ddurl=https://oapi.dingtalk.com/robot/send?access_token=xxxxxx
#是否开启 @所有人(0为关闭,1为开启)
dd_isatall=1

#是否开启微信告警通道,可同时开始多个通道0为关闭,1为开启
open-weixin=1
#默认企业微信机器人地址
wxurl=https://qyapi.weixin.qq.com/cgi-bin/webhook/send?key=xxxxxx

#是否开启飞书告警通道,可同时开始多个通道0为关闭,1为开启
open-feishu=0
#默认飞书机器人地址
fsurl=https://open.feishu.cn/open-apis/bot/hook/xxxxxxxxxx

#-----↓腾讯云接口-----
#是否开启腾讯云短信告警通道,可同时开始多个通道0为关闭,1为开启
open-tdx=0
#腾讯云短信接口key
TXY_DX_appkey=xxxxxx
#腾讯云短信模版ID 腾讯云短信模版配置可参考 prometheus告警:{1}
TXY_DX_tpl_id=xxxxxx
#腾讯云短信sdk app id
TXY_DX_sdkappid=xxxxxx
#腾讯云短信签名 根据自己审核通过的签名来填写
```

TXY\_DX\_sign=腾讯云

#是否开启腾讯云电话告警通道,可同时开始多个通道0为关闭,1为开启

open-txdh=0

#腾讯云电话接口key

TXY\_DH\_phonecallappkey=xxxxxx

#腾讯云电话模版ID

TXY\_DH\_phonecalltpl\_id=xxxxxx

#腾讯云电话sdk app id

TXY\_DH\_phonecallsdkappid=xxxxxx

#-----↓华为云接口-----

#是否开启华为云短信告警通道,可同时开始多个通道0为关闭,1为开启

open-hwdx=0

#华为云短信接口key

HWY\_DX\_APP\_Key=xxxxxxxxxxxxxxxxxxxxxxxxxx

#华为云短信接口Secret

HWY\_DX\_APP\_Secret=xxxxxxxxxxxxxxxxxxxxxxxxxx

#华为云APP接入地址(端口接口地址)

HWY\_DX\_APP\_Url=https://rtcsms.cn-north-1.myhuaweicloud.com:10743

#华为云短信模板ID

HWY\_DX\_Templateid=xxxxxxxxxxxxxxxxxxxxxxxxxx

#华为云签名名称,必须是已审核通过的,与模板类型一致的签名名称,按照自己的实际签名填写

HWY\_DX\_Signature=华为云

#华为云签名通道号

HWY\_DX\_Sender=xxxxxxxxxx

#-----↓阿里云接口-----

#是否开启阿里云短信告警通道,可同时开始多个通道0为关闭,1为开启

open-alydx=0

#阿里云短信主账号AccessKey的ID

ALY\_DX\_AccessKeyId=xxxxxxxxxxxxxxxxxxxxxxxxxx

#阿里云短信接口密钥

ALY\_DX\_AccessSecret=xxxxxxxxxxxxxxxxxxxxxxxxxx

#阿里云短信签名名称

ALY\_DX\_SignName=阿里云

#阿里云短信模板ID

ALY\_DX\_Template=xxxxxxxxxxxxxxxxxxxxxxxxxx

#是否开启阿里云电话告警通道,可同时开始多个通道0为关闭,1为开启

open-alydh=0

#阿里云电话主账号AccessKey的ID

ALY\_DH\_AccessKeyId=xxxxxxxxxxxxxxxxxxxxxxxxxx

#阿里云电话接口密钥

ALY\_DH\_AccessSecret=xxxxxxxxxxxxxxxxxxxxxxxxxx

#阿里云电话被叫显号,必须是已购买的号码

ALY\_DX\_CalledShowNumber=xxxxxxxxxx

#阿里云电话文本转语音(TTS)模板ID

ALY\_DH\_TtsCode=xxxxxxxxxx

#-----↓容联云接口-----

#是否开启容联云电话告警通道,可同时开始多个通道0为关闭,1为开启

open-rlydh=0

#容联云基础接口地址

```
RLY_URL=https://app.cloopen.com:8883/2013-12-26/Accounts/
#容联云后台SID
RLY_ACCOUNT_SID=xxxxxxxxxxx
#容联云api-token
RLY_ACCOUNT_TOKEN=xxxxxxxxxxx
#容联云app_id
RLY_APP_ID=xxxxxxxxxxxxxxxxx

#-----↓邮件配置-----
#是否开启邮件
open-email=0
#邮件发件服务器地址
Email_host=smt.qq.com
#邮件发件服务器端口
Email_port=465
#邮件帐号
Email_user=xxxxxxx@qq.com
#邮件密码
Email_password=xxxxxx
#邮件标题
Email_title=运维告警
#默认发送邮箱
Default_emails=xxxxx@qq.com,xxxxx@qq.com

#-----↓七陌云接口-----
#是否开启七陌短信告警通道,可同时开始多个通道0为关闭,1为开启
open-7moordx=0
#七陌账户ID
7MOOR_ACCOUNT_ID=Nxxx
#七陌账户APISecret
7MOOR_ACCOUNT_APISECRET=xxx
#七陌账户短信模板编号
7MOOR_DX_TEMPLATENUM=n
#注意：七陌短信变量这里只用一个var1，在代码里写死了。
#-----
#是否开启七陌webcall语音通知告警通道,可同时开始多个通道0为关闭,1为开启
open-7moordh=0
#请在七陌平台添加虚拟服务号、文本节点
#七陌账户webcall的虚拟服务号
7MOOR_WEBCALL_SERVICENO=xxx
# 文本节点里被替换的变量，我配置的是text。如果被替换的变量不是text，请修改此配置
7MOOR_WEBCALL_VOICE_VAR=text

#-----↓telegram接口-----
#是否开启telegram告警通道,可同时开始多个通道0为关闭,1为开启
open-tg=0
#tg机器人token
TG_TOKEN=xxxxx
#tg消息模式 个人消息或者频道消息 0为关闭(推送给个人)，1为开启(推送给频道)
TG_MODE_CHAN=0
#tg用户ID
TG_USERID=xxxxx
#tg频道name或者id，频道name需要以@开始
TG_CHANNAME=xxxxx
```

```

#tg api地址，可以配置为代理地址
#TG_API_PROXY="https://api.telegram.org/bot%s/%s"

#-----↓workwechat接口-----
#是否开启workwechat告警通道，可同时开始多个通道0为关闭，1为开启
open-workwechat=0
# 企业ID
WorkWechat_CropID=xxxxx
# 应用ID
WorkWechat_AgentID=xxxx
# 应用secret
WorkWechat_AgentSecret=xxxx
# 接受用户
WorkWechat_ToUser="zhangsan|lisi"
# 接受部门
WorkWechat_ToParty="ops|dev"
# 接受标签
WorkWechat_ToTag=""
# 消息类型，暂时只支持markdown
# WorkWechat_Msgtype = "markdown"

#-----↓百度云接口-----
#是否开启百度云短信告警通道，可同时开始多个通道0为关闭，1为开启
open-baidudx=0
#百度云短信接口AK(ACCESS_KEY_ID)
BDY_DX_AK=xxxxxx
#百度云短信接口SK(SECRET_ACCESS_KEY)
BDY_DX_SK=xxxxxx
#百度云短信ENDPOINT（ENDPOINT参数需要用指定区域的域名来进行定义，如服务所在区域为
北京，则为）
BDY_DX_ENDPOINT=http://smsv3.bj.baidubce.com
#百度云短信模版ID，根据自己审核通过的模版来填写（模版支持一个参数code：如prometheus
告警:{code}）
BDY_DX_TEMPLATE_ID=xxxxxx
#百度云短信签名ID，根据自己审核通过的签名来填写
TXY_DX_SIGNATURE_ID=xxxxxx

#-----↓百度Hi(如流)-----
#是否开启百度Hi(如流)告警通道，可同时开始多个通道0为关闭，1为开启
open-ruliu=0
#默认百度Hi(如流)机器人地址
BDRL_URL=https://api.im.baidu.com/api/msg/groupmsgsend?
access_token=xxxxxxxxxxxxxx
#百度Hi(如流)群ID
BDRL_ID=123456

#-----↓bark接口-----
#是否开启telegram告警通道，可同时开始多个通道0为关闭，1为开启
open-bark=0
#bark默认地址，建议自行部署bark-server
BARK_URL=https://api.day.app
#bark key，多个key使用分割
BARK_KEYS=xxxxxx
# 复制，推荐开启
BARK_COPY=1

```

```
# 历史记录保存,推荐开启
BARK_ARCHIVE=1
# 消息分组
BARK_GROUP=PrometheusAlert
kind: ConfigMap
metadata:
  name: prometheus-alert-center-conf
  namespace: kube-system
```

根据实际需要进行配置文件的修改,这里我们对接飞书,主要有以下几个参数需要修改

```
login_user=prometheusalert ## UI登陆用户名
login_password=prometheusalert ##UI登陆密码
open-feishu=1 ##修改为1,启用飞书告警
fsurl=https://open.feishu.cn/open-apis/bot/hook/xxxxxxxxxx ##飞书机器人
webhook地址(也可以在后续页面中填写)
```

## 部署工作负载

通过yaml部署PrometheusAlert工作负载和Nodeport类型的Service以用于访问PrometheusAlert UI(SUSE Rancher UI可以直接导入);

```
apiVersion: apps/v1
kind: Deployment
metadata:
  labels:
    app: prometheus-alert-center
    alertname: prometheus-alert-center
  name: prometheus-alert-center
  namespace: kube-system
spec:
  replicas: 1
  selector:
    matchLabels:
      app: prometheus-alert-center
      alertname: prometheus-alert-center
  strategy:
    rollingUpdate:
      maxSurge: 1
      maxUnavailable: 0
    type: RollingUpdate
  template:
    metadata:
      labels:
        app: prometheus-alert-center
        alertname: prometheus-alert-center
    spec:
      containers:
```

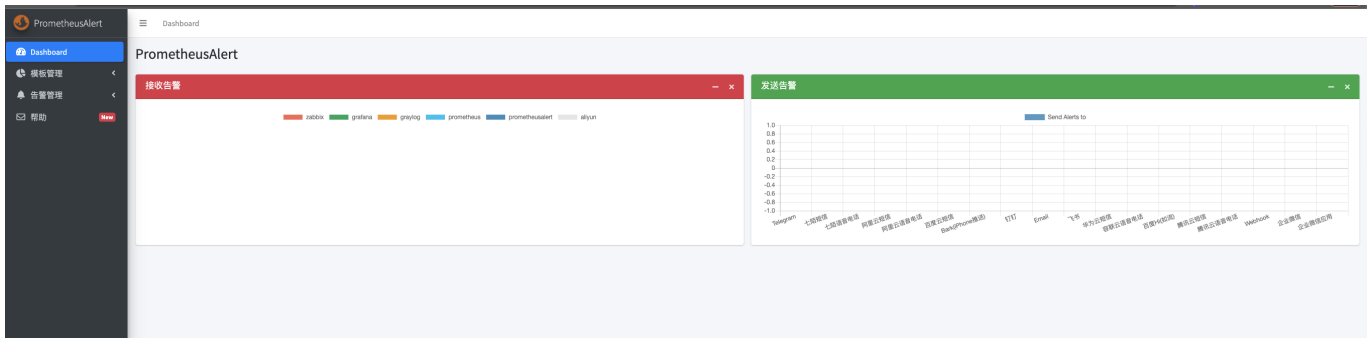


```

- env:
  - name: TZ
    value: Asia/Shanghai
  image: feiyu563/prometheus-alert
  imagePullPolicy: IfNotPresent
  name: prometheus-alert-center
  resources:
    limits:
      cpu: 500m
      memory: 512Mi
    requests:
      cpu: 200m
      memory: 128Mi
  volumeMounts:
    - mountPath: /app/conf/
      name: vol1
  dnsPolicy: ClusterFirst
  restartPolicy: Always
  schedulerName: default-scheduler
  securityContext: {}
  terminationGracePeriodSeconds: 30
  volumes:
    - configMap:
        defaultMode: 256
        name: prometheus-alert-center-conf
        optional: false
        name: vol1
----
apiVersion: v1
kind: Service
metadata:
  name: prometheus-alert-center
  namespace: kube-system
spec:
  type: NodePort
  selector:
    app: prometheus-alert-center
  ports:
    - port: 8080
      targetPort: 8080
      nodePort: 30080

```

- 部署完成后通过访问[http://<node\\_ip>:30080](http://<node_ip>:30080)即可访问PrometheusAlert UI，用户名和密码为Configmap中的配置；至此，PrometheusAlert全家桶部署完毕，除了以上基础配置以外PrometheusAlert全家桶还有更多的部署方式和配置，比如说使用后端mysql存储进行模版的存储等，更多部署细节可以查看项目文档链接：<https://feiyu563.gitbook.io/prometheusalert/base-install>



## 创建自定义告警模版

PrometheusAlert全家桶部署完成后，因为使用SUSE Rancher通知发送出去的Webhook消息与原生的有所不同，因此我们需要添加PrometheusAlert全家桶中的自定义告警模版；

- 注意：SUSE Rancher告警通知模版和PrometheusAlert全家桶自定义模版需要有一定的Go template 基础；

## 添加自定义PrometheusAlert告警模版

在PrometheusAlert全家桶中添加自定义模版，以下内容必填项

- 模版名称
- 模版类型
- 模版内容
- 模版用途

这里我们选择模版类型为飞书，模版用途为Prometheus，模版内容如下

```
{{ $var := .externalURL }}{{ range $k,$v:=.alerts }}
{{if eq $v.status "resolved"}}
**[Rancher容器云平台恢复告警信息]({{$v.labels.server_url}})**
*[{{{$v.labels.alertname}}]}]({{$var}})*
告警级别: {{{$v.labels.severity}}}
开始时间: {{{$TimeFormat $v.startsAt "2006-01-02 15:04:05 UTC"}}}
结束时间: {{{$TimeFormat $v.endsAt "2006-01-02 15:04:05 UTC"}}}
集群ID: {{{$v.labels.cluster_name}}}
故障主机IP: {{{$v.labels.instance}}}
PromQL: {{{$v.alert.expression}}}
触发告警持续时间: {{{$v.labels.duration}}}
当前值为: **[{{{$v.annotations.current_value}}}]**
{{else}}
**[Rancher容器云平台告警信息]({{{$v.labels.server_url}}})**
*[{{{$v.labels.alertname}}]}]({{$var}})*
告警级别: {{{$v.labels.severity}}}
开始时间: {{{$TimeFormat $v.startsAt "2006-01-02 15:04:05 UTC"}}}
{{if eq $v.endsAt "0001-01-01T00:00:00Z"}}
结束时间: {{ printf "告警当前仍然存在!" }}
{{else}}
结束时间: {{{$TimeFormat $v.endsAt "2006-01-02 15:04:05 UTC"}}}
{{end}}
集群ID: {{{$v.labels.cluster_name}}}
```

```
故障主机IP: {{$v.labels.instance}}
PromQL: {{$v.labels.expression}}
触发告警持续时间: {{$v.labels.duration}}
当前值为: **{{$v.annotations.current_value}}**
{{end}}
{{ end }}
```

添加自定义模版

\*模版名称  
模版名称不可重复且必须为英文字母，如prometheusTpl

\*模版类型  
飞书

\*模版用途  
Prometheus

\*模版内容:  
[[ \$var := .externalURL | range \$k, \$v := .alerts ]]  
[[ if eq \$v.status "resolved" ]]  
"{{ \$var }}告警平台告警信息{{ \$v.labels.server\_url }}"  
"{{ \$v.labels.alertname }}"{{ \$var }}"  
告警级别: {{ \$v.labels.severity }}  
开始时间: {{ TimeFormat \$v.startAt "2006-01-02 15:04:05 UTC" }}  
结束时间: {{ TimeFormat \$v.endAt "2006-01-02 15:04:05 UTC" }}  
集群ID: {{ \$v.labels.cluster\_name }}  
故障主机IP: {{ \$v.labels.instance }}  
PromQL: {{ \$v.labels.expression }}

消息协议/JSON内容:  
可在PrometheusAlert日志中查找

飞书机器人地址  
c

@某人  
钉钉@使用的手机号码，企业微信机器人@使用的用户账号，飞书V2 @使用的用户Email邮箱

进行模版测试前请先点击保存模版

特别提醒：自定义模版仅支持 /prometheusalert 接口，其他接口均为固定模版接口，不支持自定义模版

- 这里的模版内容均为go template编写，在本文中仅以SUSE Rancher发出的默认webhook消息作为示例，对于有go template基础或者SUSE Rancher使用基础的用户，可以基于SUSE Rancher企业版的自定义告警功能和自定义告警通知模版功能对告警消息进行深度定制，然后在PrometheusAlert全家桶自定义模版中定义更高级的告警消息；

## SUSE Rancher告警对接

首先在集群/项目--工具--通知中添加Webhook类型的通知，URL填写如下：

```
http://<PrometheusAlert_url>/prometheusalert?
type=fs&tpl=RancherTofs&fsurl=https://open.feishu.cn/open-
apis/bot/v2/hook/xxxxxxxxxx
```

- <PrometheusAlert\_url> PrometheusAlert地址
- typt=fs 类型为飞书
- tpl=RancherTofs 上一步创建的PrometheusAlert模版名称
- fsurl 飞书机器人的webhook地址

- PrometheusAlert有自己的接口格式，更多接口细节请参考PrometheusAlert文档，本文仅以飞书举例；

Slack

电子邮箱

PagerDuty

Webhook

企业微信

钉钉

Microsoft Teams

阿里云短信

ServiceNow

名称 \*

添加描述

RancherTofs告警通知

URL \*

http://172.16.100.1:30080/prometheusalert?type=fs&tpl=RancherTofs&fsurl=https://open.feishu.cn/open-apis/bot/v2/hook/875a42d

Proxy URL

例如: http://my-proxy/

Bearer Token

Bearer token

发送已解决告警

☒ 启用

设置已验证

添加取消

这里需要注意的是，与以往的对接方式不通，点击测试通过后飞书并不会收到测试消息，这是因为消息格式体与PrometheusAlert中的自定义模版不匹配，我们可以通过查看PrometheusAlert组件的日志查看webhook消息是否发送到了PrometheusAlert组件，日志内容如下：

```
[{"labels":{"test_msg":"Webhook setting validated"},"annotations":null,"startsAt":"0001-01-01T00:00:00Z","endsAt":"0001-01-01T00:00:00Z","generatorURL":""}]
```

## SUSE Rancher配置告警通知

上述步骤完成后，我们就可以对告警进行通知配置了，在SUSE Rancher UI上进入集群/项目--工具--告警中我们可以对告警/告警组进行通知配置，在接收者中配置刚刚添加的Webhook通知

告警

到

RancherTofs告警通知 (Webhook)

URL

http://172.16.100.1:30080/prometheusalert?type=fs&tpl=RancherTofs&fsurl=https://open.feishu.cn/open-apis/bot/v2/hook/875a42d

默认情况下使用通知程序中的设置。您可以用另一个值覆盖它。

添加告警规则

添加接收者

保存后触发告警后就可以发送到飞书进行告警通知了；



## 故障排查

- SUSE Rancher UI测试不通过：检查Rancher Server服务到PrometheusAlert服务的网络是否能通讯；
- 飞书收不到告警信息：检查PrometheusAlert组件日志，查看是否正常发送告警消息到飞书；
- 飞书收到的告警消息有空白字段：检查PrometheusAlert中的自定义模版，通过测试发送检查告警内容是否错误；

## 总结/感悟

PrometheusAlert全家桶对市面上快速迭代的企业沟通工具的发送告警支持，无论通过SUSE Rancher的告警功能或者其他的监控平台都能很好的实现不同客户的自定义需求。使用门槛对于稍微了解Webhook消息和go template基础的用户来说都能很快的上手，但是想要更好的使用PrometheusAlert全家桶可能需要对监控告警体系和go template技能有所要求。

监控/告警体系的建设在本文作者看来不是可以一蹴而就的，而是需要一定的时间慢慢累计，不断的调整。“狼来了”的告警不应该被视为工作上的错误，反而这种“狼来了”的告警能够促进监控/告警体系的完善。监控/告警从最开始的后知后觉，慢慢演变为先知先觉，这其中有很长的路需要走。部分团队可能会觉得运维工程师无非就是上线部署，重启服务器，删库跑路，但正是有了这些“删库跑路”的运维工程师，通过正确的预警/监控/告警和深夜里处理告警的精神，才避免了一次又一次重大的事故。善待运维，完善监控体系，对技术永怀敬畏之心；

## 参考链接

- prometheusalert文档：<https://feiyu563.gitbook.io/prometheusalert/>
- prometheusalert项目：<https://github.com/feiyu563/PrometheusAlert/>
- SUSE Rancher通知：[https://docs.rancher.cn/docs/rancher2.5/monitoring-alerting/configuration/receiver/\\_index#webhook](https://docs.rancher.cn/docs/rancher2.5/monitoring-alerting/configuration/receiver/_index#webhook)
- SUSE Rancher告警表达式参考：[https://docs.rancher.cn/docs/rancher2.5/monitoring-alerting/expression/\\_index](https://docs.rancher.cn/docs/rancher2.5/monitoring-alerting/expression/_index)