**logstash beats 系列 & fluentd**

## 一、logstash

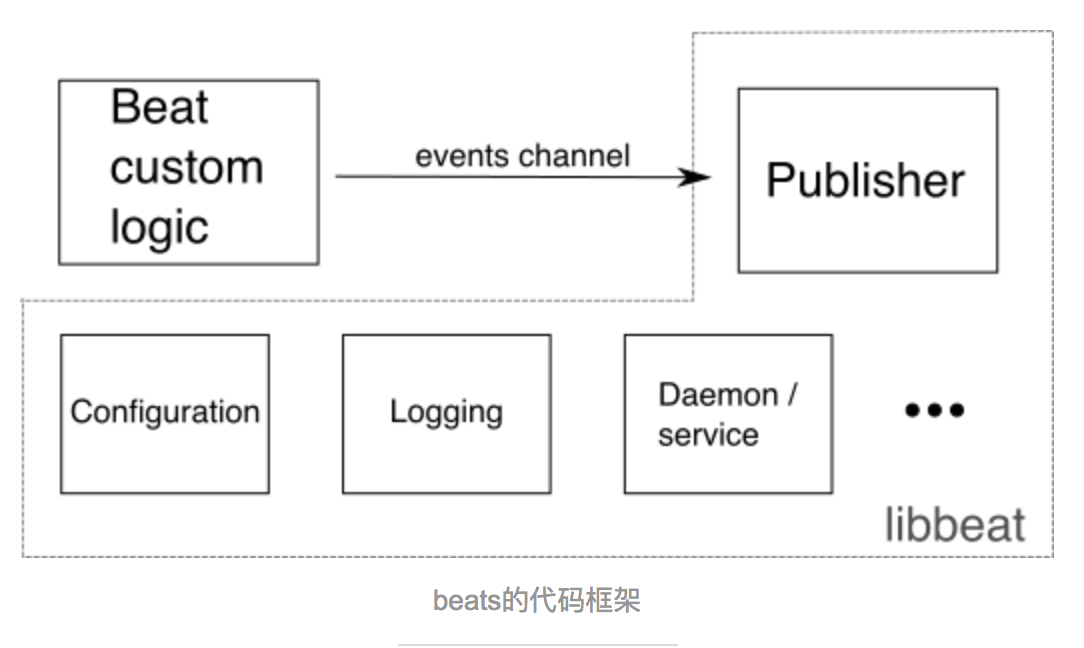
      Logstash: 是一个灵活的数据传输和处理系统，在beats出来之前，还负责进行数据收集。Logstash的任务，就是将各种各样的数据，经过配置转化规则，统一化存入Elasticsearch。使用Ruby开发的Logstash在灵活性上，非常出色。不过性能一直是被诟病的问题。

     由于Logstash在数据收集上并不出色，而且作为agent，性能并不达标。elastic发布了beats系列轻量级采集组件。至此，elastic形成了一个完整的生态链和技术栈，成为大数据市场的佼佼者。

## 二、Elastic Stack Beats 系列

    Beats 是 ELK Stack 技术栈中负责单一用途数据采集并推送给 Logstash 或 Elasticsearch 的轻量级产品。

  Beats 架构：



    beats 是一个使用 Golang 构建的平台，libbeat 是其核心库，用来提供API进行与Elasticsearch，Logstash 的连接，还能配置输入特性和实现信息收集等工作。其中封装一个输出模块(Publisher)，输出模块可以负责将收集到的数据发送给 Logstash 或者 Elasticsearch。由于 go 语言设计有 channel，收集数据的逻辑代码与 Publisher 都是通过 channel 通信的，耦合度的最低的。因此，开发一个收集器，完全不需要知道Publisher 的存在，程序运行的时候自然就“神奇”的把数据发往服务端了。除此之外，还封装了配置文件处理、日志处理、守护化等功能，方便开发者拓展 beats 的能力。

Beats 是一组轻量级采集程序的统称，我们通常常用的几个如下：

   1）  filebeat: 进行文件和目录采集，主要用于收集日志数据。

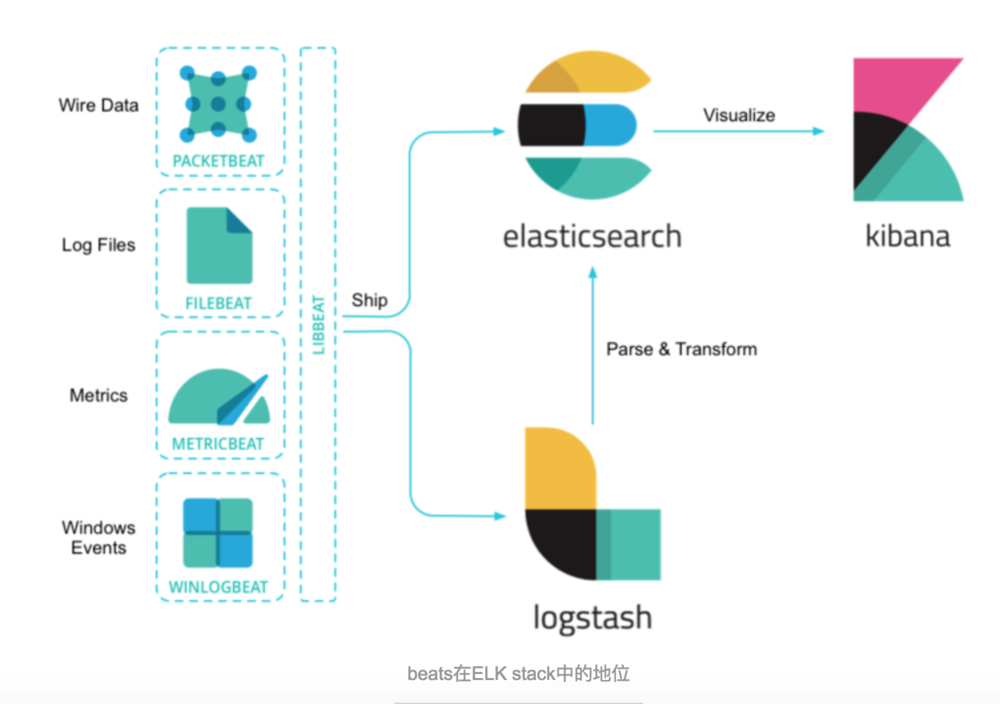
   2） metricbeat: 进行指标采集，指标可以是系统的，也可以是众多中间件产品的，主要用于监控系统和软件的性能。

   3）packetbeat: 通过网络抓包、协议分析，对一些请求响应式的系统通信进行监控和数据收集，可以收集到很多常规方式无法收集到的信息。

  4） Winlogbeat: 专门针对 windows 的 event log 进行的数据采集。

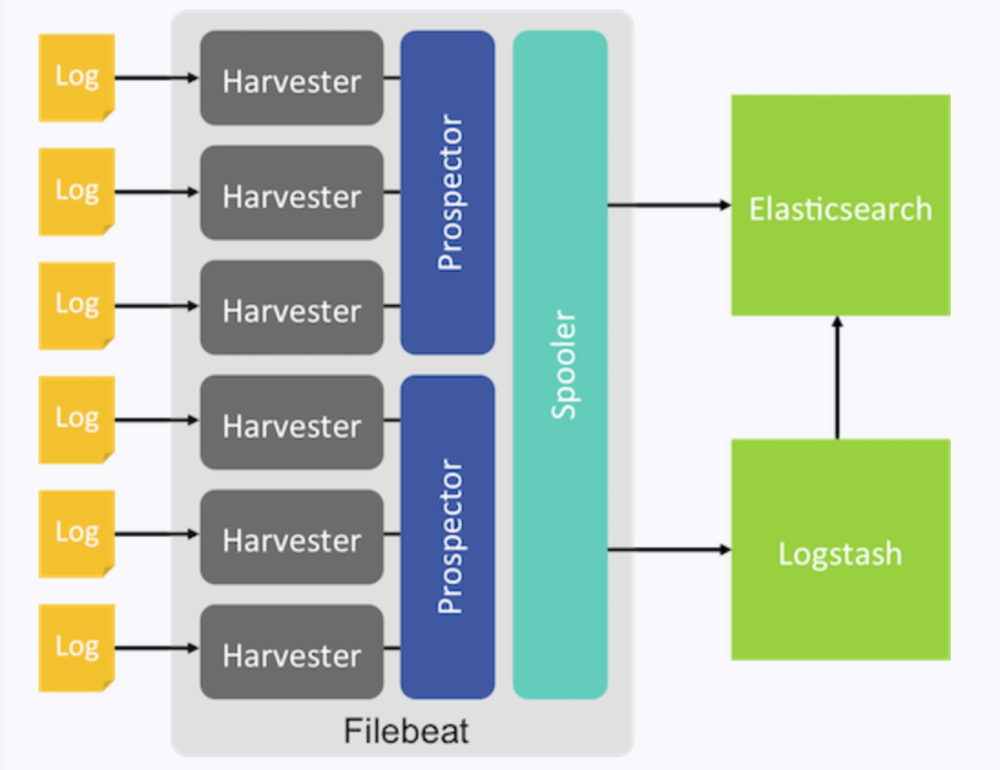
   5） Heartbeat: 系统间连通性检测，比如 icmp, tcp, http 等系统的连通性监控。

   6） 可以通过 beats 生成器来生成自己的 beats



### 1. filebeats

    filebeat是构建于beats之上的，应用于日志收集场景的实现，用来替代 **Logstash Forwarder** 的下一代 **Logstash** 收集器，是为了更快速稳定轻量低耗地进行收集工作，它可以很方便地与 **Logstash** 还有直接与 **Elasticsearch** 进行对接。

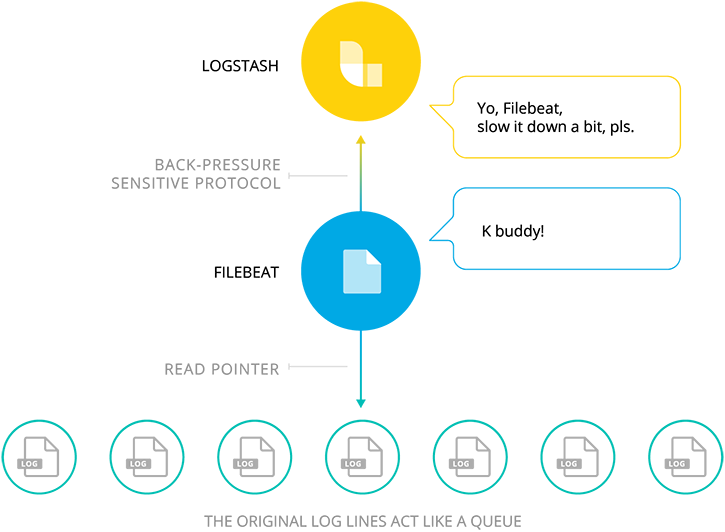


1) 健壮性

filebeat 异常中断重启后会继续上次停止的位置。（通过${filebeat\_home}\data\registry文件来记录日志的偏移量）

2) 智能调节传输速度，防止logstash、es 过载

Filebeat 使用压力敏感协议(backpressure-sensitive)来传输数据，在 logstash 忙的时候，Filebeat 会减慢读取-传输速度，一旦 logstash 恢复，则 Filebeat 恢复原来的速度。



### 2. Metricbeat

  Metricbeat 是一个轻量级的系统级性能指标监控工具。收集CPU，内存，磁盘等系统指标和 Redis，nginx等各种服务的指标。

    1）通过在Linux，Windows，Mac上部署Metricbeat，可以收集cpu，内存，文件系统，磁盘IO，网络IO等统计信息。

    2）支持采集 Apache, NGINX, MongoDB, MySQL, PostgreSQL, Redis, and ZooKeeper等服务的指标。零依赖，只需要在配置文件中启用即可

    3）如果你使用Docker管理你的服务。可以在该主机上单独起一个Metricbeat容器，他通过从proc文件系统中直接读取cgroups信息来收集有关Docker主机上每个容器的统计信息。不需要特殊权限访问Docker API

    4）Metricbeats是ELK Stack全家桶中的一员，可以和ELK无缝协同工作。例如使用Logstash二次处理数据，用Elasticsearch分析，或者用Kibana创建和共享仪表盘。

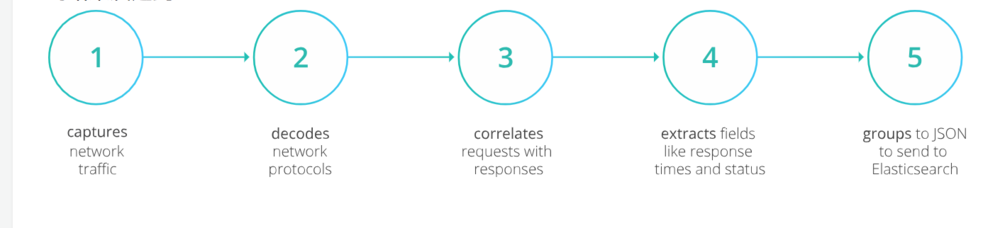
### 3. Packetbeat

Packetbeat 是一个轻量级的网络数据包分析工具。Packetbeat可以通过抓包分析应用程序的网络交互。并且将抓到的数据发送到 Logstash 或者Elasticsearch。

1）Packetbeat 轻松的实时监控并解析像HTTP这样的网络协议。以了解流量是如何经过你的网络。Packetbeat 是被动的，不增加延迟开销，无代码侵入，不干涉其他基础设施。

2）Packetbeat是一个库，支持多种应用程序层协议，如 http、dns、mysal、icmp、postgres、redis 等。

3）Packetbeat可以让你实时在目标服务器上进行抓包-解码-获取请求和响应-展开字段-将json格式的结果发送到Elasticsearch。



4）Packetbeat是ELK Stack全家桶中的一员，可以和ELK无缝协同工作。例如使用Logstash二次处理数据，用Elasticsearch分析，或者用Kibana创建和共享仪表盘。

### 4. Winlogbeat

  Winlogbeat 是一个轻量级的 Windows 事件日志收集工具。将 Windows 事件发送到 Elasticsearc h或者Logstash

  如果你有 Windows 服务器的话，其实可以从 Windows 事件日志中看到很多东西。例如，登陆(4624),登陆失败(4625),插入USB便携设备(4663)或者新装软件(11707)。WinlogBeat可以配置从任何事件日志通道读取并且结构化提供原始事件数据。使得通过 Elasticsearch 过滤和聚合结果变得很容易。

    Winlogbeat是ELK Stack全家桶中的一员，可以和ELK无缝协同工作。例如使用Logstash二次处理数据，用Elasticsearch分析，或者用Kibana创建和共享仪表盘。

### 5. Heartbeat

Heartbeat 是一个心跳检测工具，主要监控服务的可用性。监控给定的地址是否可用(官网原话：对于给定的URL列表，**Heartbeat就问一句，还活着没？活着吱一声。。。**) 可以结合ELK Stack其他产品做进一步的分析

1）不管你是测试同主机服务还是其他网络服务，Heartbeat都可以很轻松的生成正常运行时间和响应时间数据。而且修改配置不需要重启Heartbeat

2）Heartbeat通过ICMP,TCP,和HTTP进行ping，也支持TLS，身份验证（authentication ），和代理(proxies)。由于简单的DNS解析，你可以监控所有负载均衡的服务(原文:You can monitor all the hosts behind a load-balanced server thanks to simple DNS resolution)

3）现如今基础设施，服务和主机经常动态调整。Heartbeat可以修改配置文件后自动加载(原文:Heartbeat makes it easy to automate the process of adding and removing monitoring targets via a simple, file-based interface.)

4）Heartbeat是ELK Stack全家桶中的一员，可以和ELK无缝协同工作。例如使用Logstash二次处理数据，用Elasticsearch分析，或者用Kibana创建和共享仪表盘。

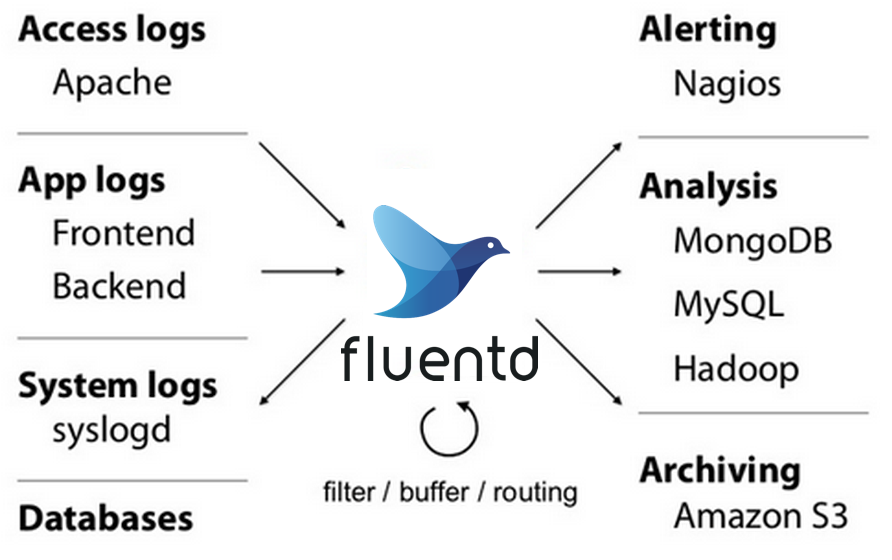
### 6. 创建一个自己的 Beat

    可以根据官方文档，使用 beats 生成器生成自己的 beats

    https://www.elastic.co/cn/blog/build-your-own-beat

## 三、 Fluentd

Fluentd是一个完全开源免费的log信息收集软件，支持超过125个系统的log信息收集。其架构图如图



Fluentd可以分为客户端和服务端两种模块。客户端为安装在被采集系统中的程序，用于读取log文件等信息，并发送到Fluentd的服务端。服务端则是一个收集器。在Fluentd服务端，我们可以进行相应的配置，使其可以对收集到的数据进行过滤和处理，并最终路由到下一跳。下一跳可以是用于存储的数据库，如MongoDB, Amazon S3, 也可以是其他的数据处理平台，比如Hadoop。

#### 1. install & 启动

由于Fluentd的安装较为麻烦，所以业界流行的稳定安装版本其实是有Treasure Data公司提供的td-agent

curl-Lhttps://toolbelt.treasuredata.com/sh/install-ubuntu-trusty-td-agent2.sh|sh

通过start, stop, restart等命令可以启动、关闭和重启Fluentd服务。此时默认的Fluentd配置文件的目录是/etc/td-agent/td-agent.conf文件。

#### 2. Post Sample Logs via HTTP

默认情况下，/etc/td-agent/td-agent.conf文件已经对td-agent进行了基本的配置。可以接收通过HTTP Post的数据，并将其路由、写入到/var/log/td-agent/td-agent.log中。

可尝试通过以下curl命令来尝试post数据。

$curl-XPOST-d'json={"json":"message"}'http://localhost:8888/debug.test

执行之后，可在输出log的最后一行找到我们输入的测试数据。

#### 3. Syntax of Config

在Fluentd中，配置文件非常重要，它定义了Fluentd应该执行的操作。

打开/etc/td-agent/td-agent.conf文件，可以看到配置文件的具体内容。配置文件中基本会出现的配置分为以下几种：

source： 定义输入

match：定义输出的目标，如写入文件，或者发送到指定地点。

filter：过滤，也即事件处理流水线，可在输入和输出之间运行。

system：系统级别的设置。

label：定义一组操作，从而实现复用和内部路由。

@include：引入其他文件，和Java、python的import类似。

1）source： 定义输入

Fluentd支持多输入。每一个输入配置必须包含类型／type，比如tcp数据输入，或者http类型输入。type将指定使用的input plugin。以下的示例中就定义了两个输入源，一个是从24224端口进入的tcp数据流，另一个是从9880端口进入的http数据。

# Receive events from 24224/tcp# This is used by log forwarding and the fluent-cat command@type forward  port 24224#[http://this.host:9880/myapp.access?json={"event":"data](http://this.host:9880/myapp.access?json=%7B%22event%22:%22data)"}@type http  port 9880

Source指定的input插件将带有{tag, time,record} 三个属性的事件／event提交给Fluentd的引擎，完成数据的输入。

2）match：定义输出的目标，如写入文件，或者发送到指定地点

Match配置了数据流的匹配规则和匹配成功后所需执行的动作，和路由表项类似。比如以下的配置中就对匹配myapp.access标签成功的数据包执行file类型动作，将数据写入到路径为/var/log/fluent/access的文件中。

# Match events tagged with "myapp.access" and# store them to /var/log/fluent/access.%Y-%m-%d# Of course, you can control how you partition your data# with the time\_slice\_format option.@type file  path /var/log/fluent/access

标准的动作有file和forward等。File表明写入文件，而forward表明转发到下一跳。

Match Pattern的设计与正常的正则匹配没有区别，具体的分类如下：

\*:匹配tag的某一部分，比如a.\*可以匹配a.b,但a.b.c无法匹配成功。\*\*:匹配0个或者多个tag部分。比如a.\*\*可以匹配a.b,a.b.c{X,Y,Z}：匹配X,Y,orZ,或关系。

此外，他们还可以混用，比如a.{b,c,d}.\*等等。当标签内，有多个匹配模式时，将支持或逻辑的匹配，即只要匹配成功人一个都执行对应的操作。比如：

匹配a和b.匹配a,a.b,a.b.c

3）Logging

Fluentd支持两种类型的logging 配置，一种是 global 全局的，另一种是针对插件的Plugin。

支持的log的输出级别有如下几种：

fatal error warn info debug trace

4）Fluentd有5种类型的插件，分别是：

Input：完成输入数据的读取，由source部分配置

Parser：解析插件

Output：完成输出数据的操作，由match部分配置

Formatter：消息格式化的插件，属于filter类型

Buffer：缓存插件，用于缓存数据

每一个类型都包含着多种的插件，比如input类型就包含了以下几种插件：

in\_forward   in\_http   in\_tail   in\_exec  in\_syslog  in\_scribe

5）Route

Route指的是数据在Fluentd中的处理流水线，一般的流程为

input -> filter -> output

input -> filter -> output with label

即由输入插件获取数据，然后交给filter做处理，然后交给output插件去转发。同时，也支持数据包／事件的重新提交，比如修改tag之后重新路由等等。

reroute event by tags

reroute event by record content

reroute event to other label

#### 4. Use case

此处将选择一个最简单的使用案例来介绍Fluentd的使用。[Fluentd收集Docker的登陆信息](http://docs.fluentd.org/v0.12/articles/docker-logging" \t "_blank)案例。

首先创建一个config file, 用于配置Fluentd的行为，可命名为”in\_docker.conf“。

type forward  port 24224  bind 0.0.0.0type stdout

然后保存文件。使用以下命令运行Fluentd。

$fluentd-cin\_docker.conf

若运行成功则可见输出信息如下所示：

$ fluentd -c in\_docker.conf2015-09-01 15:07:12 -0600 [info]: reading config file path="in\_docker.conf"2015-09-01 15:07:12 -0600 [info]: starting fluentd-0.12.152015-09-01 15:07:12 -0600 [info]: gem 'fluent-plugin-mongo' version '0.7.10'2015-09-01 15:07:12 -0600 [info]: gem 'fluentd' version '0.12.15'2015-09-01 15:07:12 -0600 [info]: adding match pattern="\*.\*" type="stdout"2015-09-01 15:07:12 -0600 [info]: adding source type="forward"2015-09-01 15:07:12 -0600 [info]: using configuration file:@type forward    port 24224    bind 0.0.0.0@type stdout2015-09-01 15:07:12 -0600 [info]: listening fluent socket on 0.0.0.0:24224

然后启动docker containner。如果之前没有安装过docker engine，请读者自行安装。由于docker 本身支持Fluentd收集信息，所以可以通过启动命令来启动Fluentd的client／客户端。

$dockerrun--log-driver=fluentdubuntuecho"Hello Fluentd!"HelloFluentd!

以上命令中的ubuntu为一个镜像，如果本地没有，docker engine会自动下载，并在此镜像上创建容器。启动容器后，查看默认的输出信息文件:/var/log/td-agent/td-agent.log,可在最后一行查看到输出的信息。

总结

Fluentd是一个优秀的log信息收集的开源免费软件，目前以支持超过125种系统的log信息获取。Fluentd结合其他数据处理平台的使用，可以搭建大数据收集和处理平台，搭建商业化的解决方案。

## 四、Fluentd & logstash 对比

logstash支持所有主流日志类型，插件支持最丰富，可以灵活DIY，但性能较差，JVM容易导致内存使用量高。

fluentd支持所有主流日志类型，插件支持较多，性能表现较好。

参考：

https://www.jianshu.com/p/9c26bd9f6ebd

https://juejin.im/entry/58bad514ac502e006bf70517

http://soft.dog/2015/12/24/beats-basic/

http://www.muzixing.com/tag/fluentd.html

作者：zpei0411  
链接：https://www.jianshu.com/p/009286216560  
來源：简书  
简书著作权归作者所有，任何形式的转载都请联系作者获得授权并注明出处。