**Logstash**

#### Logstash服务搭建

1. 下载安装包

https://artifacts.elastic.co/downloads/logstash/logstash-6.3.2.tar.gz

1. 复制到linux运行命令进行解压

tar -zvxf logstash-6.3.2.tar.gz

1. 进入根目录运行如下命令启动服务

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

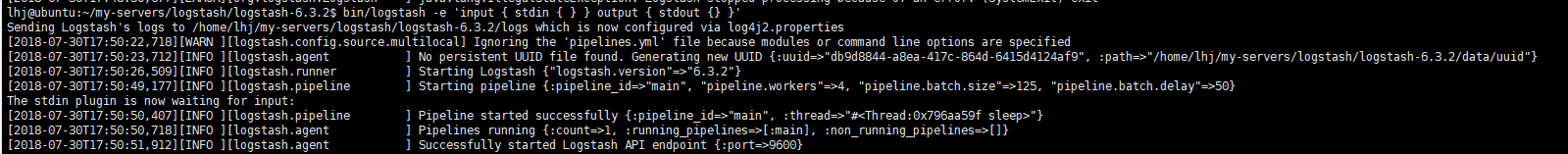
bin/logstash -e 'input { stdin { } }' --path.data=data/data

1. 然后报错了

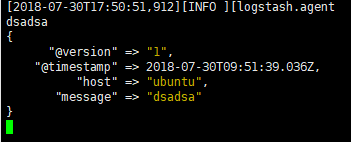
解决方法：

sudo rm /etc/ssl/certs/java/cacerts

再次运行启动命令



1. 输入数据，得到结果—>启动成功



1. 进入config目录创建配置文件logstash.conf

input {

stdin { }

}

output {

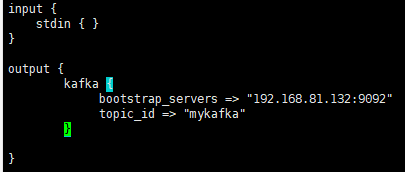
kafka {

bootstrap\_servers => "127.0.0.1:9092"

topic\_id => "logs"

}

}



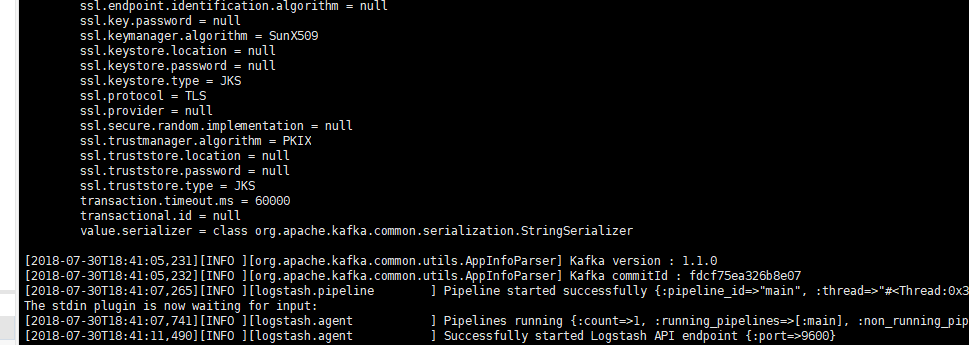
1. 输入bin/logstash -f config/logstash.conf --config.test\_and\_exit查看运行结果



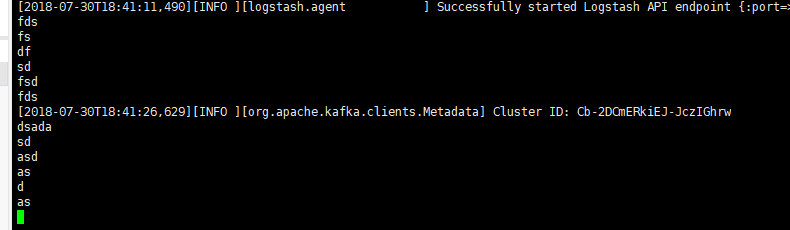
显示ok表示配置成功

1. 然后运行如下命令启动服务

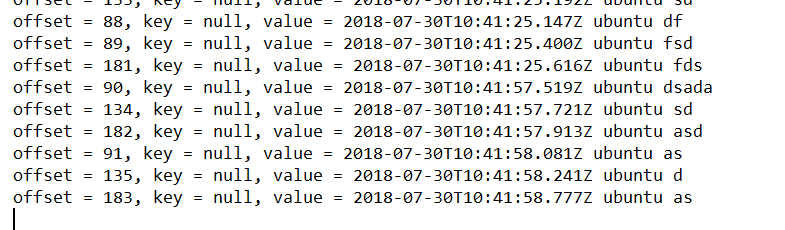
bin/logstash -f config/logstash.conf --config.reload.automatic －－这个配置可以使修改配置文件的时候自动读取配置而无需重启logstash



1. 输入任意数据



消费端可以接收到信息



1. 从tomcat的日志文件中拉取数据

input {

file {

type => "kafka"

path => "/home/lhj/my-servers/tomcat/apache-tomcat-8.5.32/logs/catalina.\*.log"

start\_position => "beginning"

}

}

output {

if [type] == "kafka" {

kafka {

bootstrap\_servers => "127.0.0.1:9092"

topic\_id => "mykafka"

}

}

}

1. 后台启动logstash

nohup bin/logstash -f config/logstash.conf --config.reload.automatic &

1. 输出到elasticsearch的配置

input {

file {

type => "elk-test"

path => "/home/lhj/my-servers/tomcat/apache-tomcat-8.5.32/logs/catalina.\*.log"

start\_position => "beginning"

}

}

output {

elasticsearch {

hosts => ["192.168.81.131:9200"]

index => "elk-test-%{+YYYY.MM.dd}"

template\_name => "elk-test"

}

}



1. 日至过滤

input {

file {

type => "elk-test"

path => "/home/lhj/my-servers/tomcat/apache-tomcat-8.5.32/logs/\*\_log\*"

start\_position => "beginning"

codec => multiline {

pattern => "^%{TIMESTAMP\_ISO8601} "

negate => true

what => "previous"

}

}

}

filter {

if [message] =~ "^at"{

drop{}

}

else if [message] =~ "com.evp.charge.remote\S\*"{

grok {

match => {

"message" => "(?<date>(\d{4}-\d{2}-\d{2}\s\d{2}:\d{2}:\d{2}))\s\*%{LOGLEVEL:log\_level}[\s\S]\*(?<service>(com.evp.charge.remote.\S\*Impl))\S%{NUMBER:line}(?<info>[\s\S]\*)"

}

}

}

else if [message] =~ "com.evp.charge.service\S\*"{

grok{

match => {

"message" => "(?<date>(\d{4}-\d{2}-\d{2}\s\d{2}:\d{2}:\d{2}))\s\*%{LOGLEVEL:log\_level}[\s\S]\*(?<service>(com.evp.charge.remote.\S\*Impl))\S%{NUMBER:line}(?<info>[\s\S]\*)"

}

}

}

else{

drop{}

}

}

output {

stdout{}

elasticsearch {

hosts => ["192.168.81.131:9200"]

index => "elk-test-%{+YYYY.MM.dd}"

template\_name => "elk-test"

}

}

1. 多行日至合并成一行

input {

file {

type => "kafka"

path => "/usr/local/tomcat/logs/\*\_log\*"

start\_position => "beginning"

codec => multiline {

pattern => "^%{TIMESTAMP\_ISO8601} "

negate => true

what => "previous"

}

}

}

其中multiline插件可以实现多行日志合并成一行