elasticsearch

启动

修改 elasticsearch-8.13.3/config/elasticsearch.yml

```
#network.host: 192.168.0.1
network.host: 127.0.0.1
```

报错解决：][o.e.h.n.Netty4HttpServerTransport] [DESKTOP-85NUTA2] received plaintext http traffic on an https channel, closing connection Netty4HttpChannel{localAddress=/127.0.0.1:9200

```
# Enable security features
#xpack.security.enabled: true
xpack.security.enabled: false
```

windows

执行 elasticsearch-8.13.3/bin/elasticsearch.bat

localhost:9200

YouTube 地图 doc peppacatt blog 翻译 ChatGP


```
{
    "name": "DESKTOP-85NUTA2",
    "cluster_name": "elasticsearch",
    "cluster_uuid": "CgLERaS6TzGnYQAcqsuP6g",
    "version": {
        "number": "8.13.3",
        "build_flavor": "default",
        "build_type": "zip",
        "build_hash": "617f7b76c4ebcb5a7f1e70d409a99c437c896aea",
        "build_date": "2024-04-29T22:05:16.051731935Z",
        "build_snapshot": false,
        "lucene_version": "9.10.0",
        "minimum_wire_compatibility_version": "7.17.0",
        "minimum_index_compatibility_version": "7.0.0"
    },
    "tagline": "You Know, for Search"
}
```

logstash

资料

https://doc.yonyoucloud.com/doc/logstash-best-practice-cn/output/file.html
https://www.elastic.co/guide/en/logstash/current/plugins-outputs-elasticsearch.html

filebeat 和 logstash

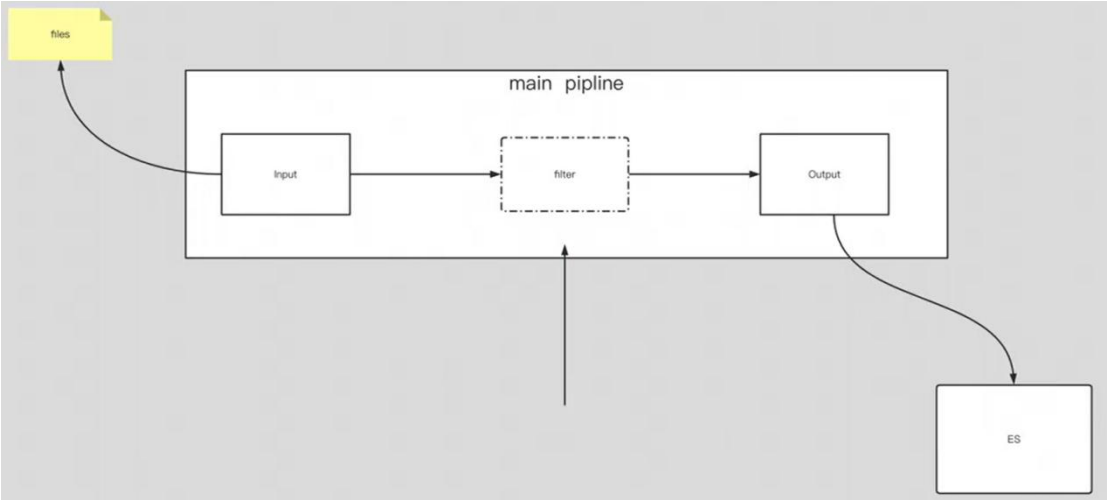
因为 logstash 是 jvm 跑的，资源消耗比较大，所以后来作者又用 golang 写了一个功能较少但是资源消耗也小的轻量级的 logstash-forwarder。不过作者只是一个人，加入 http://elastic.co 公司以后，因为 es 公司本身还收购了另一个开源项目 packetbeat，而这个项目专门就是用 golang 的，有整个团队，所以 es 公司干脆把 logstash-forwarder 的开发工作也合并到同一个 golang 团队来搞，于是新的项目就叫 filebeat 了。

logstash 和 filebeat 都具有日志收集功能，filebeat 更轻量，占用资源更少，但 logstash 具有 filter 功能，能过滤分析日志。一般结构都是 filebeat 采集日志，然后发送到消息队列，redis，kafaka。然后 logstash 去获取，利用 filter 功能过滤分析，然后存储到 elasticsearch 中

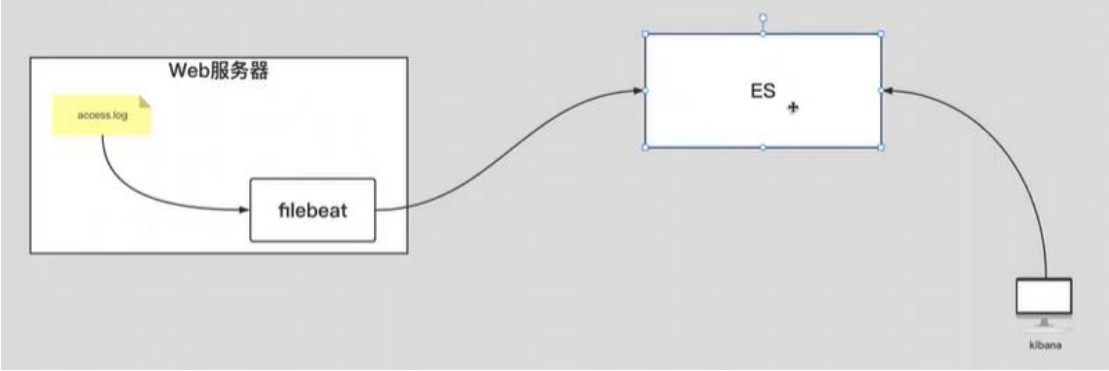
logstash 的安装包要比 filebeat 大的多

```
-rw-r--r-- 1 root root 347M Apr 20 21:18 logstash-7.17.3-linux-x86_64.tar.gz
-rw-r--r-- 1 root root  35M Apr 20 21:00 filebeat-7.17.3-linux-x86_64.tar.gz
```

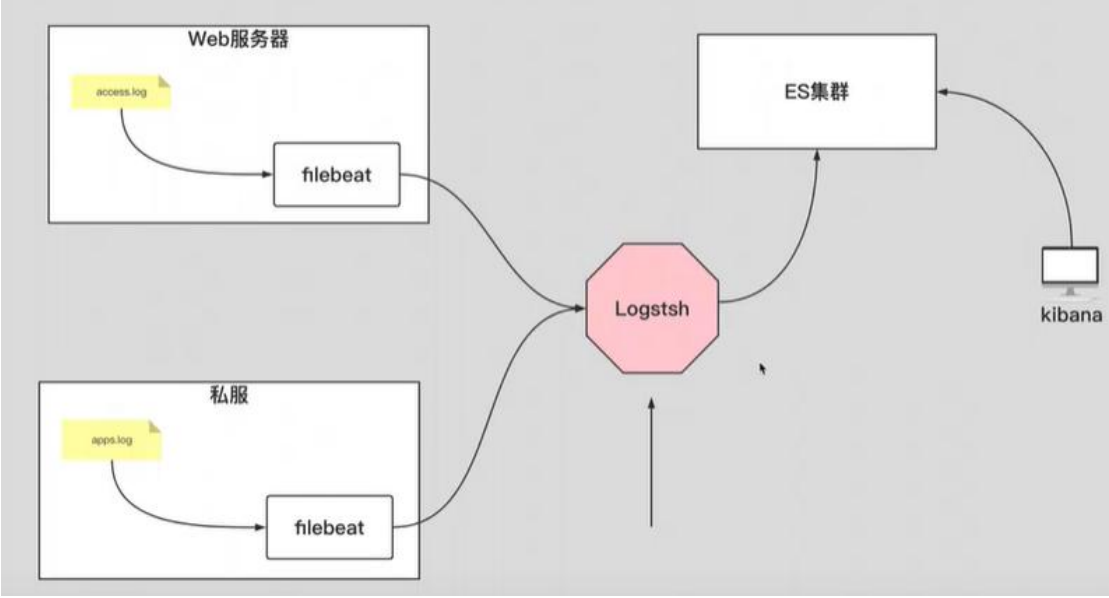
logstash 和 filebeat 都有输入和输出，但是 logstash 比 filebeat 多了个 filter 的功能,该功能就是 logstash 的强大之处



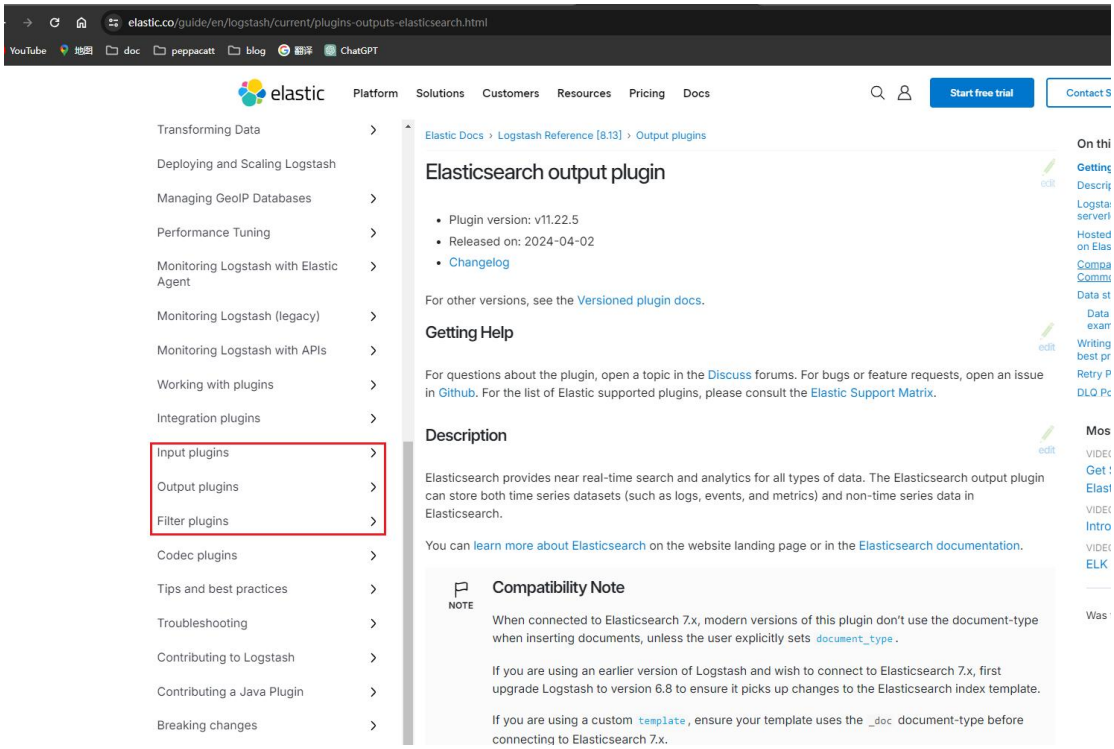
简单的数据处理



需要对数据进度分析处理



logstash 部署



安装

```
yum -y localinstall logstash-7.17.3-x86_64.rpm

ln -sv /usr/share/logstash/bin/logstash /usr/local/bin/


下载地址：
    https://www.elastic.co/downloads/past-releases#logstash
```

启动

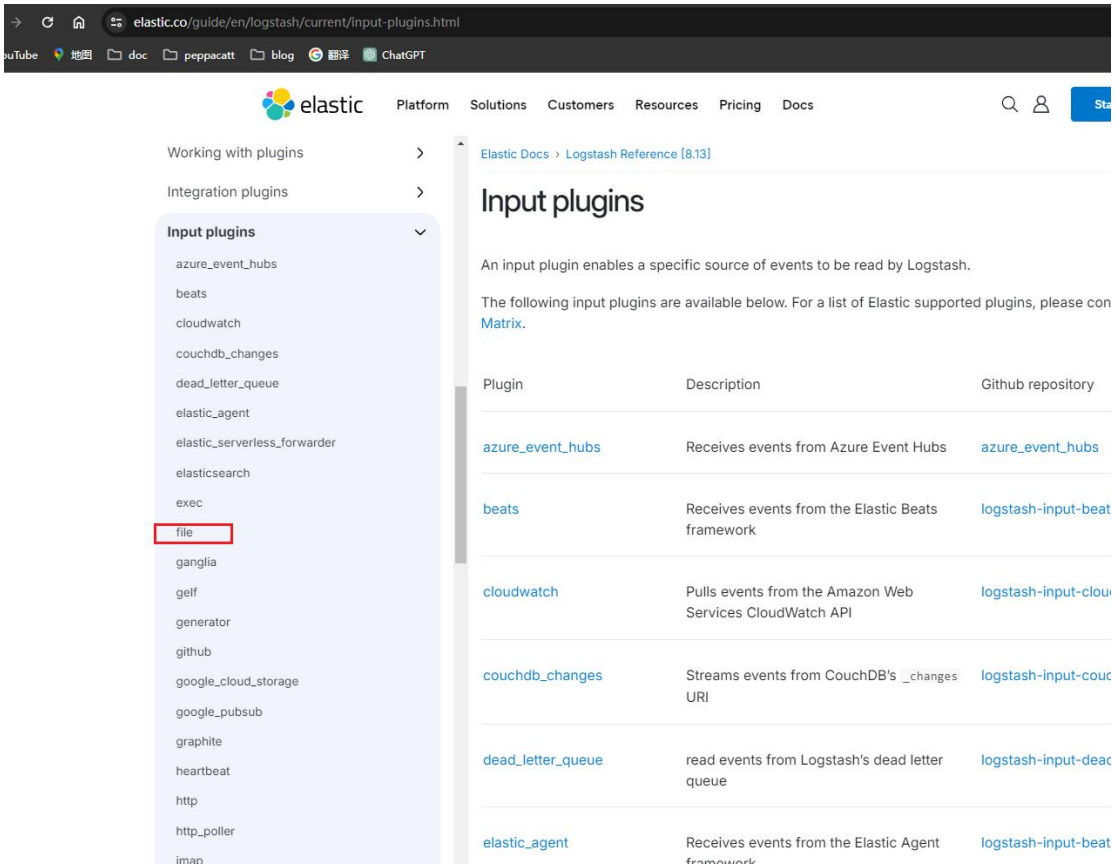
注意!!!!!!!!!!!!
myConf.conf 文件最好不要使用 word 文档编辑，最好使用 vim 编辑，否则可能会因为格式问题启动失败

rm -rf /root/logstash/input/*
rm -rf /root/logstash/output/*
rm -rf /root/logstash/logstash-8.13.3/data/plugins/inputs/file/.*
vim /root/logstash/logstash-8.13.3/config/myConf.conf

./logstash-8.13.3/bin/logstash -f /root/logstash/logstash-8.13.3/config/myConf.conf
./logstash-8.13.3/bin/logstash -e 'input{stdin{}}output{stdout{}}'

input 插件

input->file



input 中的插件 file

```
total 1319984
drwxr-xr-x 2 root root      4096 May 12 12:22 config-logstash
drwxr-xr-x 2 root root      4096 May 10 20:27 day01-实战软件及配置文件
-rw-r--r-- 1 root root 311777007 Apr 20 21:08 elasticsearch-7.17.3-linux-x86_64.tar.gz
-rw-r--r-- 1 root root 311873551 Apr 20 21:09 elasticsearch-7.17.3-x86_64.rpm
-rw-r--r-- 1 root root 363469045 Apr 20 21:18 logstash-7.17.3-linux-x86_64.tar.gz
-rw-r--r-- 1 root root 364517723 Apr 20 21:20 logstash-7.17.3-x86_64.rpm
[root@elk101.oldboyedu.com ~]#
[root@elk101.oldboyedu.com ~]#
[root@elk101.oldboyedu.com ~]# cp config-logstash/01-stdin-to-stdout.conf config-logstash/02-file-to-stdout.conf
[root@elk101.oldboyedu.com ~]#
[root@elk101.oldboyedu.com ~]# vim config-logstash/02-file-to-stdout.conf
[root@elk101.oldboyedu.com ~]#
[root@elk101.oldboyedu.com ~]#
[root@elk101.oldboyedu.com ~]# logstash -f config-logstash/02-file-to-stdout.conf
Using bundled JDK: /usr/share/logstash/jdk
OpenJDK 64-Bit Server VM warning: Option UseConcMarkSweepGC was deprecated in version 9.0 and will likely be removed in a future release.
WARNING: Could not find logstash.yml which is typically located in $LS_HOME/config or /etc/logstash. You can specify the path using --path.settings. Conti
aults
Could not find log4j2 configuration at path /usr/share/logstash/config/log4j2.properties. Using default config which logs errors to the console
[INFO ] 2022-05-12 14:36:22.050 [main] runner - Starting Logstash {"logstash.version"=>"7.17.3", "jruby.version"=>"jruby 9.2.20.1 (2.5.8) 2021-11-30 2a296
t Server VM 11.0.14.1+1 on 11.0.14.1+1 +indy +jit [linux-x86_64]"}
[INFO ] 2022-05-12 14:36:22.136 [main] runner - JVM bootstrap flags: [-Xms1g, -Xmx1g, -XX:+UseConcMarkSweepGC, -XX:CMSInitiatingOccupancyFraction=75, -XX:
cupancyOnly, -Djava.awt.headless=true, -Dfile.encoding=UTF-8, -Djruby.compile.invokedynamic=true, -Djruby.jit.threshold=0, -Djruby.regexp.interruptible=tr
OutOfMemoryError, -Djava.security.egd=file:/dev/urandom, -Dlog4j2.isThreadContextMapInheritable=true]
[WARN ] 2022-05-12 14:36:22.591 [LogStash::Runner] multilocal - Ignoring the 'pipelines.yml' file because modules or command line options are specified
[INFO ] 2022-05-12 14:36:23.642 [Api Webserver] agent - Successfully started Logstash API endpoint {:port=>9600, :ssl_enabled=>false}
[INFO ] 2022-05-12 14:36:24.397 [Converge PipelineAction::Create<main>] Reflections - Reflections took 56 ms to scan 1 urls, producing 119 keys and 419 va
[WARN ] 2022-05-12 14:36:24.988 [Converge PipelineAction::Create<main>] plain - Relying on default value of `pipeline.ecs_compatibility`, which may change
release of Logstash. To avoid unexpected changes when upgrading Logstash, please explicitly declare your desired ECS Compatibility mode.
[WARN ] 2022-05-12 14:36:25.022 [Converge PipelineAction::Create<main>] file - Relying on default value of `pipeline.ecs_compatibility`, which may change
elease of Logstash. To avoid unexpected changes when upgrading Logstash, please explicitly declare your desired ECS Compatibility mode.
[INFO ] 2022-05-12 14:36:25.244 [[main]-pipeline-manager] javapipeline - Starting pipeline {:pipeline_id=>"main", "pipeline.workers"=>2, "pipeline.batch.s
e.batch.delay"=>50, "pipeline.max_inflight"=>250, "pipeline.sources"=>["/root/config-logstash/02-file-to-stdout.conf"], :thread=>"#<Thread:0x78c44e51 run>
[INFO ] 2022-05-12 14:36:25.870 [[main]-pipeline-manager] javapipeline - Pipeline Java execution initialization time {"seconds"=>0.62}
[INFO ] 2022-05-12 14:36:25.961 [[main]-pipeline-manager] file - No sincedb_path set, generating one based on the "path" setting {:sincedb_path=>"/usr/sha
ugins/inputs/file/.sincedb_3cd99a80ca58225ec14dc0ac340abb80", :path=>["/tmp/test/*.txt"]}
[INFO ] 2022-05-12 14:36:25.980 [[main]-pipeline-manager] javapipeline - Pipeline started {"pipeline.id"=>"main"}
```

```
(1)编写logstash的配置文件
cat > conf.d/02-file-to-stdout <<'EOF'
input {
  file {
      # 指定收集的路径
      path => ["/tmp/test/*.txt"]
      # 指定文件的读取位置，仅在第一次生效.
      start_position => "beginning"
  }
}


output {
  stdout {}

 # elasticsearch {
 # }
}
EOF


(2)启动logstash实例
logstash -rf conf.d/02-file-to-stdout
```

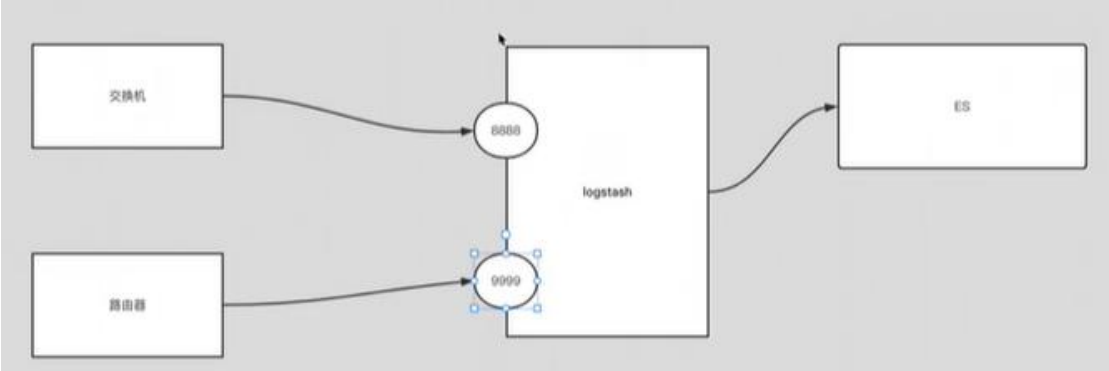
已读取过的文件会在 sincedb 中产生记录,start_position 为 beginning 时，读取的是在 sincedb 中没有记录的文件

```
[INFO ] 2022-05-12 14:48:39.368 [[main]-pipeline-manager] javapipeline - Starting pipeline {:pipeline_id=>"main", "pipeline.workers"=>2, "pipeline.batch.size"=>
e.batch.delay"=>50, "pipeline.max_inflight"=>250, "pipeline.sources"=>["/root/config-logstash/02-file-to-stdout.conf"], :thread=>"#<Thread:0x29e641d9 run>"}
[INFO ] 2022-05-12 14:48:39.886 [[main]-pipeline-manager] javapipeline - Pipeline Java execution initialization time {"seconds"=>0.52}
[INFO ] 2022-05-12 14:48:39.932 [[main]-pipeline-manager] file - No sincedb_path set, generating one based on the "path" setting {:sincedb_path=>"/usr/share/log
ugins/inputs/file/.sincedb_3d99a80ca58225ec14dc0ac340abb80", :path=>["/tmp/test/*.txt"]}
[INFO ] 2022-05-12 14:48:39.957 [[main]-pipeline-manager] javapipeline - Pipeline started {"pipeline.id"=>"main"}
[INFO ] 2022-05-12 14:48:39.998 [Agent thread] agent - Pipelines running {:count=>1, :running_pipelines=>[:main], :non_running_pipelines=>[]}
[INFO ] 2022-05-12 14:48:40.017 [[main]<file] observingtail - START, creating Discoverer, Watch with file and sincedb collections
[WARN ] 2022-05-12 14:48:40.196 [[main]<file] plain - Relying on default value of `pipeline.ecs_compatibility`, which may change in a future major release of Lo
oid unexpected changes when upgrading Logstash, please explicitly declare your desired ECS Compatibility mode.
[WARN ] 2022-05-12 14:48:40.249 [[main]<file] plain - Relying on default value of `pipeline.ecs_compatibility`, which may change in a future major release of Lo
oid unexpected changes when upgrading Logstash, please explicitly declare your desired ECS Compatibility mode.
{
          "path" => "/tmp/test/3.txt",
      "@version" => "1",
    "@timestamp" => 2022-05-12T06:48:40.214Z,
          "host" => "elk101.oldboyedu.com",
       "message" => "bbb"
}
{
          "path" => "/tmp/test/3.txt",
      "@version" => "1",
    "@timestamp" => 2022-05-12T06:48:40.227Z,
          "host" => "elk101.oldboyedu.com",
       "message" => "ccc"
}
{
          "path" => "/tmp/test/4.txt",
      "@version" => "1",
    "@timestamp" => 2022-05-12T06:48:40.250Z,
          "host" => "elk101.oldboyedu.com",
       "message" => "DDDDDD"
}
{
          "path" => "/tmp/test/4.txt",
      "@version" => "1",
    "@timestamp" => 2022-05-12T06:48:40.251Z,
```

input->tcp

tcp 使用场景

对于不支持安装客户端的交换机和路由器，可以先通过 tcp 将日志发到指定端口，由 logstash 聚合




```
[root@elk101.oldboyedu.com ~]# cp config-logstash/02-file-to-stdout.conf config-logstash/03-tcp-to-stdout.conf
[root@elk101.oldboyedu.com ~]#
[root@elk101.oldboyedu.com ~]# vim config-logstash/03-tcp-to-stdout.conf
[root@elk101.oldboyedu.com ~]#
[root@elk101.oldboyedu.com ~]# logstash -f config-logstash/03-tcp-to-stdout.conf
Using bundled JDK: /usr/share/logstash/jdk
OpenJDK 64-Bit Server VM warning: Option UseConcMarkSweepGC was deprecated in version 9.0 and will likely be removed in a future release.
```

```
input {
  tcp {
     port => 8888
  }


  tcp {
     port => 9999
  }

}


output {
  stdout {}
}
```

监听 8888 和 9999 端口

```
[root@elk101.oldboyedu.com test]# ss -ntl
State       Recv-Q Send-Q                                   Local Address:Port
LISTEN      0      128                                                 *:22
LISTEN      0      100                                         127.0.0.1:25
LISTEN      0      128                                                 *:6379
LISTEN      0      128                                              [::]:22
LISTEN      0      128                                              [::]:8888
LISTEN      0      100                                             [::1]:25
LISTEN      0      50                                  [::ffff:127.0.0.1]:9600
LISTEN      0      128                                              [::]:9200
LISTEN      0      128                                              [::]:9300
[root@elk101.oldboyedu.com test]#
[root@elk101.oldboyedu.com test]#
```

发送数据测试

使用 telnet 命令向 10.0.0.101 的 8888 端口发送数据 aaaaaaaaaaaaa

```
[root@elk103.oldboyedu.com ~]#
[root@elk103.oldboyedu.com ~]# telnet 10.0.0.101 8888
Trying 10.0.0.101...
Connected to 10.0.0.101.
Escape character is '^]'.
aaaaaaaaaaaaaaaaaaa
```

结果

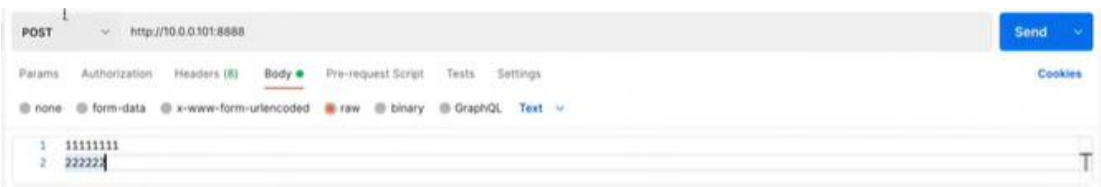
```
[root@elk101.oldboyedu.com ~]# cp config-logstash/02-file-to-stdout.conf config-logstash/03-tcp-to-stdout.conf
[root@elk101.oldboyedu.com ~]#
[root@elk101.oldboyedu.com ~]# vim config-logstash/03-tcp-to-stdout.conf
[root@elk101.oldboyedu.com ~]#
[root@elk101.oldboyedu.com ~]# logstash -f config-logstash/03-tcp-to-stdout.conf
Using bundled JDK: /usr/share/logstash/jdk
OpenJDK 64-Bit Server VM warning: Option UseConcMarkSweepGC was deprecated in version 9.0 and will likely be removed in a future release.
WARNING: Could not find logstash.yml which is typically located in $LS_HOME/config or /etc/logstash. You can specify the path using --path.settings. Cont
aults
Could not find log4j2 configuration at path /usr/share/logstash/config/log4j2.properties. Using default config which logs errors to the console
[INFO ] 2022-05-12 15:07:23.019 [main] runner - Starting Logstash {"logstash.version"=>"7.17.3", "jruby.version"=>"jruby 9.2.20.1 (2.5.8) 2021-11-30 2a29
t Server VM 11.0.14.1+1 on 11.0.14.1+1 +indy +jit [linux-x86_64]"}
[INFO ] 2022-05-12 15:07:23.061 [main] runner - JVM bootstrap flags: [-Xms1g, -Xmx1g, -XX:+UseConcMarkSweepGC, -XX:CMSInitiatingOccupancyFraction=75, -XX
cupancyOnly, -Djava.awt.headless=true, -Dfile.encoding=UTF-8, -Djruby.compile.invokedynamic=true, -Djruby.jit.threshold=0, -Djruby.regexp.interruptible=t
OutOfMemoryError, -Djava.security.egd=file:/dev/urandom, -Dlog4j2.isThreadContextMapInheritable=true]
[WARN ] 2022-05-12 15:07:23.779 [LogStash::Runner] multilocal - Ignoring the 'pipelines.yml' file because modules or command line options are specified
[INFO ] 2022-05-12 15:07:25.288 [Api Webserver] agent - Successfully started Logstash API endpoint {:port=>9600, :ssl_enabled=>false}
[INFO ] 2022-05-12 15:07:25.614 [Converge PipelineAction::Create<main>] Reflections - Reflections took 75 ms to scan 1 urls, producing 119 keys and 419 v
[WARN ] 2022-05-12 15:07:26.162 [Converge PipelineAction::Create<main>] line - Relying on default value of `pipeline.ecs_compatibility`, which may change
elease of Logstash. To avoid unexpected changes when upgrading Logstash, please explicitly declare your desired ECS Compatibility mode.
[WARN ] 2022-05-12 15:07:26.184 [Converge PipelineAction::Create<main>] tcp - Relying on default value of `pipeline.ecs_compatibility`, which may change
lease of Logstash. To avoid unexpected changes when upgrading Logstash, please explicitly declare your desired ECS Compatibility mode.
[INFO ] 2022-05-12 15:07:26.396 [[main]-pipeline-manager] javapipeline - Starting pipeline {:pipeline_id=>"main", "pipeline.workers"=>2, "pipeline.batch.
e.batch.delay"=>50, "pipeline.max_inflight"=>250, "pipeline.sources"=>["/root/config-logstash/03-tcp-to-stdout.conf"], :thread=>"#<Thread:0x38e3bcc4 run>
[INFO ] 2022-05-12 15:07:27.194 [[main]-pipeline-manager] javapipeline - Pipeline Java execution initialization time {"seconds"=>0.8}
[INFO ] 2022-05-12 15:07:27.423 [[main]-pipeline-manager] javapipeline - Pipeline started {"pipeline.id"=>"main"}
[INFO ] 2022-05-12 15:07:27.454 [Agent thread] agent - Pipelines running {:count=>1, :running_pipelines=>[:main], :non_running_pipelines=>[]}
[INFO ] 2022-05-12 15:07:27.498 [[main]<tcp] tcp - Starting tcp input listener {:address=>"0.0.0.0:8888", :ssl_enable=>false}
[WARN ] 2022-05-12 15:07:52.073 [nioEventLoopGroup-2-1] line - Relying on default value of `pipeline.ecs_compatibility`, which may change in a future maj
sh. To avoid unexpected changes when upgrading Logstash, please explicitly declare your desired ECS Compatibility mode.
{
          "port" => 44482,
      "@version" => "1",
          "host" => "elk103.oldboyedu.com",
       "message" => "aaaaaaaaaaaaaaaaaa\r",
    "@timestamp" => 2022-05-12T07:07:58.184Z
}
```

input->http

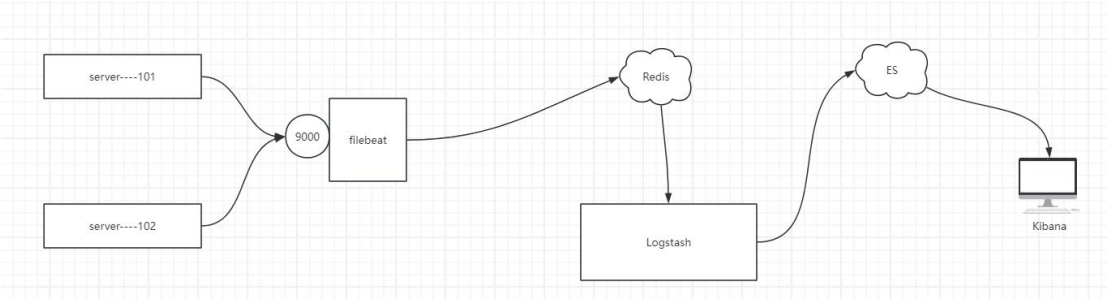
http 和 tcp 在 7 层协议中对应的层级不同

```
[root@elk101.oldboyedu.com ~]# cat config-logstash/04-http-to-stdout.conf
input {
  http {
     port => 8888
  }


  http {
     port => 9999
  }

}


output {
  stdout {}
}
[root@elk101.oldboyedu.com ~]#
[root@elk101.oldboyedu.com ~]#
[root@elk101.oldboyedu.com ~]# logstash -f config-logstash/04-http-to-stdout.conf
Using bundled JDK: /usr/share/logstash/jdk
OpenJDK 64-Bit Server VM warning: Option UseConcMarkSweepGC was deprecated in version 9.0 and will likely be removed in a future release.
```

发送数据测试(可使用 postman)



input->redis




```
input {
  redis {
     # 指定的是REDIS的键(key)的类型
     data_type => "list"
     # 指定数据库的编号,默认值是0号数据库
     db => 5
     # 指定数据库的ip地址,默认值是localhost
     host => "10.0.0.101"
     # 指定数据库的端口号, 默认值为6379
     port => 6379
     # 指定redis的认证密码
     password => "oldboyedu"
     # 指定从redis的哪个key取数据
     key => "oldboyedu-linux80-filebeat"

  }

}


output {
  stdout {}
}
```

input->beats

filebeat 输出到 logstash

```
    filbeat配置:
filebeat.inputs:
- type: tcp
  host: "0.0.0.0:9000"

output.logstash:
  hosts: ["10.0.0.101:5044"]



    logstsh配置:
input {
  beats {
     port => 5044
  }

}


output {
  stdout {}
}
```

filebeat 配置修改

```
filebeat.inputs:
- type: tcp
  host: "0.0.0.0:9000"


output.logstash:
  hosts: ["10.0.0.101:5044"]
```

logstash 配置修改

```
input {
  beats {
     port => 5044
  }

}


output {
  stdout {}
}
```

filter 插件

filter 插件添加/删除字段、tag

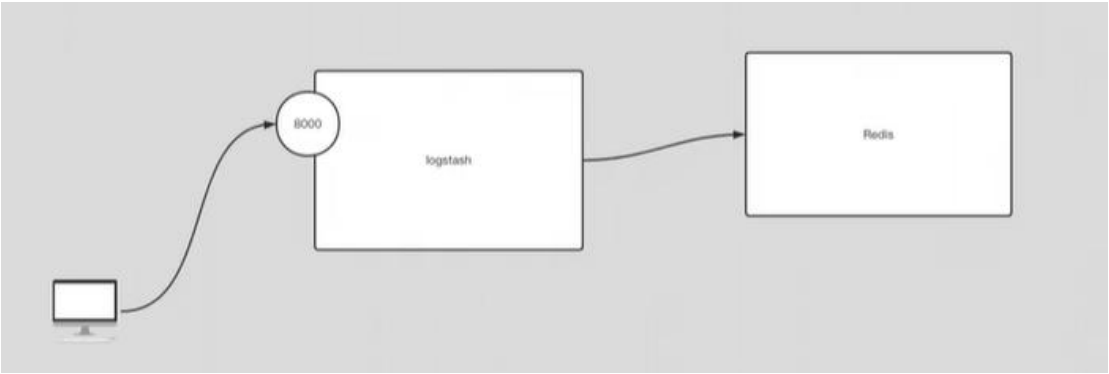
```
cat >> stdin-remove_add_field-stout.conf << EOF
input {
  beats {
    port => 5044
  }
}

filter {
  mutate {
    #移除指定的字段，使用逗号分隔
    remove_field => [
"tags","agent","input","log","ecs","version","@version","ident","referrer","auth" ]


    #添加指定的字段，使用逗号分隔
    #"%{clientip}"使用%可以将已有字段的值当作变量使用
    add_field => {
     "app_name" => "nginx"
     "test_clientip" => "clientip---->%{clientip}"
    }


    #添加tag
    add_tag => [ "linux","web","nginx","test" ]


    #移除tag
    remove_tag => [ "linux","test" ]

  }
}

output {
  stdout {}
}
EOF
```

date 插件修改写入 ES 的时间案例

```
cat >> stdin-date-es.conf << EOF
input {
  file {
    #指定收集的路径
    path => "/var/log/messages"
  }
}


filter {

  json {
  #JSON解析器 可以将json形式的数据转换为logstash实际的数据结构 (根据key:value拆分成字段形式)
    source => "message"
  }


  date {
    #匹配时间字段并解析
    match => [ "log_time", "yyyy-MM-dd HH:mm:ss.SSS" ]
    #将匹配到的时间字段解析后存储到目标字段，默认字段为"@timestamp"
    target => "@timestamp"
    timezone => "Asia/Shanghai"
  }

}

output {
  stdout {}

  elasticsearch {
    #定义es集群的主机地址
    hosts => ["192.168.182.110:9200"]
    #定义索引名称
    index => "hqt-application-pro-%{+YYYY.MM.dd}"
  }
}
EOF
```

output 插件

output->redis




```
input {
  tcp {
     port => 9999
  }

}


output {
  stdout {}

  redis {
      # 指定redis的主机地址
      host => "10.0.0.101"
      # 指定redis的端口号
      port => "6379"
      # 指定redis数据库编号
      db => 10
      # 指定redis的密码
      password => "oldboyedu"
      # 指定写入数据的key类型
      data_type => "list"
      # 指定的写入的key名称
      key => "oldboyedu-linux80-logstash"
  }
}
```

```
[root@elk101.oldboyedu.com ~]# logstash -f config-logstash/07-tcp-to-redis.conf
Using bundled JDK: /usr/share/logstash/jdk
OpenJDK 64-Bit Server VM warning: Option UseConcMarkSweepGC was deprecated in version 9.0 and will likely be removed in a future release.
```

output->file

```
input {
  tcp {
     port => 9999
  }

}


output {
  stdout {}

  file {
      # 指定磁盘的落地位置
      path => /tmp/oldboyedu-linux80-logstash.log
  }
}
```

其他

logstash 的多 if 分支案例

```
cat >> homework-to-es.conf << EOF
input {
   beats {
      type => "test-nginx-applogs"
      port => 5044
   }
   file {
      type => "test-product-applogs"
      path => "/tmp/app.logs"
   }
   beats {
      type => "test-dw-applogs"
      port => 8888
   }
   file {
      type => "test-payment-applogs"
      path => "/tmp/payment.log"
   }
}


filter {
   if [type] == "test-nginx-applogs"{
      mutate {
```

```
        remove_field                                                                    =>
[ "tags","agent","input","log","ecs","version","@version","ident","referrer","auth","xff","referer",
"upstreamtime","upstreamhost","tcp_xff"]
      }
      geoip {
        source => "clientip"
        database                                                                        =>
"/hqtbj/hqtwww/logstash_workspace/data/plugins/filters/geoip/CC/GeoLite2-City.mmdb"
      }
      useragent {
        source => "http_user_agent"
      }
    }

    if [type] == "test-product-applogs" {
      mutate {
        split => { "message" => "|" }
      }
      mutate {
        add_field => {
          "user_id" => "%{[message][1]}"
          "action" => "%{[message][2]}"
          "svip" => "%{[message][3]}"
          "price" => "%{[message][4]}"
        }
      }
      mutate {
        convert => {
        "user_id" => "integer"
        "svip" => "boolean"
        "price" => "float"
        }
      }
    }

    if [type] in [ "test-dw-applogs","test-payment-applogs" ] {
      json {
        source => "message"
      }
      date {
        match => [ "log_time", "yyyy-MM-dd HH:mm:ss.SSS" ]
        target => "@timestamp"
      }
    }
```

```
}


output {
  stdout {}
  if [type] == "test-nginx-applogs" {
    elasticsearch {
      hosts => ["192.168.182.110:9200"]
      index => "test-nginx-logs-%{+YYYY.MM.dd}"
    }
  }

  if [type] == "test-product-applogs" {
    elasticsearch {
      hosts => ["192.168.182.110:9200"]
      index => "test-product-applogs-%{+YYYY.MM.dd}"
    }
  }

  if [type] in [ "test-dw-applogs","test-payment-applogs" ] {
    elasticsearch {
      hosts => ["192.168.182.110:9200"]
      index => "test-center-applogs-%{+YYYY.MM.dd}"
    }
  }
}
EOF
```

type 和 tags

```
input {
    stdin {
        add_field => {"key" => "value"}
        codec => "plain"
        tags => ["add"]
        type => "std"
    }
}
```

type 和 tags 是 logstash 事件中两个特殊的字段。通常来说我们会在输入区段中通过 type 来标记事件类型 —— 我们肯定是提前能知道这个事件属于什么类型的。而 tags 则是在数据处理过程中，由具体的插件来添加或者删除的。
最常见的用法是像下面这样：

```
input {
    stdin {
        type => "web"
    }
}
filter {
    if [type] == "web" {
        grok {
            match => ["message", %{COMBINEDAPACHELOG}]
        }
    }
}
output {
    if "_grokparsefailure" in [tags] {
        nagios_nsca {
            nagios_status => "1"
        }
    } else {
        elasticsearch {
        }
    }
}
```

kibana

Kibana 的版本和 es 选一模一样的

启动

linux

cd /root/kibana

vim /root/kibana/kibana-8.13.3/config/kibana.yml
./kibana-8.13.3/bin/kibana

windows

双击 kibana-8.13.3/bin/kibana.bat