目 录

1 学习目标

2 ELK简介

3 Beats

4 Logstash

- **学完本课题，你应达成如下目标：**

1. 了解ELK是什么，用途。
2. 掌握beats的用途、种类、使用。
3. 掌握logstash的用途、工作原理、安装、配置。
4. 掌握logstash的使用

目录

1 学习目标

2 ELK简介

3 Beats

4 Logstash

ELK是什么?

Elasticsearch Logstash Kibana 原来称为 ELK Stack ， 现在称为Elastic Stack，加入了 beats 来优化Logstash。

从官网介绍了解它们：https://www.elastic.co/cn/products

ELK的主要用途是什么?

大型分布式系统的日志集中分析。

为什么要用ELK来做日志集中分析?

问1：在生产系统中出现问题，你该如何来定位问题？

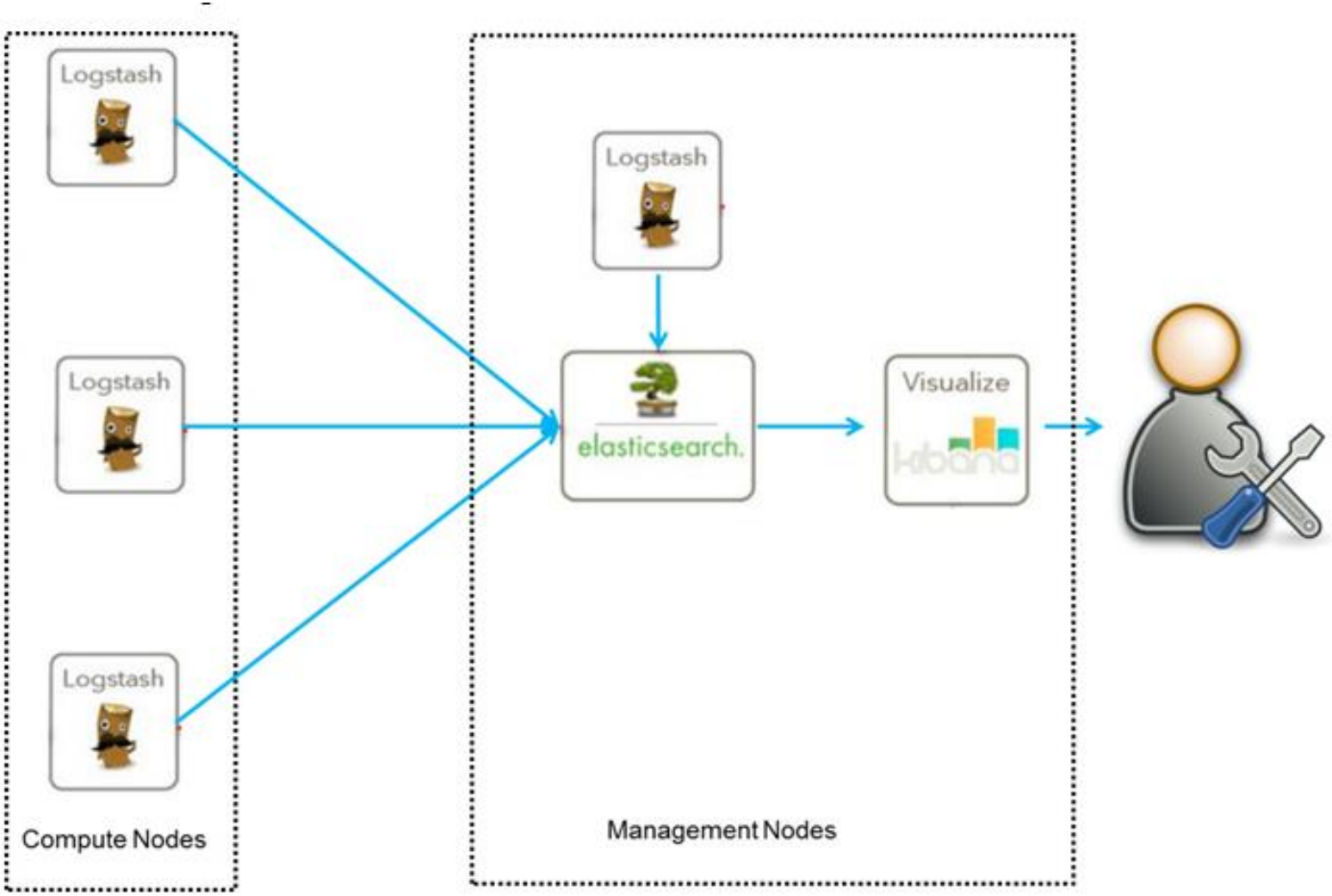
问2：在大型的分布式系统中如出现问题，你该如何定位问题？

一个完整的集中式日志系统，需要包含以下几个主要特点

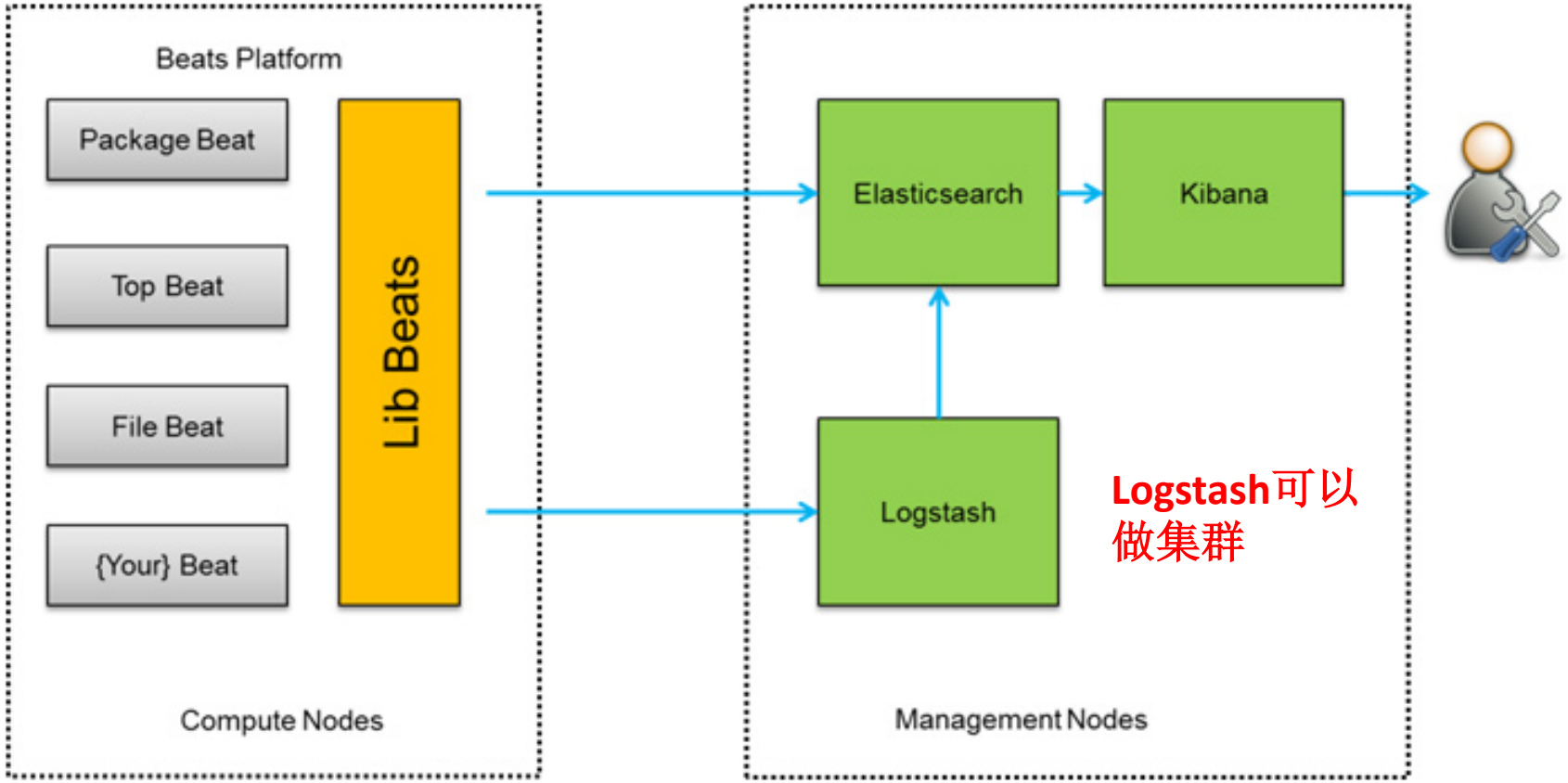
- 收集－能够采集多种来源的日志数据
- 传输－能够稳定的把日志数据传输到中央系统
- 转换－ 能够对收集的日志数据进行转换处理
- 存储－如何存储日志数据
- 分析－可以支持 UI 分析
- 告警－能够提供错误报告，监控机制

ELK提供了一整套解决方案，并且都是开源软件，之间互相配合使用，完美衔接，高效的满足了很多场合的应用。目前主流的一种日志系统。

- ELK架构（一）　老的架构


Logstash
Logstash
Logstash
Compute Nodes
Logstash
elasticsearch.
Visualize
kibana
Management Nodes

- ELK架构（二）　用beats来进行采集的架构



你是否明白了各部件的分工？？

目录

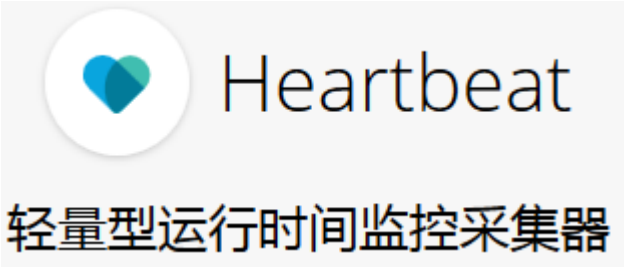
1 学习目标
2 ELK简介
3 Beats
4 Logstash

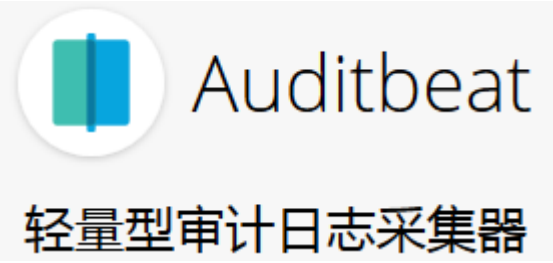
- **Beats是什么?**

  轻量型数据采集器。负责从目标源上采集数据。

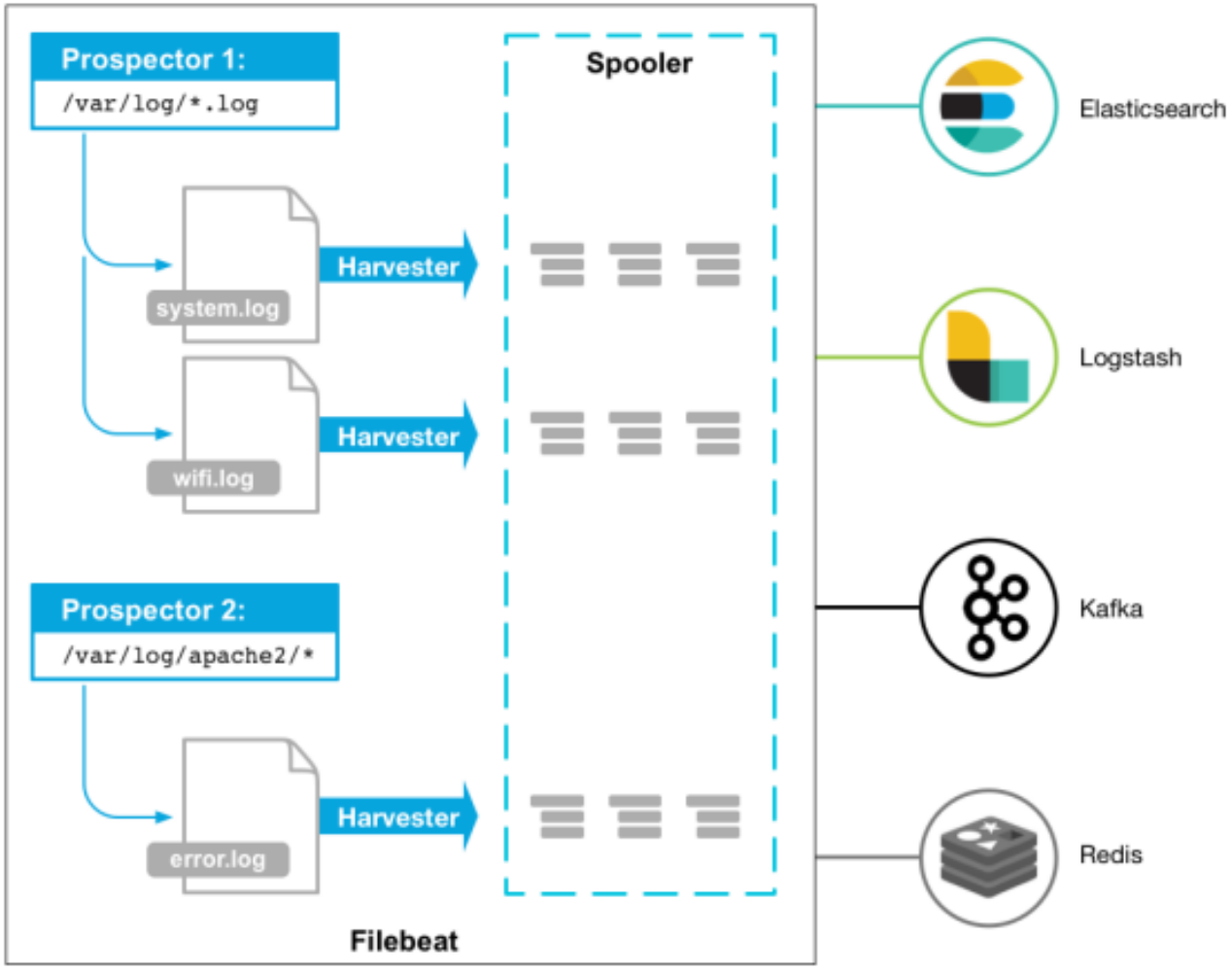
  官网介绍：https://www.elastic.co/cn/products/beats

- **Beats 的已有种类**





FileBeat 日志文件采集器 工作原理



Prospector 勘测者

负责管理Harvester并找到所有读取源。6.3开始叫 input 了。

Harvestor 收割机

负责读取单个文件内容，发送到输出

获取 FileBeat

6.2.4 版本下载地址：
https://www.elastic.co/downloads/past-releases/filebeat-6-2-4

最新版下载地址：
https://www.elastic.co/cn/downloads/beats/filebeat

- **使用步骤** https://www.elastic.co/guide/en/beats/filebeat/6.2/filebeat-getting-started.html

1、安装

windows：解压到安装目录即可

linux： rpm -ivh filebeat-6.2.4-x86_64.rpm

安装后的目录结构：

https://www.elastic.co/guide/en/beats/filebeat/6.2/directory-layout.html

使用步骤

2、配置：在 filebeat.yml 中配置从哪些文件读取数据，送到哪里去

```
filebeat.prospectors:
- type: log
  enabled: true
  paths:
    - /var/log/*.log
    #- c:\programdata\elasticsearch\logs\*
```

1、配置勘测采集源

```
output.elasticsearch:
  hosts: ["192.168.1.42:9200"]
```

2、配置输出送到es中去

若ES有认证，配置用户密码

```
output.elasticsearch:
  hosts: ["myEShost:9200"]
  username: "elastic"
  password: "elastic"
```

使用步骤

3、启动filebeat

linux rpm： sudo service filebeat start

windows：

安装了服务： PS C:\Program Files\Filebeat> Start-Service filebeat

如果没有安装服务，在安装目录直接运行启动程序 filebeat

sudo ./filebeat

可加启动选项：-e 输入日志到标准输出， -c 指定配置文件

如： sudo ./filebeat -e -c filebeat.yml

GET /_cat/indices?v 查看创建的索引

GET /filebeat*/_search?q=* 查看索引的数据格式

使用步骤

https://www.elastic.co/guide/en/beats/filebeat/6.2/filebeat-template.html

4、配置索引模板

默认情况下，如果输出是elasticsearch，filebeat自动创建推荐的索引模板（定义在fields.yml中）。

●如果你想使用自定义的模板，可在 filebeat.yml中配置指定你的模板

```
setup.template.name: "your_template_name"
setup.template.fields: "path/to/fields.yml"
```

●覆盖已存在的模板

```
setup.template.overwrite: true
```

●改变索引的名字。默认为filebeat-6.2.4-yyyy.MM.dd

```
output.elasticsearch.index: "customname-%{[beat.version]}-%{+yyyy.MM.dd}"
setup.template.name: "customname"
setup.template.pattern: "customname-*"
#setup.dashboards.index: "customname-*"
```

名字中应包含版本和日期部分

使用kibana的dashboard时需要

重启filebeat后才会创建

使用步骤

https://www.elastic.co/guide/en/beats/filebeat/6.2/filebeat-template.html

4、配置索引模板

●手动载入模板。当输出是logstash时，需手动执行命令来向es创建模板

```
filebeat setup --template -E output.logstash.enabled=false -E
'output.elasticsearch.hosts=["localhost:9200"]'
```

使用步骤

5、配置使用kibana的dashboards。在 filebeat.yml 中：

```
setup.dashboards.enabled: true

setup.kibana:
  host: "mykibanahost:5601"
```

有认证的配置

```
setup.kibana:
  host: "mykibanahost:5601"
  username: "elastic"
  password: "elastic"
```

重启filebeat，在kibana中浏览 Discover Visualize DashBoard

配置输出到logstash

```
output.logstash:
  hosts: ["127.0.0.1:5044"]
```

请记得，需要手动载入索引模板。

Filebeat的各种配置的详细说明请参考:

https://www.elastic.co/guide/en/beats/filebeat/6.2/configuring-howto-filebeat.html

- **FileBeat 的运行命令说明**

https://www.elastic.co/guide/en/beats/filebeat/6.2/command-line-options.html

FileBeat modules 模块

思考：

1. 日志信息只是作为一个文本字段放入ES中，还是应该将其解析为多个特定意义的字段，方便统计分析？
2. 各种应用（如 nginx tomcat mysql redis ）输出的日志格式一样吗？包含的信息域一样吗？

FileBeat modules 模块

fileBeat中提供了很多常见应用日志格式的读取解析模块，来简化我们的使用。
官网参考：

https://www.elastic.co/guide/en/beats/filebeat/6.2/filebeat-modules-quickstart.html

https://www.elastic.co/guide/en/beats/filebeat/6.2/filebeat-modules-overview.html

https://www.elastic.co/guide/en/beats/filebeat/6.2/configuration-filebeat-modules.html

```
sudo bin/elasticsearch-plugin install ingest-geoip
sudo bin/elasticsearch-plugin install ingest-user-agent
```

安装对应插件

```
filebeat modules list
filebeat modules enable apache2 auditd mysql
```

查看、启用模块

目录

1 学习目标

2 ELK简介

3 Beats

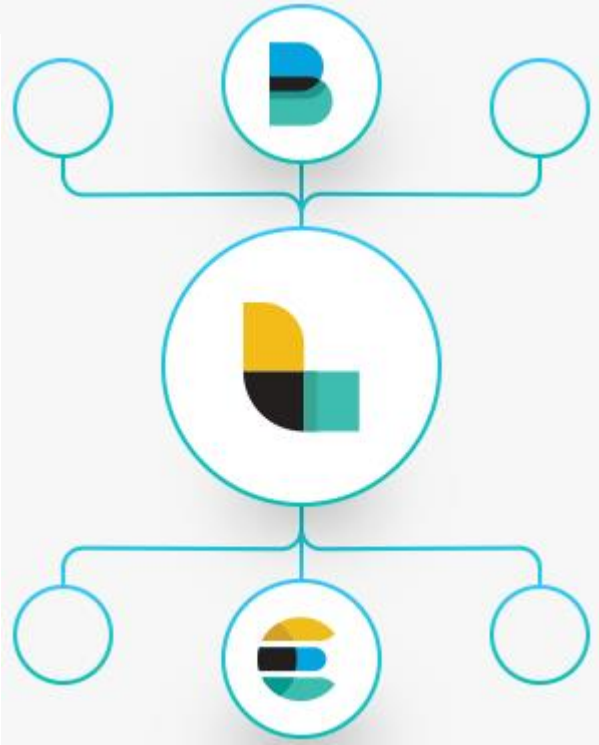
4 Logstash

- **Logstash的角色**

https://www.elastic.co/cn/products/logstash

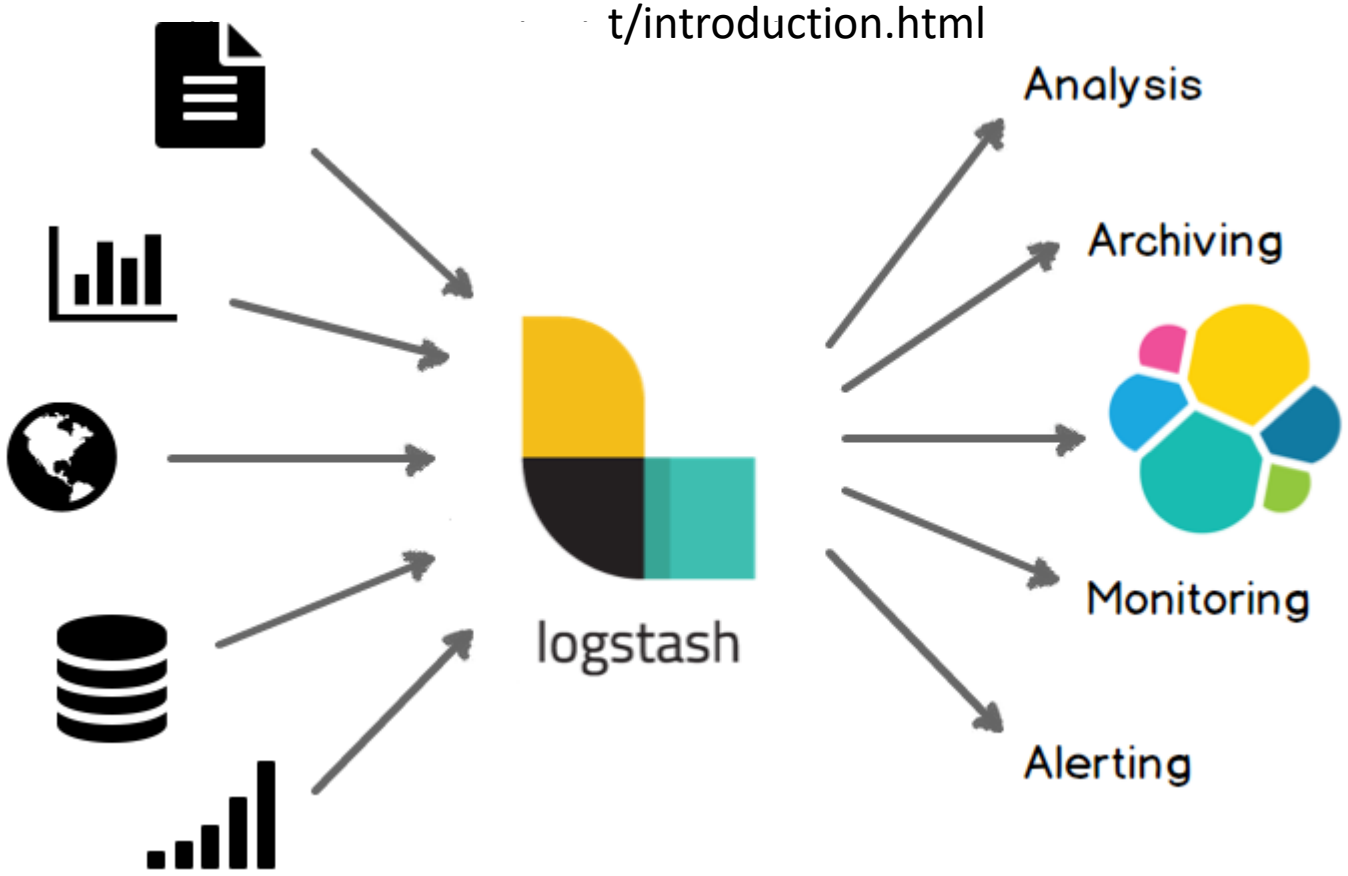
集中、转换和存储数据

Logstash 是开源的服务器端数据处理管道，能够同时 从多个来源采集数据、转换数据，然后将数据发送到您最喜欢的 “存储库” 中。（我们的存储库当然是 Elasticsearch。）



- **Logstash 介绍**

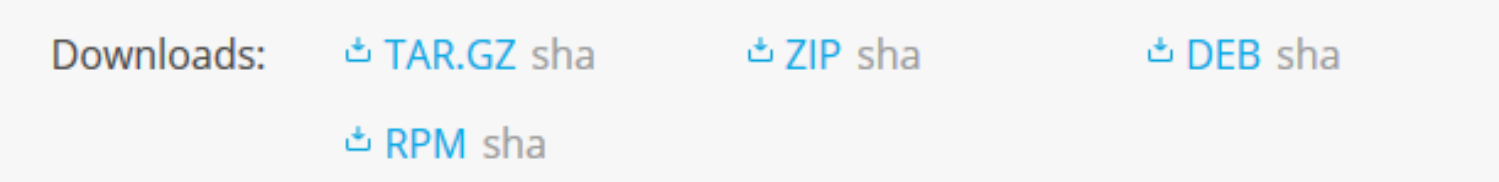
https://www.elastic.co/guide/en/logstash/current/introduction.html



获取 Logstash

https://www.elastic.co/cn/downloads/logstash

6.2.4 版本：https://www.elastic.co/downloads/past-releases/logstash-6-2-4

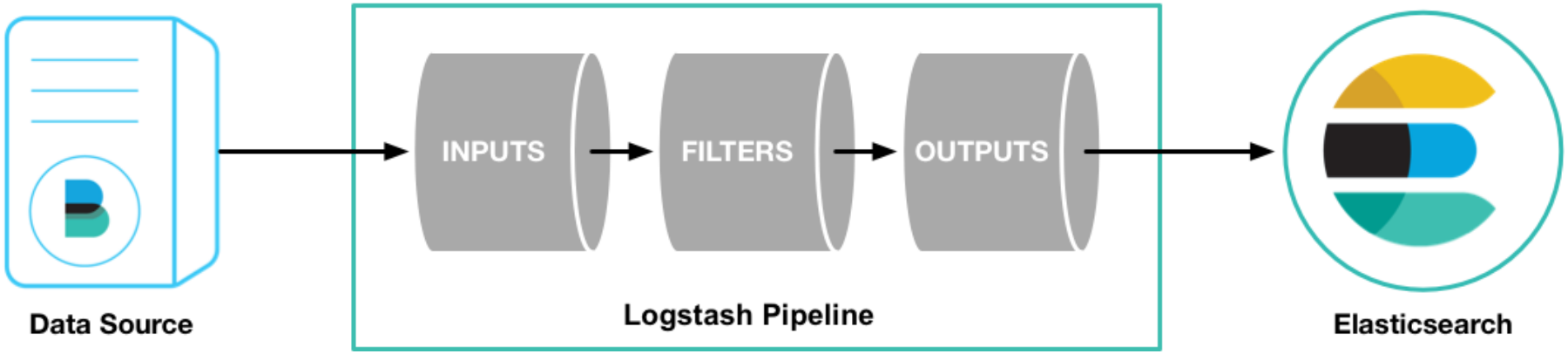


安装

开箱即用：
压缩包：解压到安装目录
rpm包：rpm -ivh logstash-6.2.4.rpm

logstash Pipeline 管道　工作原理

启动logstash实例时需为其指定管道定义。



https://www.elastic.co/guide/en/logstash/6.2/pipeline.html
https://www.elastic.co/guide/en/logstash/6.2/execution-model.html

试用

用最简单的管道定义来启动logstash实例

```
cd logstash-6.2.4
bin/logstash -e 'input { stdin { } } output { stdout {} }'
```

```
[2018-06-21T17:47:08,750][INFO ][logstash.agent
count=>1, :pipelines=>["main"]}
aaaaaa
{
       "message" => "aaaaaa",
      "@version" => "1",
    "@timestamp" => 2018-06-21T09:47:51.015Z,
          "host" => "localhost.localdomain"
}
```

用logstash来解析日志

1、配置 Filebeat 将日志发送给logstash

```
filebeat.prospectors:
- type: log
  paths:
    - /path/to/file/logstash-tutorial.log
output.logstash:
  hosts: ["localhost:5044"]
```

logstash-tutorial.log

用logstash来解析日志

2、配置Logstash的管道输入为 Filebeat

管道定义模板

```
# The # character at the beginning of a line indicates a comment. Use
# comments to describe your configuration.
input {
}
# The filter part of this file is commented out to indicate that it is
# optional.
# filter {
#
# }
output {
}
```

用logstash来解析日志

2、配置Logstash的管道输入为 Filebeat

在logstash home目中创建文件管道配置文件 first-pipeline.conf，配置如下：

```
input {
   beats {                          定义 beats 输入
      port => "5044"
   }
}
# The filter part of this file is commented out to indicate that it is
# optional.
# filter {
#
# }
output {
   stdout { codec => rubydebug }    定义 输出到标准输出
}
```

用logstash来解析日志

2、配置Logstash的管道输入为 Filebeat

测试管道配置是否正确

```
bin/logstash -f first-pipeline.conf --config.test_and_exit
```

启动logstash实例

```
bin/logstash -f first-pipeline.conf --config.reload.automatic
```

用logstash来解析日志

3、配置filter 来解析日志

```
input {
    beats {
        port => "5044"
    }
}
filter {
    grok {
        match => { "message" => "%{COMBINEDAPACHELOG}"}
    }
}
output {
    stdout { codec => rubydebug }
}
```

配置使用 grok 过进行滤转换

关闭filebeat实例，执行 sudo rm data/registry 删除filebeat记录，再启动filebeat

用logstash来解析日志

4、再加入一个过滤器

```
geoip {
      source => "clientip"
   }
```

实现ip转地理坐标

关闭filebeat实例，执行 sudo rm data/registry 删除filebeat记录，再启动filebeat

用logstash来解析日志

5、配置输出到elasticsearch

```
output {
   elasticsearch {
      hosts => [ "localhost:9200" ]
   }
}
```

关闭filebeat实例，执行 sudo rm data/registry 删除filebeat记录，再启动filebeat