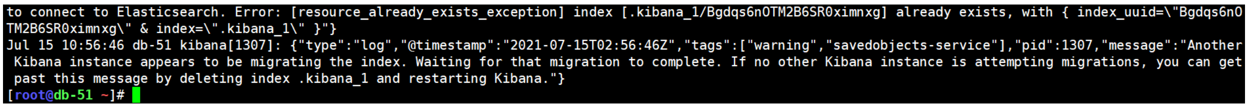
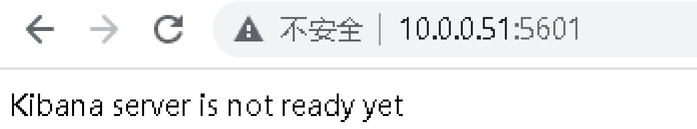
报错：



```
to connect to Elasticsearch. Error: [resource_already_exists_exception] index [.kibana_1/Bgdqs6nOTM2B6SR0ximnxg] already exists, with { index_uuid=\"Bgdqs6nO
TM2B6SR0ximnxg\" & index=\".kibana_1\" }"}
Jul 15 10:56:46 db-51 kibana[1307]: {"type":"log","@timestamp":"2021-07-15T02:56:46Z","tags":["warning","savedobjects-service"],"pid":1307,"message":"Another
 Kibana instance appears to be migrating the index. Waiting for that migration to complete. If no other Kibana instance is attempting migrations, you can get
 past this message by deleting index .kibana_1 and restarting Kibana."}
[root@db-51 ~]# 
```

登录kibana页面一直显示



Kibana server is not ready yet

不小心删除了kibana页面的所有索引，然后登录不进去

解决：删掉之前的elsticsearch目录和kibana目录，重新生成日志

```
systemctl stop elasticsearch.service
rm -rf /var/lib/elasticsearch/*
systemctl restart elasticsearch.service
systemctl stop kibana.service
rm -rf /var/lib/kibana/*
systemctl start kibana
```

再重新登录kibana页面就好了

使用filebeat区分不同日志索引

目前还不太完善的地方

1.访问日志和错误日志混在一起了

2.访问日志的格式和错误日志也不一样

我们期望的结果:

nginx-acess-7.9.1-2021.07

nginx-error-7.9.1-2021.07

filebeat配置文件:

```
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

- type: log
 enabled: true
 paths:
 - /var/log/nginx/error.log

output.elasticsearch:
 hosts: ["10.0.0.51:9200"]
 indices:
 - index: "nginx-access-%{[agent.version]}-%{+yyyy.MM}"
 when.contains:
 log.file.path: "/var/log/nginx/access.log"
 - index: "nginx-error-%{[agent.version]}-%{+yyyy.MM}"
 when.contains:
 log.file.path: "/var/log/nginx/error.log"

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
```

使用ES-pipeline转换Nginx普通日志

0.grok转换语法:

```
127.0.0.1 ==> %{IP:clientip}
- ==> -
- ==> -
[08/Oct/2020:16:34:40 +0800] ==> \\[%{HTTPDATE:nginx.access.time}\\]
"GET / HTTP/1.1" ==> "%{DATA:nginx.access.info}"
200 ==> %{NUMBER:http.response.status_code:long}
5 ==> %{NUMBER:http.response.body.bytes:long}
"-" ==> "(-|%{DATA:http.request.referrer})"
"curl/7.29.0" ==> "(-|%{DATA:user_agent.original})"
 "-" ==> "(-|%{IP:clientip})"
```

1.修改nginx日志为普通格式

```
systemctl stop filebeat
> /var/log/nginx/access.log
vim /etc/nginx/nginx.conf
systemctl restart nginx
curl 127.0.0.1
cat /var/log/nginx/access.log
```

2.创建ES的pipeline

```
GET _ingest/pipeline
PUT _ingest/pipeline/pipeline-nginx-access
{
 "description" : "nginx access log",
 "processors": [
 {
 "grok": {
 "field": "message",
 "patterns": ["%{IP:clientip} - - \\[%{HTTPDATE:nginx.access.time}\\] \"%{DATA:nginx.access.info}\" %{NUMBER:http.response.status_code:long} %{NUMBER:http.response.body.bytes:long} \"(-|%{DATA:http.request.referrer})\" \"(-|%{DATA:user_agent.original})\""]
  }
  },{
  "remove": {
  "field": "message"
 }
 }
 ]
}
```

3.修改filebeat配置文件

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

- type: log
 enabled: true
 paths:
 - /var/log/nginx/error.log
 tags: ["error"]

processors:
 - drop_fields:
 fields: ["ecs","log"]

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

 pipelines:
 - pipeline: "pipeline-nginx-access"
 when.contains:
 tags: "access"

 indices:
 - index: "nginx-access-%{[agent.version]}-%{+yyyy.MM}"
 when.contains:
```

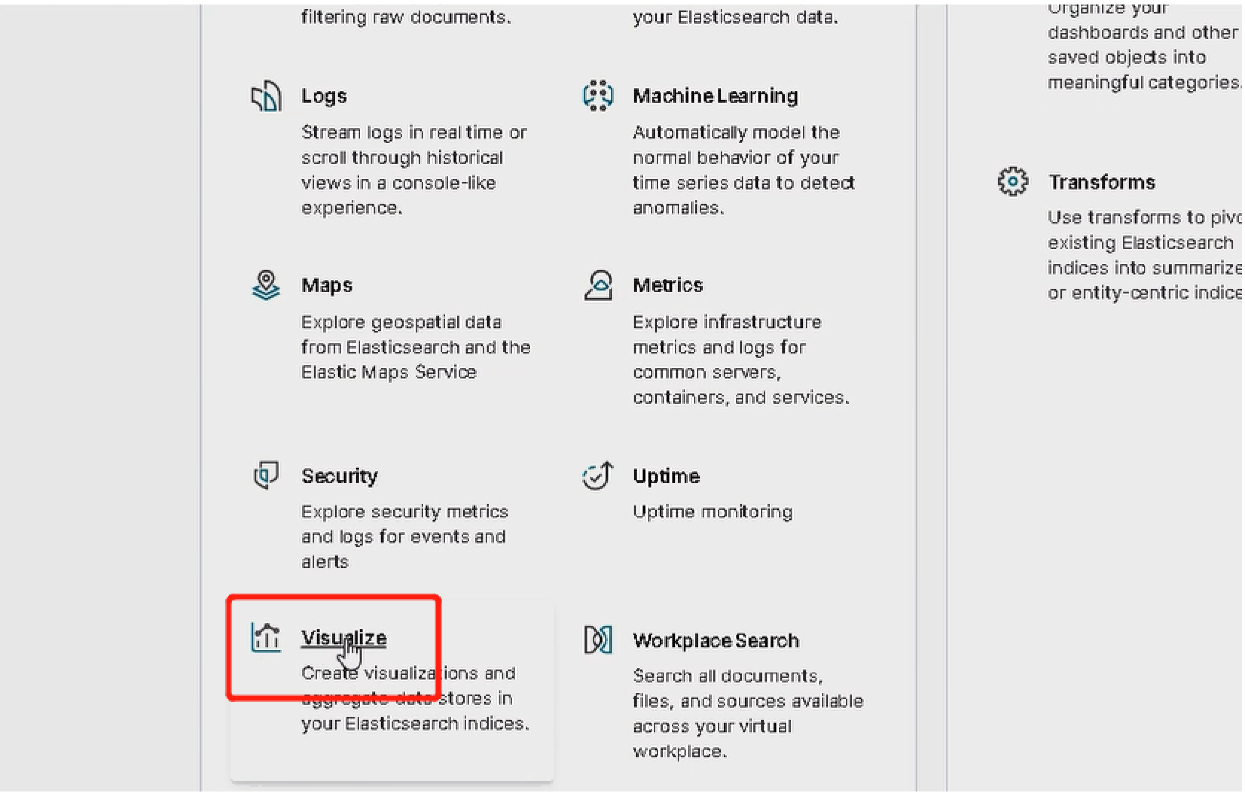
```
 tags: "access"

 - index: "nginx-error-%{[agent.version]}-%{+yyyy.MM}"
 when.contains:
 tags: "error"

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

logging.level: info
logging.to_files: true
```

使用kibana绘制日志面板



filtering raw documents.

your Elasticsearch data.

Organize your dashboards and other saved objects into meaningful categories.

Logs

Stream logs in real time or scroll through historical views in a console-like experience.

Machine Learning

Automatically model the normal behavior of your time series data to detect anomalies.

Transforms

Use transforms to pivo existing Elasticsearch indices into summarize or entity-centric indice

Maps

Explore geospatial data from Elasticsearch and the Elastic Maps Service

Metrics

Explore infrastructure metrics and logs for common servers, containers, and services.

Security

Explore security metrics and logs for events and alerts

Uptime

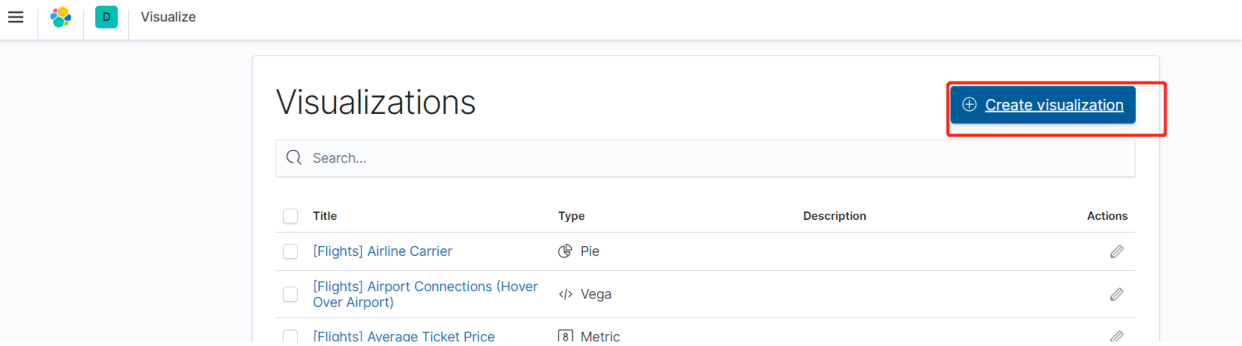
Uptime monitoring

Visualize

Create visualizations and aggregate data stores in your Elasticsearch indices.

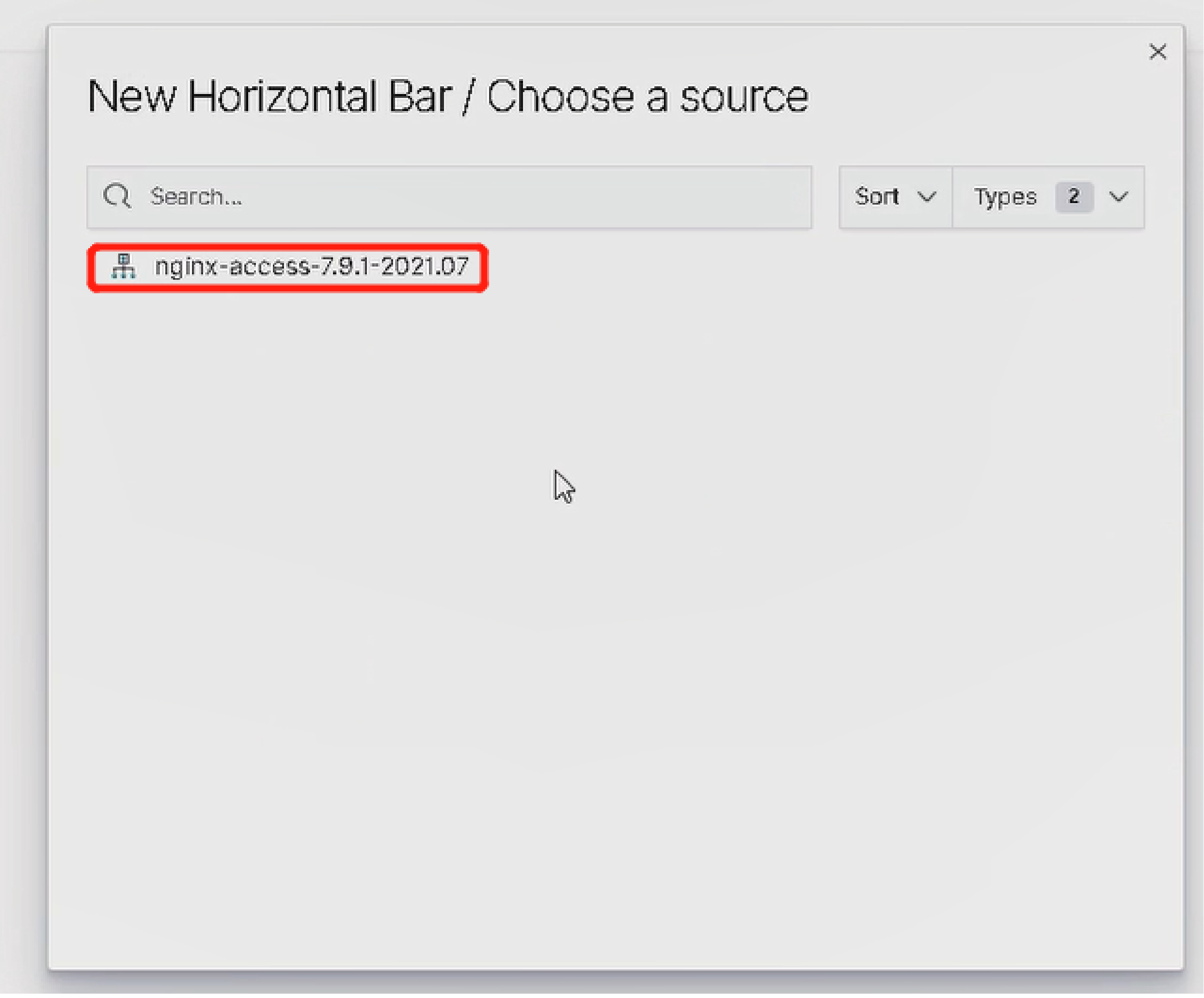
Workplace Search

Search all documents, files, and sources available across your virtual workplace.



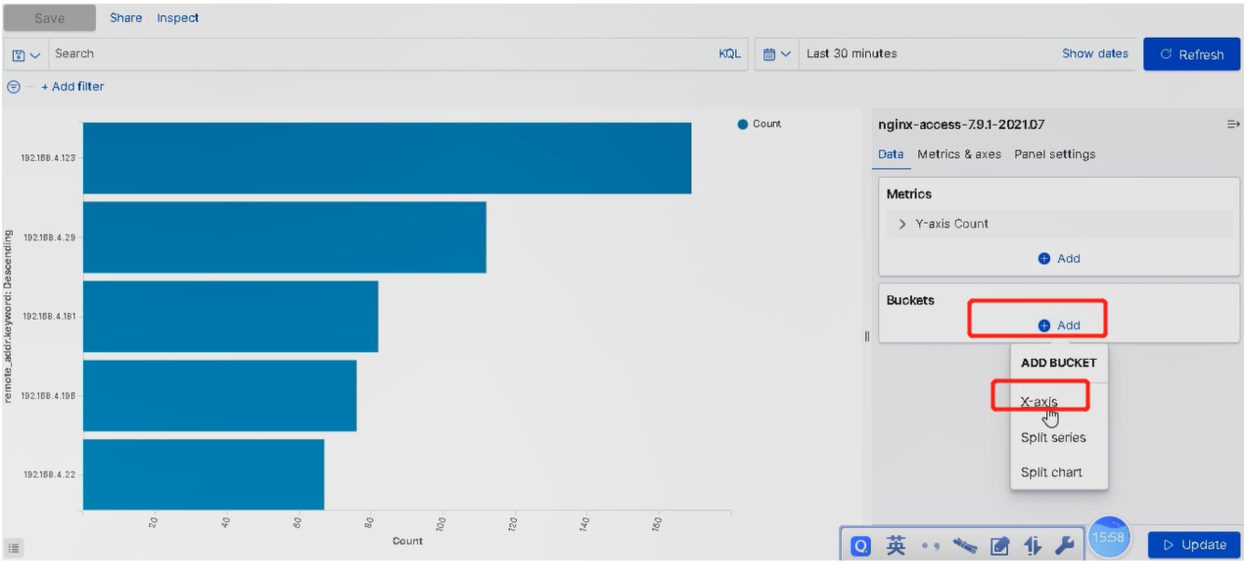
Visualizations

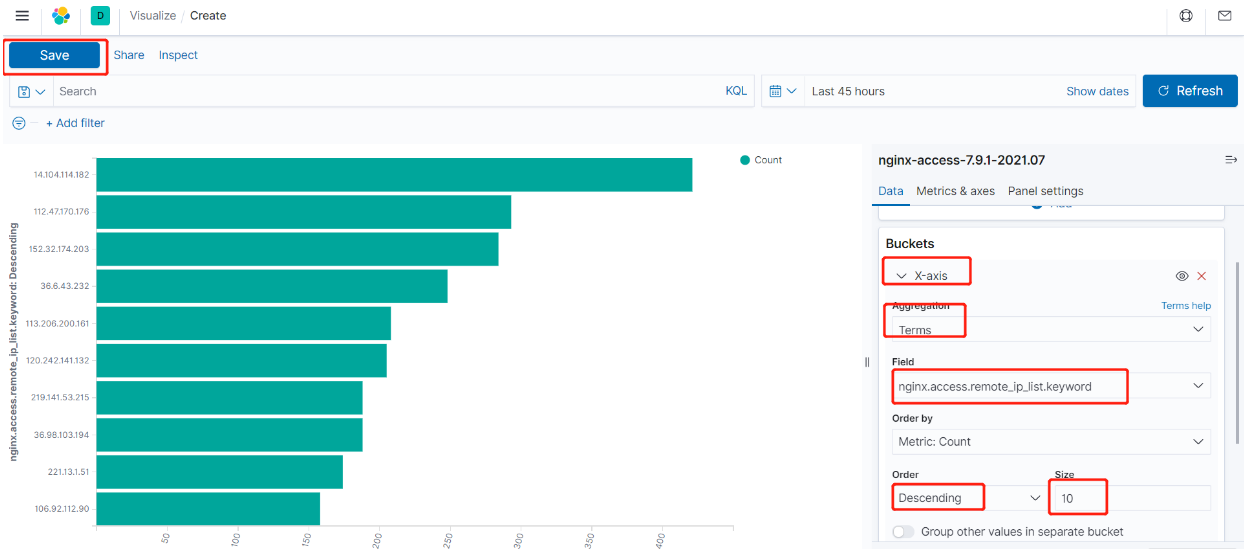
Title	Type	Description	Actions
[Flights] Airline Carrier	Pie		
[Flights] Airport Connections (Hover Over Airport)	Vega		
[Flights] Average Ticket Price	Metric		



New Horizontal Bar / Choose a source

nginx-access-7.9.1-2021.07





Aggregation: Terms

Field: nginx.access.remote_ip_list.keyword

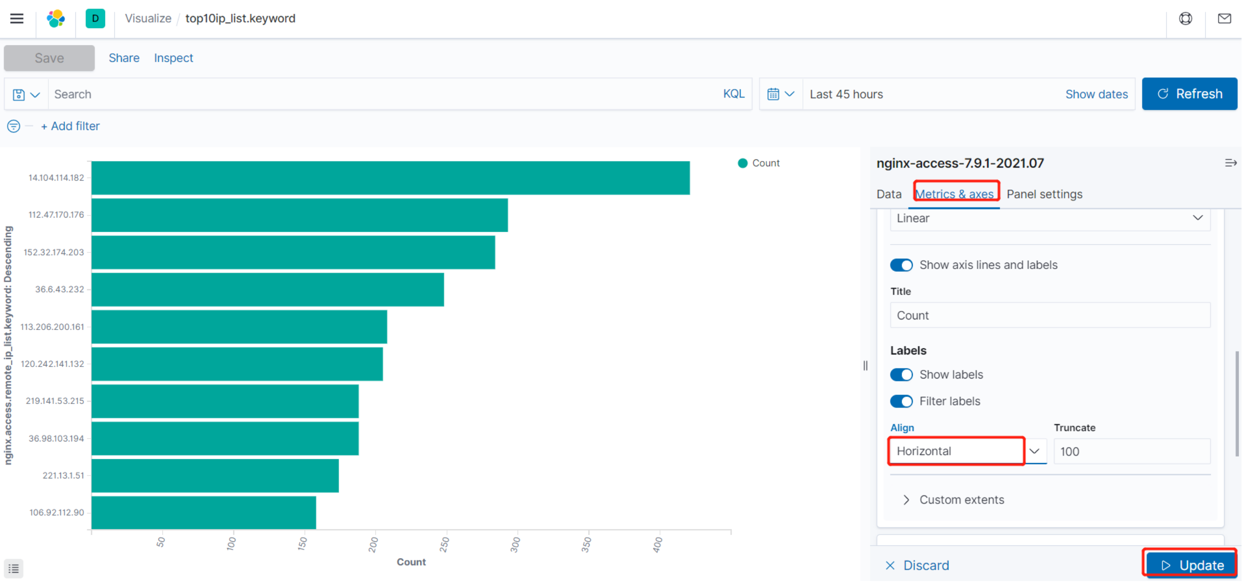
Order by: Metric: Count

Order: Descending

Size: 10

○ Group other values in separate bucket

调整y轴数字



Linear

● Show axis lines and labels

Title: Count

Labels

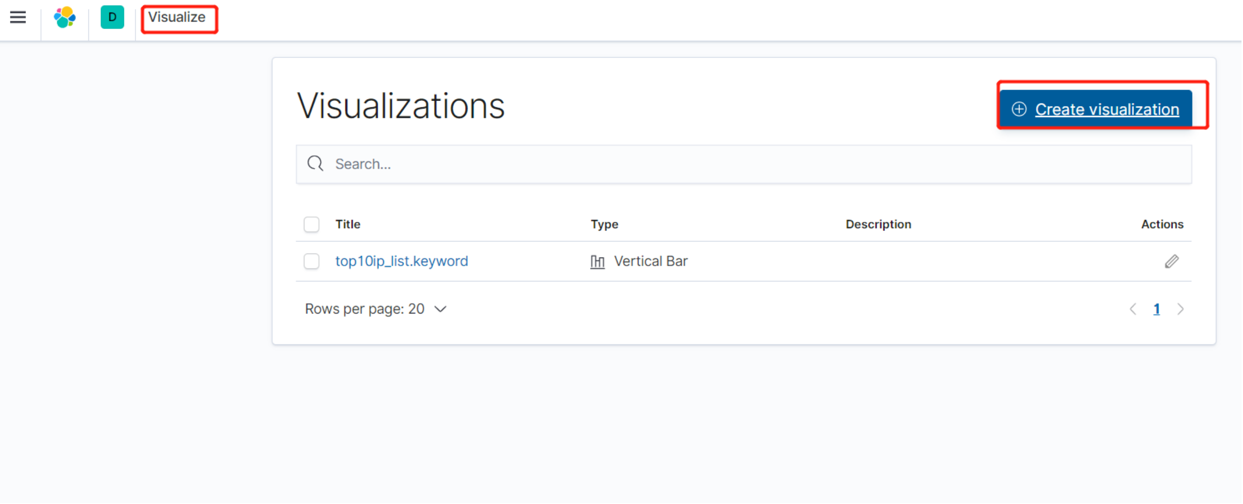
● Show labels

● Filter labels

Align: Horizontal

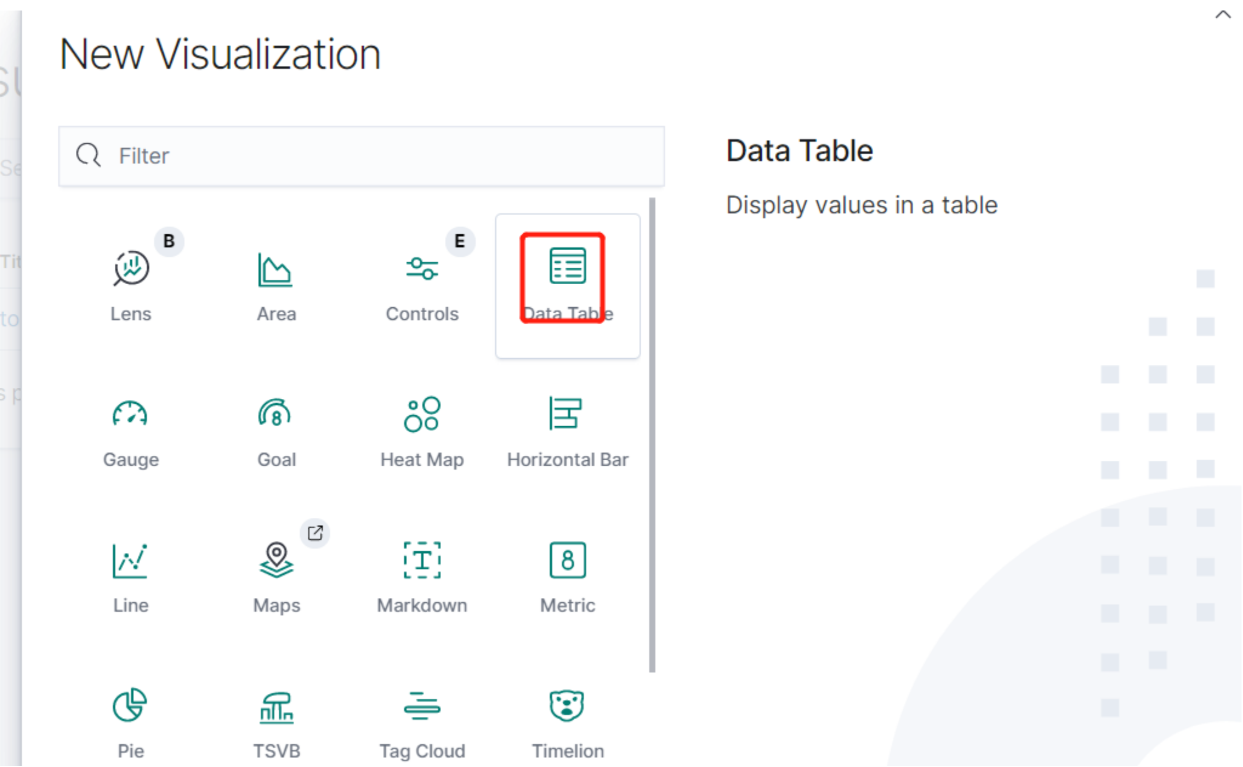
Truncate: 100

› Custom extents



Visualizations

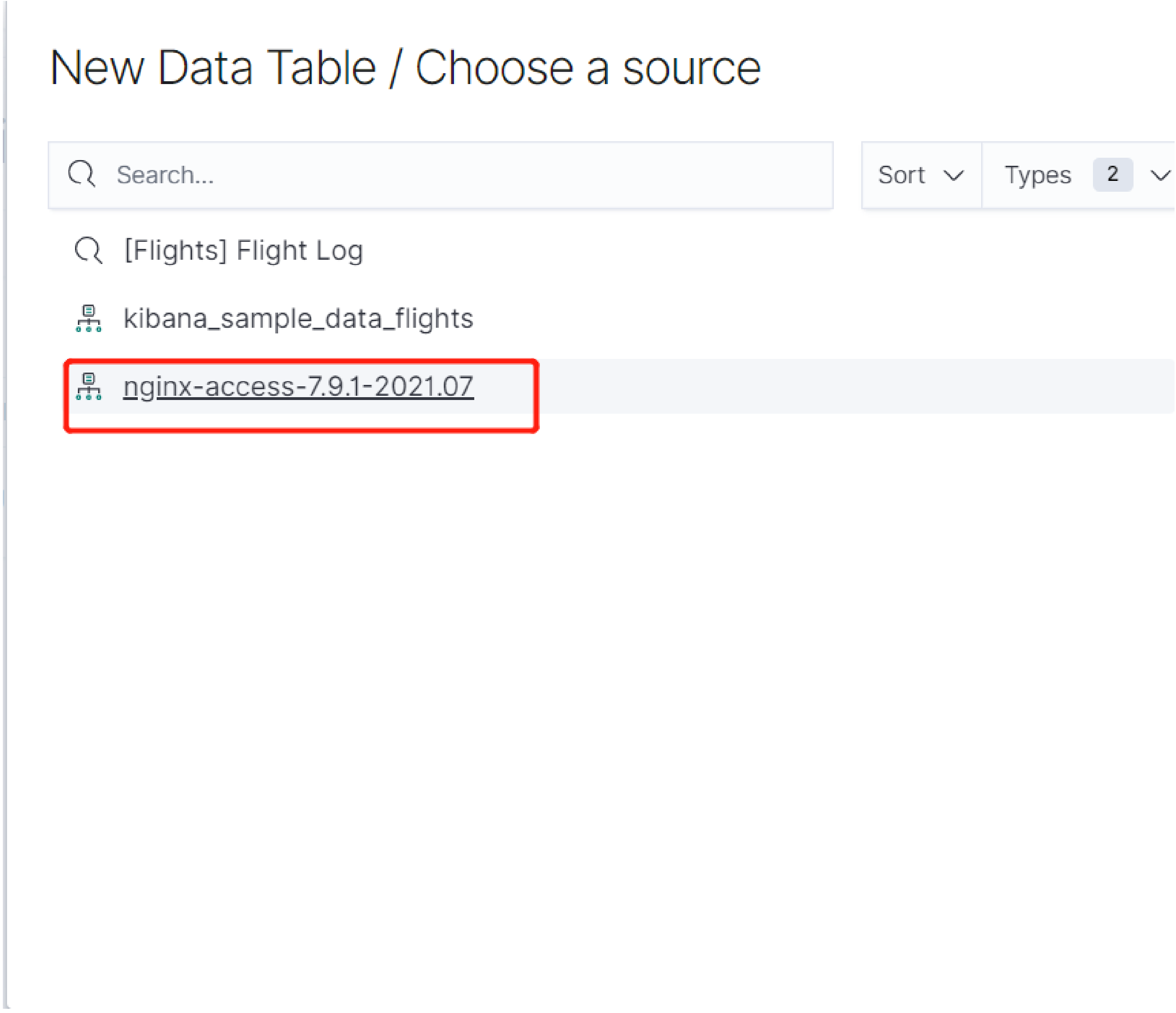
Title	Type	Description	Actions
top10ip_list.keyword	Vertical Bar		



New Visualization

Data Table

Display values in a table



New Data Table / Choose a source

Search...

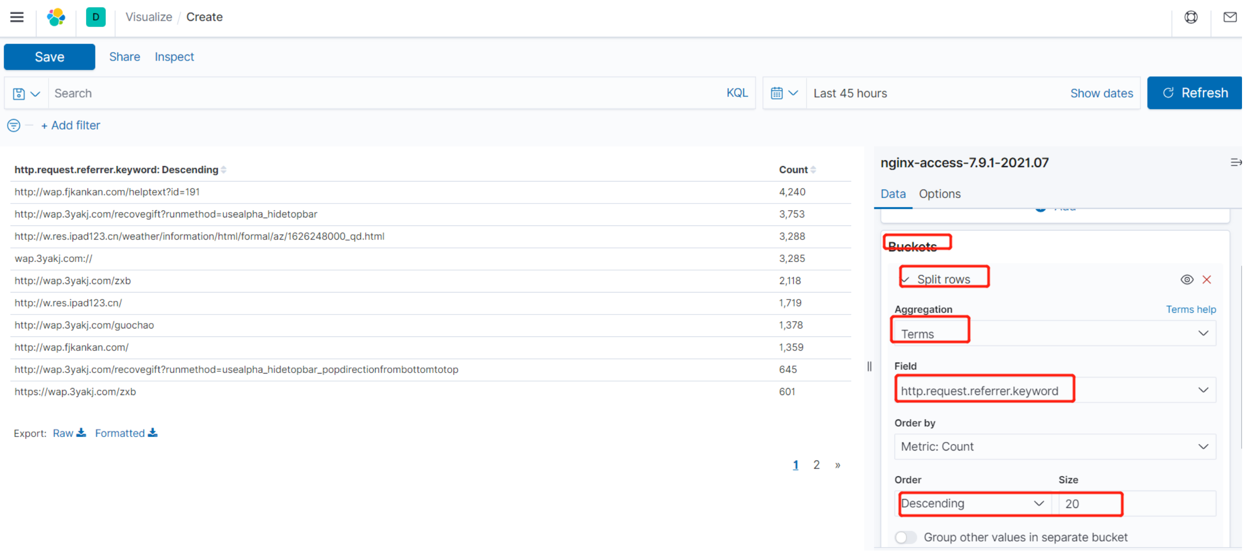
Sort

Types 2

[Flights] Flight Log

kibana_sample_data_flights

nginx-access-7.9.1-2021.07



Visualize / Create

Save Share Inspect

Search KQL

Last 45 hours Show dates Refresh

+ Add filter

http.request.referrer.keyword: Descending	Count
http://wap.fjkankan.com/helptext?id=191	4,240
http://wap.3yakj.com/recovegift?runmethod=usealpha_hidetopbar	3,753
http://w.res.ipad123.cn/weather/information/html/formal/az/1626248000_qd.html	3,288
wap.3yakj.com://	3,285
http://wap.3yakj.com/zxb	2,118
http://w.res.ipad123.cn/	1,719
http://wap.3yakj.com/guochao	1,378
http://wap.fjkankan.com/	1,359
http://wap.3yakj.com/recovegift?runmethod=usealpha_hidetopbar_popdirectionfrombottomtotop	645
https://wap.3yakj.com/zxb	601

Export: Raw Formatted

1 2 »

nginx-access-7.9.1-2021.07

Data Options

Buckets

Split rows

Aggregation: Terms

Field: http.request.referrer.keyword

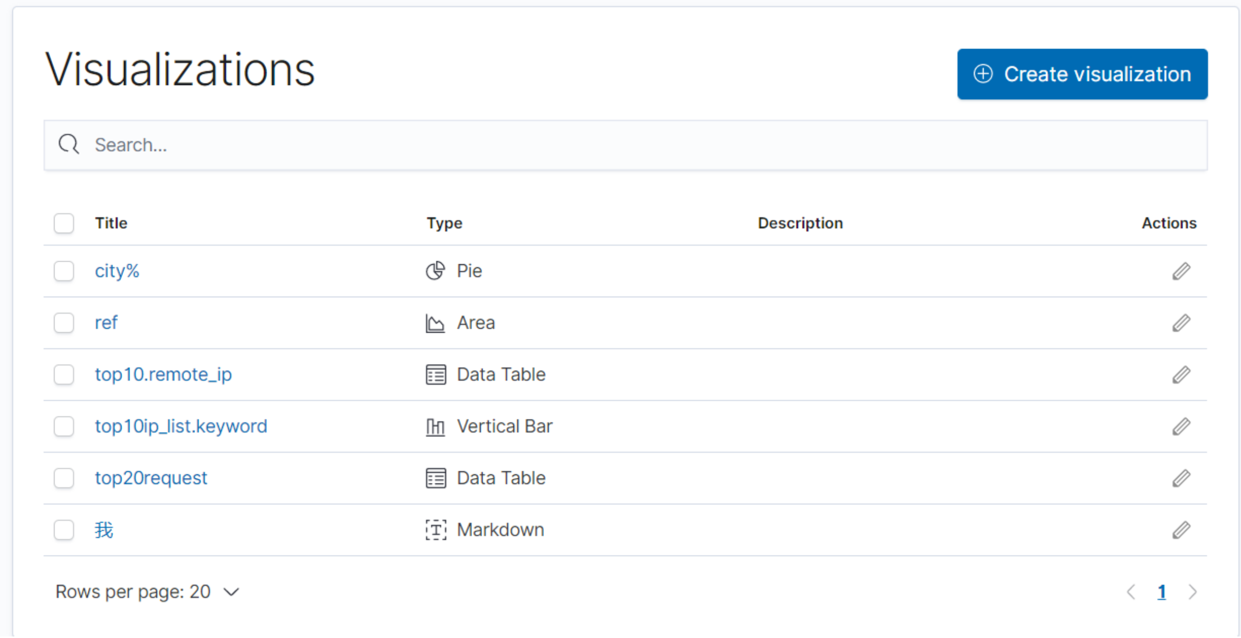
Order by: Metric: Count

Order: Descending

Size: 20

Group other values in separate bucket

继续选择添加不一样的图

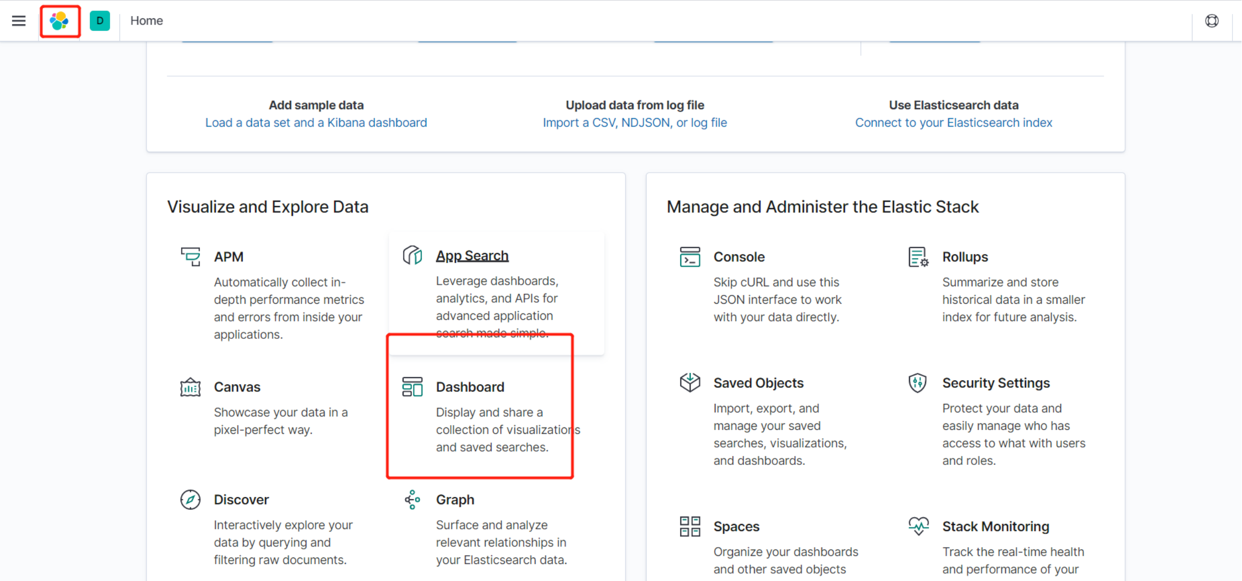


Visualizations

□	Title	Type	Description	Actions
□	city%	Pie		✎
□	ref	Area		✎
□	top10.remote_ip	Data Table		✎
□	top10ip_list.keyword	Vertical Bar		✎
□	top20request	Data Table		✎
□	我	Markdown		✎

Rows per page: 20

把刚刚画的放到一起



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
Automatically collect in-depth performance metrics and errors from inside your applications.

App Search
Leverage dashboards, analytics, and APIs for advanced application search made simple.

Canvas
Showcase your data in a pixel-perfect way.

Dashboard
Display and share a collection of visualizations and saved searches.

Discover
Interactively explore your data by querying and filtering raw documents.

Graph
Surface and analyze relevant relationships in your Elasticsearch data.

Manage and Administer the Elastic Stack

Console
Skip cURL and use this JSON interface to work with your data directly.

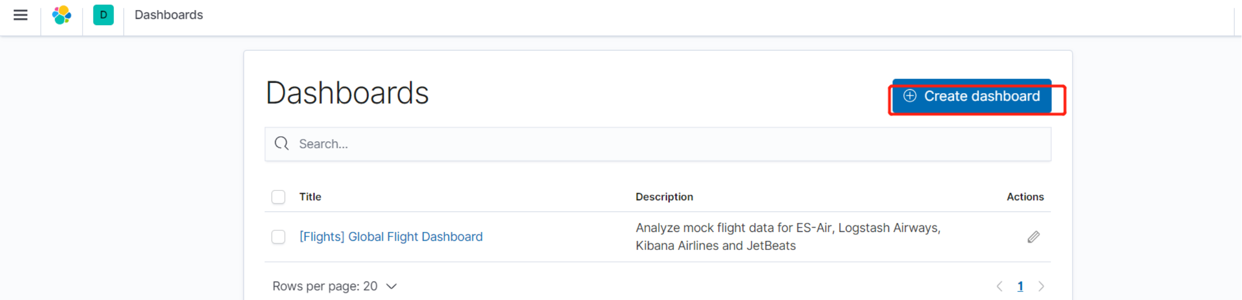
Rollups
Summarize and store historical data in a smaller index for future analysis.

Saved Objects
Import, export, and manage your saved searches, visualizations, and dashboards.

Security Settings
Protect your data and easily manage who has access to what with users and roles.

Spaces
Organize your dashboards and other saved objects

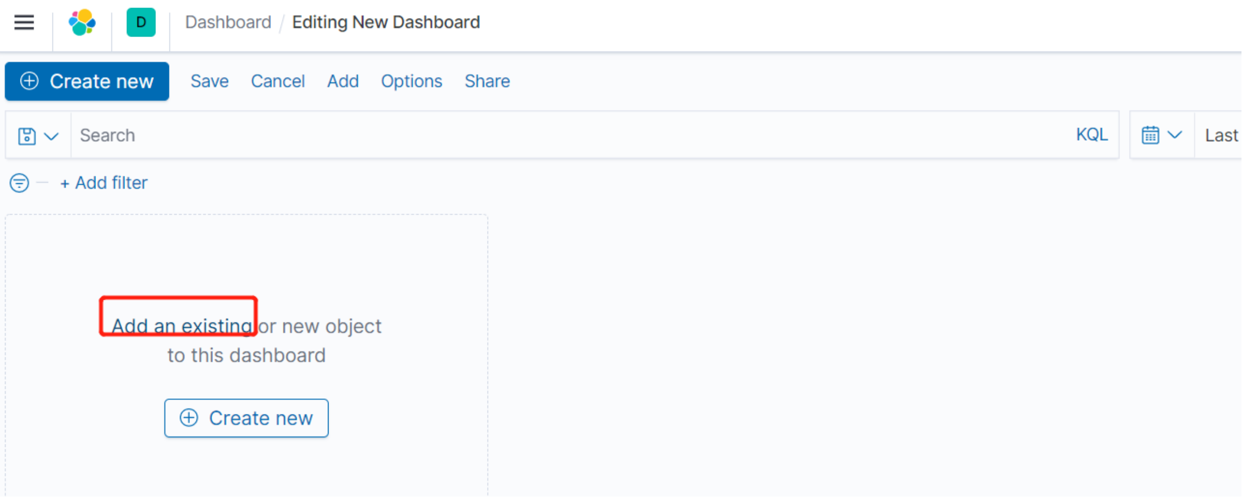
Stack Monitoring
Track the real-time health and performance of your



Dashboards

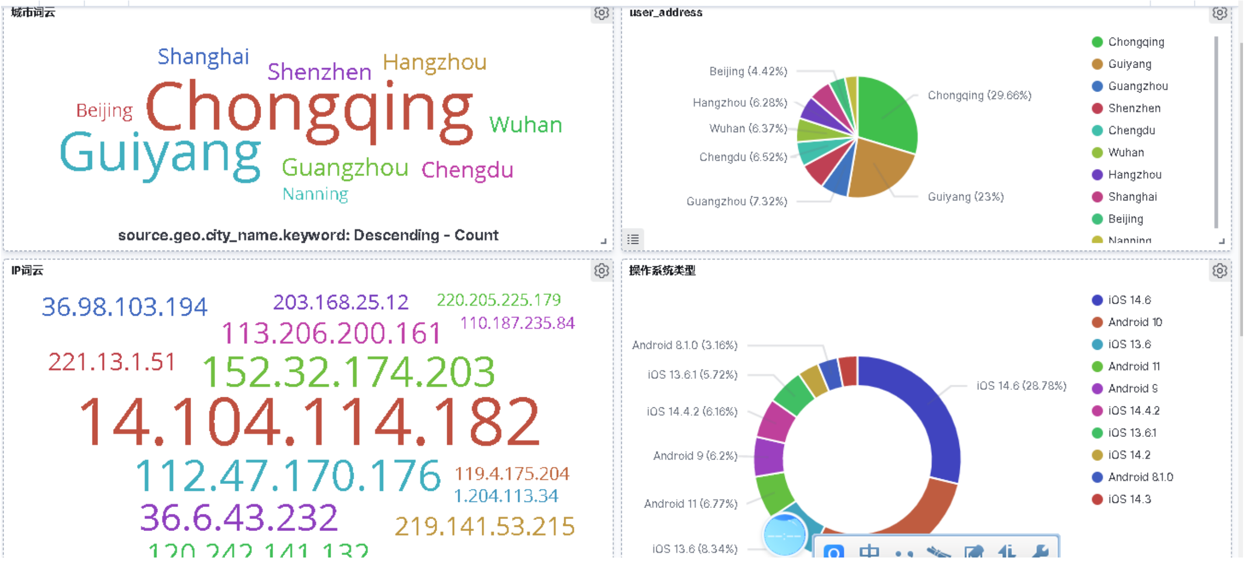
□	Title	Description	Actions
□	[Flights] Global Flight Dashboard	Analyze mock flight data for ES-Air, Logstash Airways, Kibana Airlines and JetBeats	✎

Rows per page: 20



Add an existing or new object to this dashboard

根据自己调整大小方向



使用filebeat模块将nginx普通日志转换成json格式

0.把nginx修改为普通格式日志

```
systemctl stop nginx
> /var/log/nginx/access.log
vim /etc/nginx/nginx.conf
access_log /var/log/nginx/access.log main;
----------------------------------------------
systemctl restart nginx
curl 127.0.0.1
tail -f /var/log/nginx/access.log
```

1.配置filebeat

```
cat > /etc/filebeat/filebeat.yml << 'EOF'
filebeat.config.modules:
 path: ${path.config}/modules.d/*.yml
 reload.enabled: true

output.elasticsearch:
 hosts: ["10.0.0.51:9200"]
 indices:
 - index: "nginx-access-%{[agent.version]}-%{+yyyy.MM}"
 when.contains:
 log.file.path: "/var/log/nginx/access.log"

 - index: "nginx-error-%{[agent.version]}-%{+yyyy.MM}"
 when.contains:
 log.file.path: "/var/log/nginx/error.log"

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

logging.level: info
logging.to_files: true
EOF
```

2.激活模块

```
filebeat modules list
filebeat modules enable nginx
filebeat modules list
```

3.配置日志路径

```
cat > /etc/filebeat/modules.d/nginx.yml << 'EOF'
- module: nginx
 access:
```

```
 enabled: true
 var.paths: ["/var/log/nginx/access.log"]
 error:
 enabled: true
 var.paths: ["/var/log/nginx/error.log"]
 ingress_controller:
 enabled: false
EOF
```

4.重启filebeat

```
systemctl restart filebeat
```

5.访问测试

使用filebeat模块收集MySQL慢日志

0.清空ES以前的索引

1.MySQL安装

```
yum install -y libaio-devel
rpm -qa|grep mariadb
yum remove mariadb-libs -y
rm -rf /etc/my.cnf
tar zxf mysql-5.7.28-linux-glibc2.12-x86_64.tar.gz -C /opt/
mv /opt/mysql-5.7.28-linux-glibc2.12-x86_64 /opt/mysql-5.7.28
ln -s /opt/mysql-5.7.28 /opt/mysql
echo 'export PATH=$PATH:/opt/mysql/bin' >>/etc/profile
source /etc/profile
mysql -V
useradd -s /sbin/nologin -M mysql
```

```
mkdir -p /data/mysql_3306/
chown -R mysql.mysql /data/
chown -R mysql.mysql /opt/mysql*
mysqld --initialize-insecure --user=mysql --basedir=/opt/mysql --datadir=/data/mysql_3306/
```

```
cat> /etc/my.cnf <<EOF
[mysqld]
port=3306
user=mysql
basedir=/opt/mysql
datadir=/data/mysql_3306
socket=/tmp/mysql.sock
slow_query_log=ON
slow_query_log_file=/data/mysql_3306/slow.log
long_query_time=1

[mysql]
socket=/tmp/mysql.sock
EOF
```

```
cp /opt/mysql/support-files/mysql.server /etc/init.d/mysqld
chkconfig --add mysqld
systemctl start mysqld
netstat -lntup|grep 3306
```

2.生成慢日志

```
mysql
select sleep(2) user from mysql.user;
select sleep(2) user,host from mysql.user;
```

3.filebeat激活mysql模块

```
filebeat modules list
filebeat modules enable mysql
filebeat modules list
```

4.配置filebeat模块文件

```
cat > /etc/filebeat/modules.d/mysql.yml << 'EOF'
- module: mysql
 error:
 enabled: true
 var.paths: ["/data/mysql_3306/web-7.err"]
 slowlog:
 enabled: true
 var.paths: ["/data/mysql_3306/slow.log"]
EOF
```

5.配置filebeat配置文件

```
cat > /etc/filebeat/filebeat.yml << 'EOF'
filebeat.config.modules:
 path: ${path.config}/modules.d/*.yml
 reload.enabled: true

output.elasticsearch:
 hosts: ["10.0.0.51:9200"]
 indices:
 - index: "nginx-access-%{[agent.version]}-%{+yyyy.MM}"
  when.contains:
  log.file.path: "/var/log/nginx/access.log"

  - index: "nginx-error-%{[agent.version]}-%{+yyyy.MM}"
  when.contains:
  log.file.path: "/var/log/nginx/error.log"

  - index: "mysql-error-%{[agent.version]}-%{+yyyy.MM}"
  when.contains:
  log.file.path: "/data/mysql_3306/web-7.err"

  - index: "mysql-slow-%{[agent.version]}-%{+yyyy.MM}"
  when.contains:
```

```
 log.file.path: "/data/mysql_3306/slow.log"

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

logging.level: info
logging.to_files: true
EOF
```

6.重启filebeat

```
systemctl restart filebeat
```

使用filebeat收集tomcat的json日志

1.安装tomcat

```
tar zxf apache-tomcat-8.0.27.tar.gz -C /opt/
tar zxf apache-tomcat-8.5.53.tar.gz -C /opt/
cd /opt
ln -s apache-tomcat-8.5.53 tomcat
/opt/tomcat/bin/startup.sh
netstat -lntup|grep 8080
ps -ef|grep tomcat
curl -I 127.0.0.1:8080
```

2.修改tomcat配置文件

```
[root@web-7 ~]# sed -n '137p' /opt/tomcat/conf/server.xml
 pattern="{"clientip":"%h","ClientUser":"%l","authenticated":"%u","AccessTime":"%t","method":"%r","status":"%s","SendBytes":"%b","Query?string":"%q","partner":"%{Referer}i","AgentVersion":"%{User-Agent}i"}"/>
```

3.重启tomcat

```
/opt/tomcat/bin/shutdown.sh
/opt/tomcat/bin/startup.sh
```

4.访问并查看日志是否为json格式

```
cat /opt/tomcat/logs/localhost_access_log.2021-07-15.txt
```

5.配置filebeat文件

```
cat > /etc/filebeat/filebeat.yml << 'EOF'
filebeat.inputs:
- type: log
 enabled: true
 paths:
 - /opt/tomcat/logs/localhost_access_log.*.txt
 json.keys_under_root: true
 json.overwrite_keys: true
 tags: ["tomcat"]

filebeat.config.modules:
 path: ${path.config}/modules.d/*.yml
 reload.enabled: true

output.elasticsearch:
 hosts: ["10.0.0.51:9200"]
 indices:
 - index: "nginx-access-%{[agent.version]}-%{+yyyy.MM}"
 when.contains:
 log.file.path: "/var/log/nginx/access.log"

 - index: "nginx-error-%{[agent.version]}-%{+yyyy.MM}"
 when.contains:
 log.file.path: "/var/log/nginx/error.log"

 - index: "mysql-error-%{[agent.version]}-%{+yyyy.MM}"
 when.contains:
 log.file.path: "/data/mysql_3306/web-7.err"

 - index: "mysql-slow-%{[agent.version]}-%{+yyyy.MM}"
 when.contains:
 log.file.path: "/data/mysql_3306/slow.log"

 - index: "tomcat-access-%{[agent.version]}-%{+yyyy.MM}"
 when.contains:
 tags: "tomcat"

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

logging.level: info
logging.to_files: true
EOF
```

6.重启filebeat

```
systemctl restart filebeat
```

7.访问测试