第1章 ELK简介

```
E: elasticsearch 存储数据 java
L: logstash 收集,过滤,转发,匹配 java
K: kibana 过滤,分析,图形展示 java
F: filebeat 收集日志,过滤 go
```

第2章: 传统日志分析需求

```
1.找出访问网站频次最高的 IP 排名前十
2.找出访问网站排名前十的 URL
3.找出中午 10 点到 2 点之间 www 网站访问频次最高的 IP
4.对比昨天这个时间段和今天这个时间段访问频次有什么变化
5.对比上周这个时间和今天这个时间的区别
6.找出特定的页面被访问了多少次
7.找出有问题的 IP 地址，并告诉我这个 IP 地址都访问了什么页面，在对比前
几天他来过吗？他从什么时间段开
始访问的，什么时间段走了
8.找出来访问最慢的前十个页面并统计平均响应时间，对比昨天这也页面访问也
这么慢吗？
9.找出搜索引擎今天各抓取了多少次？抓取了哪些页面？响应时间如何？
10.找出伪造成搜索引擎的 IP 地址
11.5 分钟之内告诉我结果
```

第3章: 日志收集分类

```
代理层: nginx haproxy
web层: nginx tomcat java php
db层: mysql mongo redis es
系统层: message secure
```

第4章 准备ES单机环境

es实验环境配置

1.单节点ES配置

```
rpm -ivh elasticsearch-7.9.1-x86_64.rpm
```

```
cat > /etc/elasticsearch/elasticsearch.yml << 'EOF'
node.name: node-1
path.data: /var/lib/elasticsearch
path.logs: /var/log/elasticsearch
network.host: 127.0.0.1,10.0.0.51
http.port: 9200
discovery.seed_hosts: ["10.0.0.51"]
cluster.initial_master_nodes: ["10.0.0.51"]
EOF
systemctl daemon-reload
systemctl start elasticsearch.service
netstat -lntup|grep 9200
curl 127.0.0.1:9200
```

2.kibana安装部署

```
rpm -ivh kibana-7.9.1-x86_64.rpm
cat > /etc/kibana/kibana.yml << 'EOF'
server.port: 5601
server.host: "10.0.0.51"
elasticsearch.hosts: ["http://10.0.0.51:9200"]
kibana.index: ".kibana"
EOF
systemctl start kibana
```

旧环境安装

```
systemctl stop elasticsearch.service
rm -rf /var/lib/elasticsearch/*
cat > /etc/elasticsearch/elasticsearch.yml << 'EOF'
node.name: node-1
path.data: /var/lib/elasticsearch
path.logs: /var/log/elasticsearch
network.host: 127.0.0.1,10.0.0.51
http.port: 9200
discovery.seed_hosts: ["10.0.0.51"]
cluster.initial_master_nodes: ["10.0.0.51"]
```

```
EOF
systemctl restart elasticsearch.service

systemctl stop kibana.service
rm -rf /var/lib/kibana/*
cat > /etc/kibana/kibana.yml << 'EOF'
server.port: 5601
server.host: "10.0.0.51"
elasticsearch.hosts: ["http://10.0.0.51:9200"]
kibana.index: ".kibana"
EOF
systemctl start kibana
```

第5章 filebeat收集Nginx普通格式日志

0.更新系统时间

`ntpdate time1.aliyun.com`

1.安装nginx   web-7

```
[root@web-7 ~]# cat /etc/yum.repos.d/nginx.repo
[nginx-stable]
name=nginx stable repo
baseurl=http://nginx.org/packages/centos/$releasever/$basearch/
gpgcheck=0
enabled=1
gpgkey=https://nginx.org/keys/nginx_signing.key

[nginx-mainline]
name=nginx mainline repo
baseurl=http://nginx.org/packages/mainline/centos/$releasever/$arch/
gpgcheck=0
enabled=0
gpgkey=https://nginx.org/keys/nginx_signing.key

yum makecache fast
yum install nginx -y
systemctl start nginx
```

1.Nginx配置 web-7

```
rm -rf /etc/nginx/conf.d/*
rm -rf /var/log/nginx/*
cat > /etc/nginx/conf.d/web.conf << 'EOF'
server {
 listen 80;
 server_name www.oldboy.com;
 root /code/www;
 index index.php index.html;
}
EOF
mkdir /code/www -p
echo web-7 > /code/www/index.html
systemctl restart nginx
curl 127.0.0.1
tail -f /var/log/nginx/access.log
```

2.安装filebeat

```
rpm -ivh filebeat-6.6.0-x86_64.rpm
rpm -qc filebeat
```

3.配置filebeat

```
cp /etc/filebeat/filebeat.yml /opt/
cat > /etc/filebeat/filebeat.yml << 'EOF'
filebeat.inputs:
- type: log
 enabled: true
 paths:
 - /var/log/nginx/access.log
setup.kibana:
output.elasticsearch:
 hosts: ["10.0.0.51:9200"]
```

```
EOF
```

5.启动并检查

```
systemctl start filebeat
tail -f /var/log/filebeat/filebeat
```

6.查看日志结果

```
es-head查看
filebeat-7.9.1-2021.07.14-000001
```

7.kibana添加

```
Management >> Index Patterns >> filebeat-7.9.1-2021.07.14-000001>>@timestamp >>create >> discover
```

第6章: filebeat收集Nginx的json格式日志

1.上面方案不完善的地方

所有日志都存储在message的value里,不能拆分单独显示

2.理想中的情况

```
可以把日志所有字段拆分出来
{
 $remote_addr : 192.168.12.254
 - : -
 $remote_user : -
 [$time_local]: [10/Sep/2019:10:52:08 +0800]
 $request: GET /jhdgsjfgjhshj HTTP/1.0
 $status : 404
 $body_bytes_sent : 153
  $http_referer : -
  $http_user_agent :ApacheBench/2.3
  $http_x_forwarded_for:-
 }
```

3.目标

如何使nginx日志格式转换成我们想要的json格式

4.修改nginx配置文件使日志转换成json web-7

把log_format main格式注释掉 下面添加如下:

```
vim /etc/nginx/nginx.conf
log_format json '{ "time_local": "$time_local", '
 '"remote_addr": "$remote_addr", '
 '"referer": "$http_referer", '
 '"request": "$request", '
 '"status": $status, '
 '"bytes": $body_bytes_sent, '
 '"agent": "$http_user_agent", '
 '"x_forwarded": "$http_x_forwarded_for", '
  '"up_addr": "$upstream_addr",'
  '"up_host": "$upstream_http_host",'
  '"upstream_time": "$upstream_response_time",'
  '"request_time": "$request_time"'
  ' }';
  access_log /var/log/nginx/access.log json;
```

清除旧日志

```
> /var/log/nginx/access.log
```

检查并重启nginx

```
nginx -t
systemctl restart nginx
```

5.nginx转换成json之后仍然不完善的地方

通过查看发现,虽然nginx日志变成了json,但是es里还是存储在message里仍然不能拆分

6.目标

如何在ES里展示的是json格式

7.修改filebeat配置文件支持json解析

```
cat >/etc/filebeat/filebeat.yml<<EOF
filebeat.inputs:
- type: log
```

```
enabled: true
paths:
- /var/log/nginx/access.log
json.keys_under_root: true
json.overwrite_keys: true

output.elasticsearch:
 hosts: ["10.0.0.51:9200"]
EOF
```

8.删除ES里以前的索引

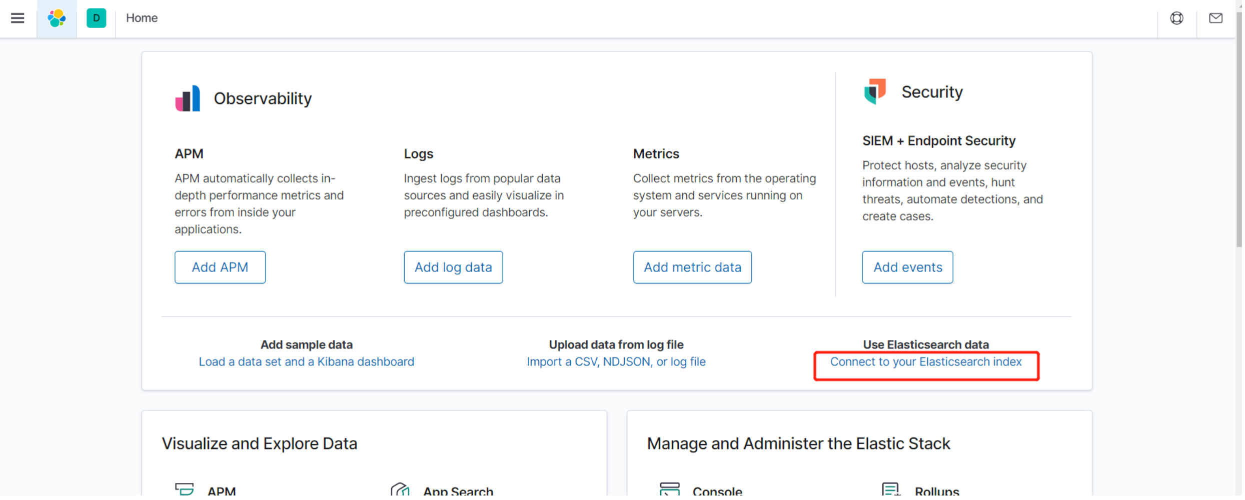
```
es-head >> filebeat-7.9.1-2021.07.14-000001 >> 动作 >>删除
```

9.重启filebeat

```
systemctl restart filebeat
```

10.访问并测试

11.kibana删除旧索引,创建新索引



Observability

APM

APM automatically collects in-depth performance metrics and errors from inside your applications.

Add APM

Logs

Ingest logs from popular data sources and easily visualize in preconfigured dashboards.

Add log data

Metrics

Collect metrics from the operating system and services running on your servers.

Add metric data

Security

SIEM + Endpoint Security

Protect hosts, analyze security information and events, hunt threats, automate detections, and create cases.

Add events

Add sample data
Load a data set and a Kibana dashboard

Upload data from log file
Import a CSV, NDJSON, or log file

Use Elasticsearch data
Connect to your Elasticsearch index

Visualize and Explore Data

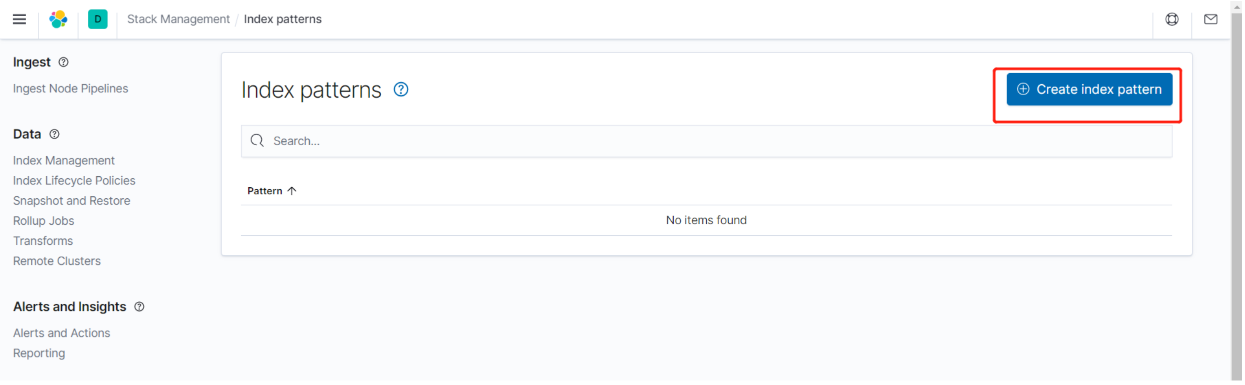
APM

App Search

Manage and Administer the Elastic Stack

Console

Rollups



D Stack Management / Index patterns

Ingest

Ingest Node Pipelines

Data

Index Management
Snapshot and Restore
Index Lifecycle Policies
Rollup Jobs
Transforms
Remote Clusters

Alerts and Insights

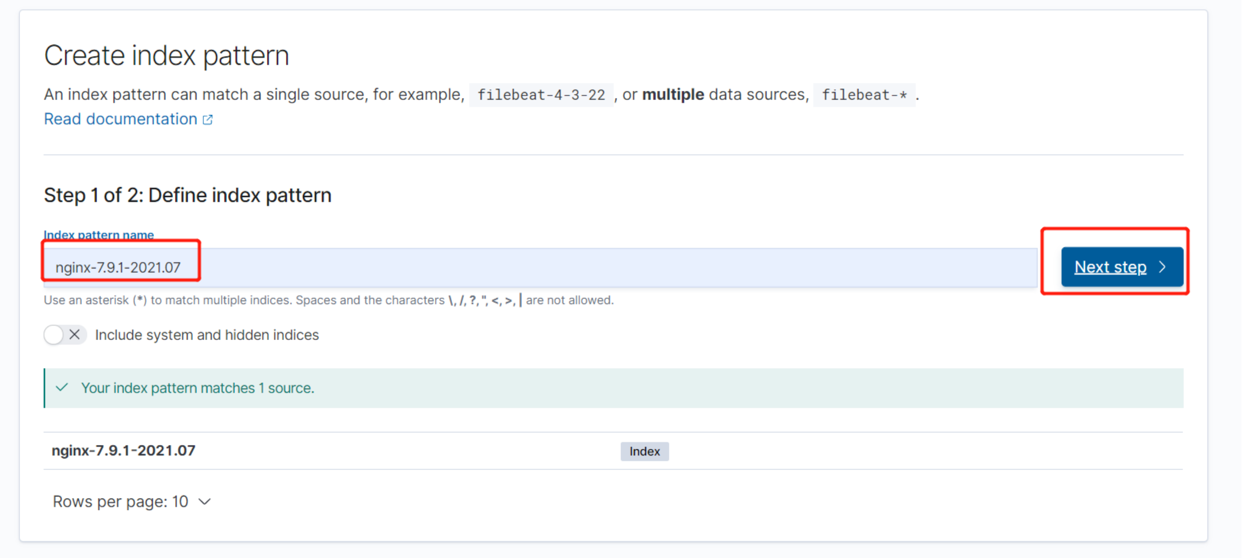
Alerts and Actions
Reporting

Index patterns

Create index pattern

Search...

Pattern ↑
No items found



Create index pattern

An index pattern can match a single source, for example, `filebeat-4-3-22`, or **multiple** data sources, `filebeat-*`.
Read documentation

Step 1 of 2: Define index pattern

Index pattern name

nginx-7.9.1-2021.07

Next step >

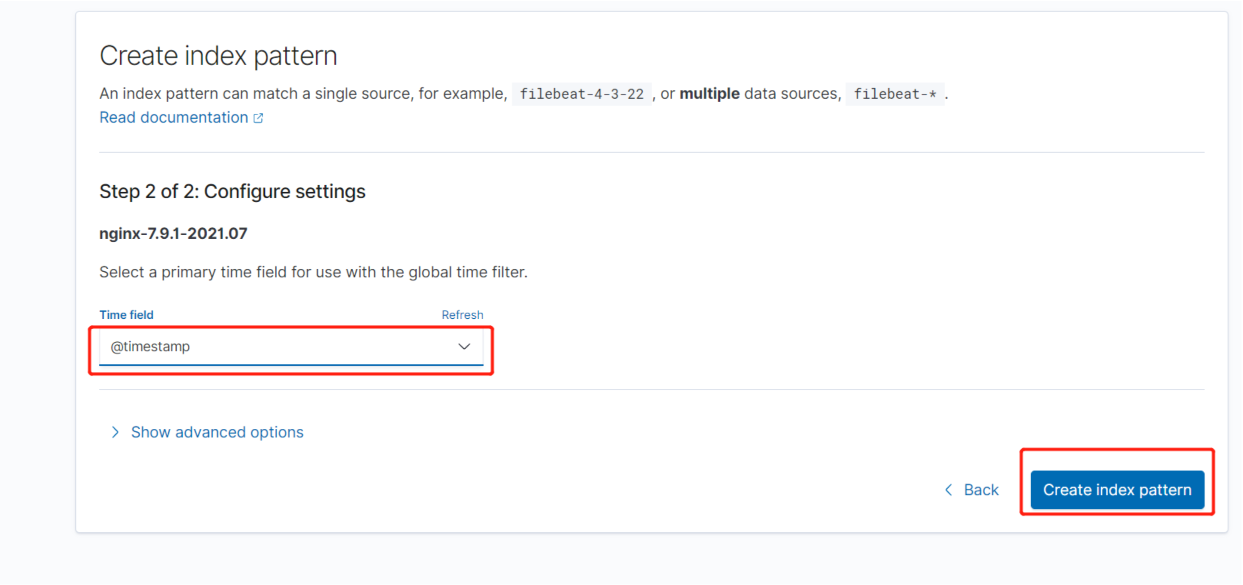
Use an asterisk (*) to match multiple indices. Spaces and the characters \, /, ?, ", <, >, | are not allowed.

○ Include system and hidden indices

✓ Your index pattern matches 1 source.

nginx-7.9.1-2021.07	Index

Rows per page: 10



Create index pattern

An index pattern can match a single source, for example, `filebeat-4-3-22`, or **multiple** data sources, `filebeat-*`.
Read documentation

Step 2 of 2: Configure settings

nginx-7.9.1-2021.07

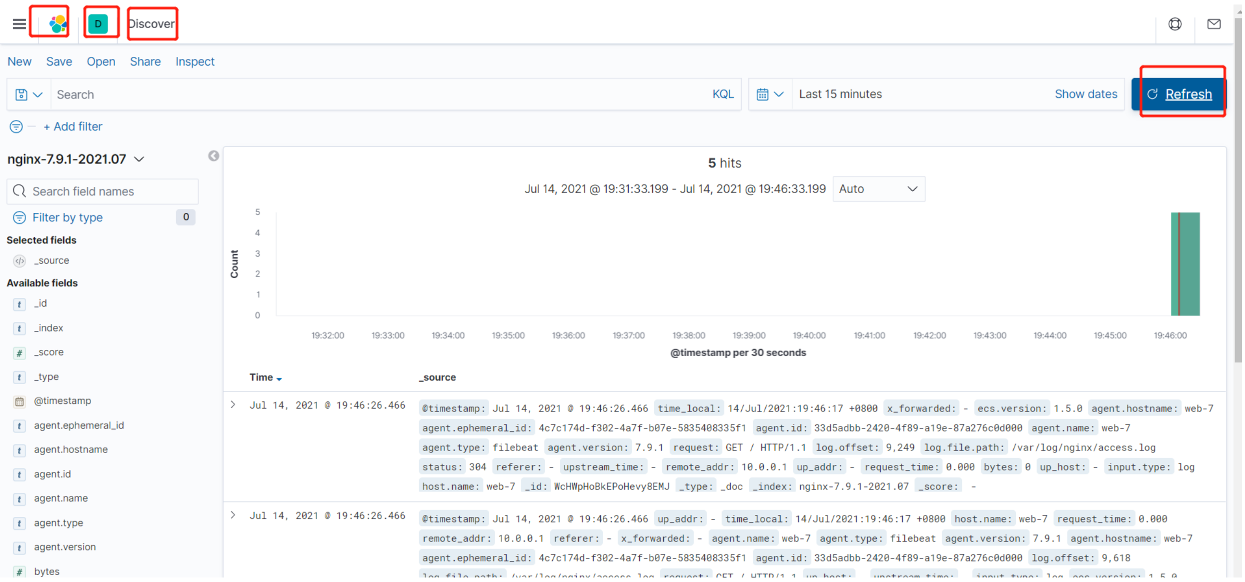
Select a primary time field for use with the global time filter.

Time field Refresh

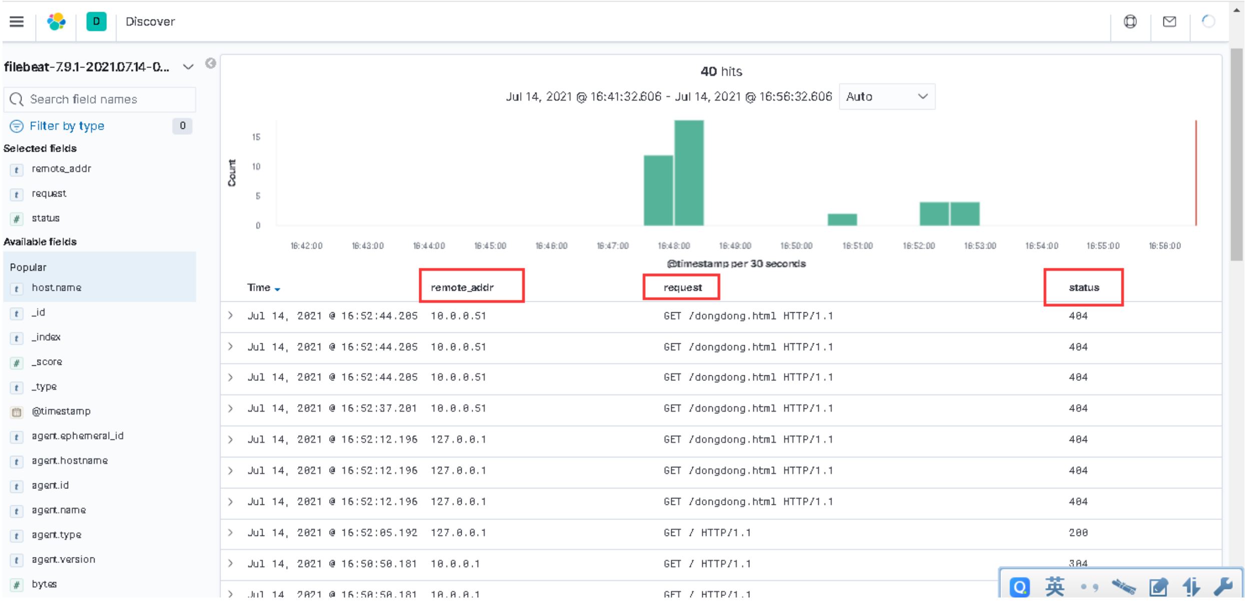
@timestamp

> Show advanced options

< Back Create index pattern



Time	_source
Jul 14, 2021 @ 19:46:26.466	@timestamp: Jul 14, 2021 @ 19:46:26.466 time_local: 14/Jul/2021:19:46:17 +0800 x_forwarded: - ecs.version: 1.5.0 agent.hostname: web-7 agent.ephemeral_id: 4c7c174d-f302-4a7f-b07e-5835408335f1 agent.id: 33d5adbb-2420-4f89-a19e-87a276c0d000 agent.name: web-7 agent.type: filebeat agent.version: 7.9.1 request: GET / HTTP/1.1 log.offset: 9,249 log.file.path: /var/log/nginx/access.log status: 304 referer: - upstream_time: - remote_addr: 10.0.0.1 up_addr: - request_time: 0.000 bytes: 0 up_host: - input.type: log host.name: web-7 _id: WcHWpHoBkEPoHevy8EMJ _type: _doc _index: nginx-7.9.1-2021.07 _score: -
Jul 14, 2021 @ 19:46:26.466	@timestamp: Jul 14, 2021 @ 19:46:26.466 up_addr: - time_local: 14/Jul/2021:19:46:17 +0800 host.name: web-7 request_time: 0.000 remote_addr: 10.0.0.1 referer: - x_forwarded: - agent.name: web-7 agent.type: filebeat agent.version: 7.9.1 agent.hostname: web-7 agent.ephemeral_id: 4c7c174d-f302-4a7f-b07e-5835408335f1 agent.id: 33d5adbb-2420-4f89-a19e-87a276c0d000 log.offset: 9,618



Time	remote_addr	request	status
Jul 14, 2021 @ 16:52:44.205	10.0.0.51	GET /dongdong.html HTTP/1.1	404
Jul 14, 2021 @ 16:52:44.205	10.0.0.51	GET /dongdong.html HTTP/1.1	404
Jul 14, 2021 @ 16:52:44.205	10.0.0.51	GET /dongdong.html HTTP/1.1	404
Jul 14, 2021 @ 16:52:37.201	10.0.0.51	GET /dongdong.html HTTP/1.1	404
Jul 14, 2021 @ 16:52:12.196	127.0.0.1	GET /dongdong.html HTTP/1.1	404
Jul 14, 2021 @ 16:52:12.196	127.0.0.1	GET /dongdong.html HTTP/1.1	404
Jul 14, 2021 @ 16:52:12.196	127.0.0.1	GET /dongdong.html HTTP/1.1	404
Jul 14, 2021 @ 16:52:05.192	127.0.0.1	GET / HTTP/1.1	200
Jul 14, 2021 @ 16:50:50.181	10.0.0.1	GET / HTTP/1.1	304
Jul 14, 2021 @ 16:50:50.181	10.0.0.1	GET / HTTP/1.1	

第7章 filebeat自定义ES索引名称

1.理想中的索引名称

```
nginx-7.9.1-2021.07
```

2.filebeat配置

```
vim /etc/filebeat/filebeat.yml
filebeat.inputs:
- type: log
 enabled: true
 paths:
 - /var/log/nginx/access.log
```

```
json.keys_under_root: true
json.overwrite_keys: true

output.elasticsearch:
 hosts: ["10.0.0.51:9200"]
 index: "nginx-%{[agent.version]}-%{+yyyy.MM}"

setup.ilm.enabled: false
setup.template.enabled: false

logging.level: info
logging.to_files: true
logging.files:
 path: /var/log/filebeat
 name: filebeat
 keepfiles: 7
 permissions: 0644
```