



链滴

基于 SkyWalking 的分布式跟踪系统 - 环境搭建

作者: [jianzh5](#)

原文链接: <https://ld246.com/article/1574728983725>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

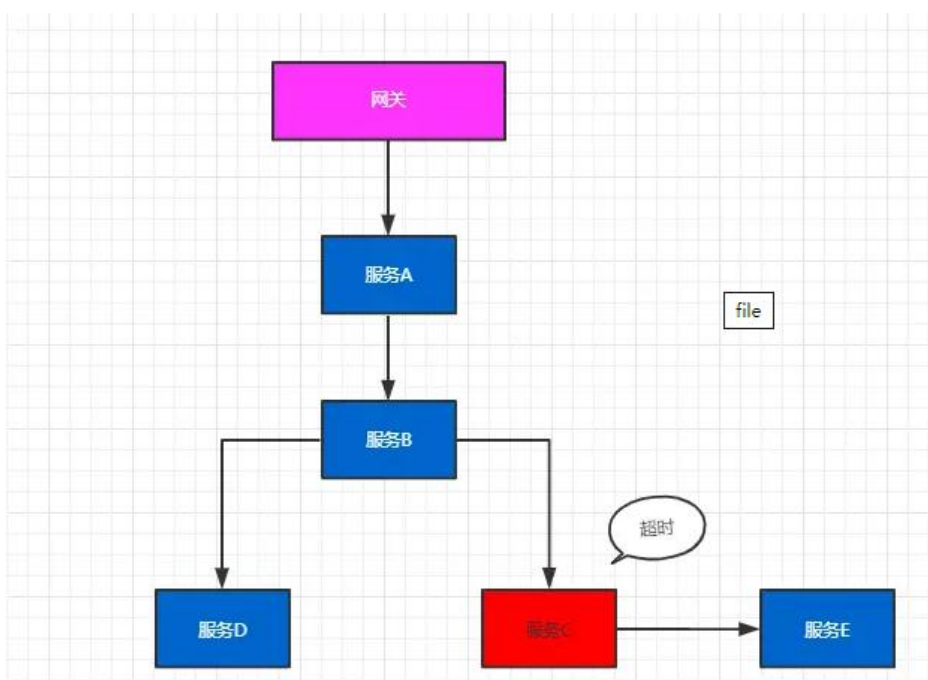


前面的几篇文章我们聊了基于Metrics的监控Prometheus，利用Prometheus和Grafana可以全方位控你的服务器及应用的性能指标，在出现异常时利用Alertmanager告警及时通知运维处理。今天我聊聊基于Trace的分布式跟踪系统 - SkyWalking

初识SkyWalking

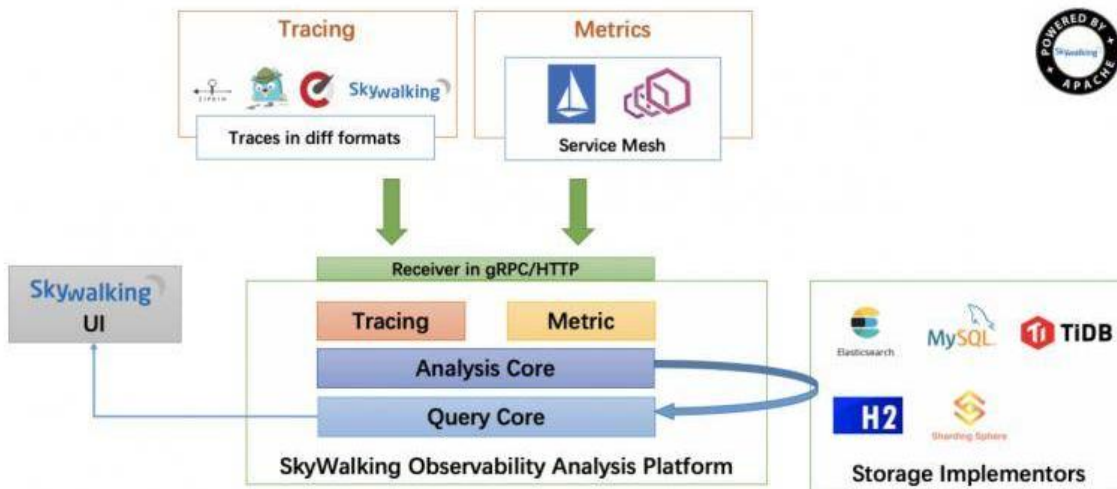
应用场景

随着微服务架构的流行，一些微服务架构下的问题也会越来越突出，比如一个请求会涉及多个服务，服务本身可能也会依赖其他服务，整个请求路径就构成了一个网状的调用链，而在整个调用链中一旦个节点发生异常，整个调用链的稳定性就会受到影响，如果没有及时处理很有可能会造成整个系统崩



面对以上情况，我们就需要一些可以帮助理解系统行为、用于分析性能问题的工具，以便发生故障的时候，能够快速定位和解决问题。

架构



SkyWalking 逻辑上分为四部分: 探针, 平台后端, 存储和用户界面。

• 探针

主要负责从客户端收集数据，将数据转换成SkyWalking适用的格式，探针对客户端程序没有任何代侵入，使用起来简单方便，使用如下命令即可完成对应用的监控

```
java -javaagent:/path/skywalking-agent.jar -jar youApp.jar
```

• 平台后端 (OAP Server)

主要用于数据聚合, 数据分析以及驱动数据流从探针到用户界面的流程。通过 gRPC/Http 收集客户端gent的采集信息，Http默认端口 12800，gRPC默认端口 11800。

• 存储

SkyWalking支持很多存储：H2（用作演示环境）、MySQL（当数据量大时检索性能下降很厉害）、S（主流生产级别的存储）

• 用户界面

炫酷的界面，调用请求监控一目了然。

安装配置

安装

直接从[官网](#)下载最新的安装包，并上传到服务器解压。解压后的文件如下：

```
drwxrwxr-x. 8 elkuser 1002 143 9月 9 02:14 agent
drwxr-xr-x. 2 root root 121 10月 26 22:48 bin
drwxr-xr-x. 2 root root 224 10月 26 21:15 config
-rwxrwxr-x. 1 elkuser 1002 28903 9月 9 02:04 LICENSE
drwxrwxr-x. 3 elkuser 1002 4096 10月 26 21:15 licenses
drwxr-xr-x. 2 root root 50 10月 26 22:49 logs
-rwxrwxr-x. 1 elkuser 1002 31850 9月 9 02:04 NOTICE
drwxrwxr-x. 2 elkuser 1002 12288 9月 9 02:22 oap-libs
-rw-rw-r--. 1 elkuser 1002 1978 9月 9 02:04 README.txt
drwxr-xr-x. 2 root root 53 10月 26 21:15 webapp
```

(需要提前准备好JAVA(1.8)和ES(6.x)的环境。)

关注一下几个重要的目录:

- agent: 代理模块 (探针)
- bin: 启动脚本 (包括UI和OAP SERVER)
- config: 后端相关配置
- webapp: UI界面

配置

- 存储相关配置

打开`application.yml`, 修改`storage`相关配置。关闭H2, 打开ES, 然后启动 `./bin/startup.sh`

storage:

elasticsearch:

```
nameSpace: ${SW_NAMESPACE:"elk-online"}
clusterNodes: ${SW_STORAGE_ES_CLUSTER_NODES:192.168.136.129:9200}
protocol: ${SW_STORAGE_ES_HTTP_PROTOCOL:"http"}
```

- agent 配置

将`agent`文件夹从服务器上拷贝出来, 放在客户端服务器。打开`agent\config\agent.config`作如下修改

- `agent.service_name`修改成你应用名称: `blog`
- `collector.backend_service`修改成OAP Server地址: `192.168.136.129:11800`

- IDEA配置 (可选)



配置完成后启动你的客户端应用。

效果

← → ↻ 不安全 | 192.168.136.129:8080/topology

Skywalking Rocketbot

仪表盘 拓扑图 追踪 告警

All services

服务详情

名称 blog

类型

Skywalking Rocketbot

仪表盘 拓扑图 追踪 告警

服务 All 实例 All 状态 All 输入名称

追踪ID: 持续时间: 时间范围: 2019-10-27 13:45:00 - 2019-10-27 14:00:00

1 / 1 持续时间

/ShardingSphere/JDBCRootInvoke/

6.78.15721559723090003

起始时间 2019-10-27 13:59:32 持续时间 17 ms 跨度

列表 树结构 表格

blog

/ShardingSphere/JDBCRootInvoke/ (ShardingSphere)

/ShardingSphere/executeSQL/ (ShardingSphere)

Mysql/JDBI/Statement/executeQuery/ (Database - mysql-connector-java)

Skywalking Rocketbot

仪表盘 拓扑图 追踪 告警

Service Dashboard

当前服务 blog 当前实例 blog-pid:13820@janzhang11

Global Service Endpoint Instance

JVM Heap (MB)

JVM Non-Heap (MB)

JVM GC (ms)

JVM GC Count

YoungGC Count 14

OldGC Count 3

JVM CPU (%)

CLR CPU (%)

CLR GC (Count)

CLR HeapMemory (MB)

环境搭建好了，下一步就是全面监控你的应用了，咱们下期有缘再见。