

Prometheus+Grafana全栈监控告警方案

概述

技术栈：Prometheus + Grafana + Node Exporter + Alertmanager

项目目标：

- 部署Prometheus监控服务，采集服务器CPU/内存/磁盘等重要指标
- 设计Grafana可视化看板，实现核心指标实时仪表盘
- 配置Alertmanager邮件告警规则，系统异常状态实时通过QQ邮箱等方式告警

部署流程

1. 下载工具

- prometheus: <https://github.com/prometheus/prometheus/releases/download/v2.47.0/prometheus-2.47.0.linux-amd64.tar.gz>
- pushgateway: <https://github.com/prometheus/pushgateway/releases/download/v1.4.2/pushgateway-1.4.2.linux-amd64.tar.gz>
- node-exporter: https://github.com/prometheus/node_exporter/releases/download/v1.3.1/node_exporter-1.3.1.linux-amd64.tar.gz
- grafana: <https://repo.huaweicloud.com/grafana/8.4.7/grafana-enterprise-8.4.7.linux-amd64.tar.gz>
- Alertmanager: <https://github.com/prometheus/alertmanager/releases/download/v0.24.0/alertmanager-0.24.0.linux-amd64.tar.gz>

在 /opt/prometheus_env/ 目录（在 /opt 下mkdir）下，通过wget命令下载上述工具，并解压缩如下：

```
tar -zxvf prometheus-2.47.0.linux-amd64.tar.gz
tar -zxvf pushgateway-1.4.2.linux-amd64.tar.gz
tar -zxvf pushgateway-1.4.2.linux-amd64.tar.gz
tar -zxvf alertmanager-0.24.0.linux-amd64.tar.gz
tar -zxvf grafana-enterprise-8.4.7.linux-amd64.tar.gz
```

2. 配置

(1) 修改prometheus.yml配置文件

```
cd /opt/prometheus_env/prometheus-2.47.0.linux-amd64
sudo gedit prometheus.yml
//或使用vim, vi prometheus.yml
```

内容如下：

```
# my global config
global:
  scrape_interval: 15s # Set the scrape interval to every 15 seconds. Default is every 1 minute.
```

```

    evaluation_interval: 15s # Evaluate rules every 15 seconds. The default is
every 1 minute.
    # scrape_timeout is set to the global default (10s).

# Alertmanager configuration
alerting:
  alertmanagers:
    - static_configs:
      - targets:
        - 127.0.0.1:9093

rule_files:
  - "/opt/prometheus_env/prometheus-2.47.0.linux-amd64/alarm_rules.yml"

scrape_configs:
  - job_name: 'prometheus'
    static_configs:
      - targets: ['127.0.0.1:9090']
        labels:
          instance: 'prometheus'
  - job_name: 'linux'
    static_configs:
      - targets: ['127.0.0.1:9100']
        labels:
          instance: 'localhost'
  - job_name: 'pushgateway'
    static_configs:
      - targets: ['127.0.0.1:9091']
        labels:
          instance: 'pushgateway'

```

(2) 新增alarm_rules.yml文件

```

cd /opt/prometheus_env/prometheus-2.47.0.linux-amd64
vi alarm_rules.yml

```

可以自定义规则，以下给出一个规则示例：

```

groups:
- name: node
  rules:
  - alert: server_status1
    expr: up{} == 0
    for: 15s
    labels:
      severity: warning
    annotations:
      summary: "机器{{ $labels.instance }} 挂了"
      description: "请立即查看问题!"

```

(3) 修改alertmanager.yml文件

```
cd /opt/prometheus_env/alertmanager-0.24.0.linux-amd64
sudo gedit alertmanager.yml
```

以下给出配置QQ邮箱告警的内容，其中授权码的获取方式可参考[链接](#)：

```
global:
  resolve_timeout: 5m
  # 配置邮件发送信息
  smtp_smarthost: 'smtp.qq.com:465'
  smtp_from: '发送者@qq.com'
  smtp_auth_username: '发送者@qq.com'
  smtp_auth_password: 'xxxxxxx' #发送者邮箱的授权码
  smtp_hello: 'qq.com'
  smtp_require_tls: false #不启用tls
route:
  #这里的标签列表是接收到报警信息后的重新分组标签
  group_by: ['alertname','severity']
  #当一个新的报警分组被创建后，需要等待至少group_wait时间来初始化通知
  group_wait: 30s
  #当第一个报警发送后，等待'group_interval'时间来发送新的一组报警信息。
  group_interval: 40s
  #如果一个报警信息已经发送成功了，等待'repeat_interval'时间来重新发送他们
  repeat_interval: 30s
  #默认的receiver：如果一个报警没有被一个route匹配，则发送给默认接收器
  receiver: 'email'
  #上面所有的属性都由所有子路由继承，并且可以在每个子路由上进行覆盖。
  routes:
    - receiver: email
  #使用标签匹配报警，这里匹配告警规则中severity为warning的告警信息
  match:
    severity: warning
receivers:
  - name: 'email'
    email_configs:
      - to: '接收方@qq.com'
        send_resolved: true
inhibit_rules:
  - source_match:
      severity: 'critical'
    target_match:
      severity: 'warning'
    equal: ['alertname', 'severity', 'instance']
```

(4) 修改defaults.ini文件

```
sudo gedit /opt/prometheus_env/grafana-8.4.7/conf/defaults.ini
```

找到内容并修改如下，使grafana可以匿名访问：

```
##### Anonymous Auth #####
[auth.anonymous]
# enable anonymous access
enabled = true
```

(5) 增加service文件

```
cd /usr/lib/systemd/system
//system目录可能需要新建
```

需要增加五个service文件，内容如下：

- prometheus.service

```
[Unit]
Description=Prometheus Service
After=network.target

[Service]
ExecStart=/opt/prometheus_env/prometheus-2.47.0.linux-amd64/prometheus \
--config.file=/opt/prometheus_env/prometheus-2.47.0.linux-amd64/prometheus.yml \
--web.read-timeout=5m \
--web.max-connections=10 \
--storage.tsdb.retention=15d \
--storage.tsdb.path=/prometheus/data \
--query.max-concurrency=20 \
--query.timeout=2m

User=root
[Install]
WantedBy=multi-user.target
```

- grafana.service

```
[Unit]
Description=Grafana
After=network.target

[Service]
ExecStart=/opt/prometheus_env/grafana-8.4.7/bin/grafana-server \
--config=/opt/prometheus_env/grafana-8.4.7/conf/defaults.ini \
--homepath=/opt/prometheus_env/grafana-8.4.7

[Install]
WantedBy=multi-user.target
```

- alertmanager.service

```
[Unit]
Description=Prometheus alertmanager
After=local-fs.target network-online.target network.target
Wants=local-fs.target network-online.target network.target

[Service]
ExecStart=/opt/prometheus_env/alertmanager-0.24.0.linux-amd64/alertmanager \
--storage.path=/opt/prometheus_env/alertmanager-0.24.0.linux-amd64/data \
--config.file=/opt/prometheus_env/alertmanager-0.24.0.linux-
amd64/alertmanager.yml
Restart=on-failure
```

```
User=root
[Install]
WantedBy=multi-user.target
```

- pushgateway.service

```
[Unit]
Description=Prometheus Push Gateway
After=network.target

[Service]
ExecStart=/opt/prometheus_env/pushgateway-1.4.2.linux-amd64/pushgateway

User=root
[Install]
WantedBy=multi-user.target
```

- node_exporter.service

```
[Unit]
Description=Prometheus Node Exporter
After=network.target

[Service]
ExecStart=/opt/prometheus_env/node_exporter-1.3.1.linux-amd64/node_exporter

User=root
[Install]
WantedBy=multi-user.target
```

3.开启服务

重载配置

```
systemctl daemon-reload
```

启动服务

```
systemctl start pushgateway
systemctl start node_exporter
systemctl start prometheus
systemctl start grafana
systemctl start alertmanager
//可以有.service后缀，不影响
```

设置开机自启动

```
systemctl enable pushgateway
systemctl enable node_exporter
systemctl enable prometheus
systemctl enable grafana
systemctl enable alertmanager
```

查看服务状态

```
systemctl status <服务名>
//例如 sudo systemctl status alertmanager
```

查看prometheus工作日志

```
journalctl -u prometheus.service -f
```

系统监控

prometheus

访问 `http://localhost:9090/targets` , 例如 `http://127.0.0.1:9090/targets`

运行效果示例如下:

- targets界面, 显示目标是否启动, 如未启动会有红色标识

Prometheus Alerts Graph Status Help

Targets

All scrape pools All Unhealthy Collapse All Filter by endpoint or labels Unknown Unhealthy Healthy

linux (1/1 up) show less

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://127.0.0.1:9100/metrics	UP	instance="localhost" job="linux"	4.514s ago	23.313ms	

prometheus (1/1 up) show less

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://127.0.0.1:9090/metrics	UP	instance="prometheus" job="prometheus"	742.000ms ago	8.971ms	

pushgateway (1/1 up) show less

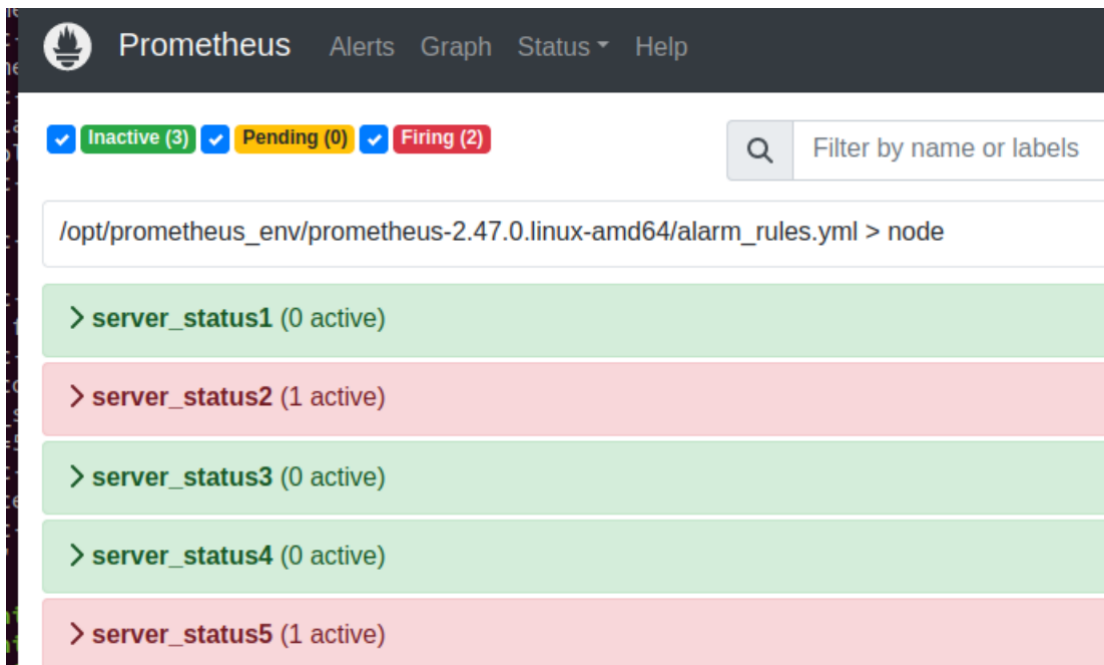
Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://127.0.0.1:9091/metrics	UP	instance="pushgateway" job="pushgateway"	12.766s ago	13.507ms	

- Status -> rules界面, 查看告警规则是否有被下发

Rules

node			11.820s ago	0.341ms
Rule	State	Error	Last Evaluation	Evaluation Time
alert: server_status1 expr: up == 0 for: 15s labels: severity: warning annotations: description: 请立即查看问题! summary: 机器{{ \$labels.instance }} 挂了	OK		11.824s ago	0.323ms

- Alerts栏目, 查看告警情况

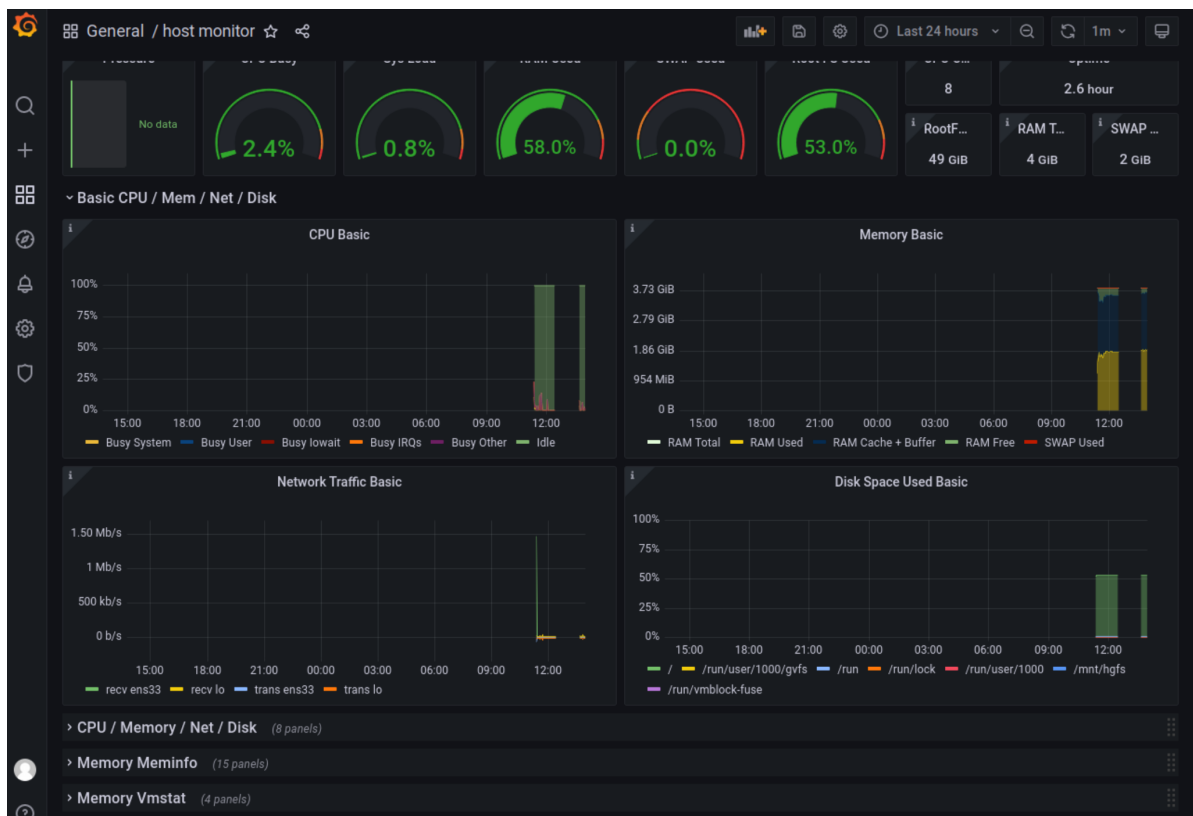


grafana

访问 <http://127.0.0.1/3000>

- 注意，grafana需要先登录才能使用，初始账号密码均为admin

创建dashboard时，可自定义指标，也可以选择从库中import模板，例如



alertmanager

访问 <http://127.0.0.1/9093>，可以查看其发出的告警

AlertmanagerAlertsSilencesStatusHelp

New Silence

FilterGroup

Receiver: AllSilencedInhibited

+

Silence

Custom matcher, e.g. env="production"

+ Expand all groups

+ alertname="server_status2" severity="warning" + 1 alert

+ alertname="server_status5" severity="warning" + 1 alert

告警邮件

当告警规则被触发时，目标邮箱可以收到类似下述内容的邮件：

1 alert for alertname=server_status2 severity=warning

View In Alertmanager

[1] Firing

Labels
alertname = server_status2
instance = localhost
job = linux
severity = warning

Annotations
description = 请立即查看问题!
summary = 机器localhost 内存大于50%

[Source](#)