## ElasticSearch：

下载ElasticSearch安装包：

Linux：<https://artifacts.elastic.co/downloads/elasticsearch/elasticsearch-7.0.0-linux-x86_64.tar.gz>

Windows：<https://artifacts.elastic.co/downloads/elasticsearch/elasticsearch-7.0.0-windows-x86_64.zip>

解压：tar -xzf elasticsearch-7.0.0-linux-x86\_64.tar.gz

编辑ElasticSearch配置文件 elasticsearch-7.0.0/config/elasticsearch.yml，修改以下参数内容（示例）：

cluster.name: ck-elasticsearch

node.name: node-01

network.host: 10.0.1.4

discovery.seed\_hosts: ["127.0.0.1"]

cluster.initial\_master\_nodes: ["node-01"]

其中network.host的ip地址为VM的ifconfig得到的eth0地址（azure VM对外访问的nic内网地址）

切换到root，修改系统limit参数：

**运行：vi /etc/sysctl.conf，**添加：

vm.max\_map\_count=655350

保存后运行 **sysctl -p**

**运行：vi /etc/security/limits.conf**，加入：

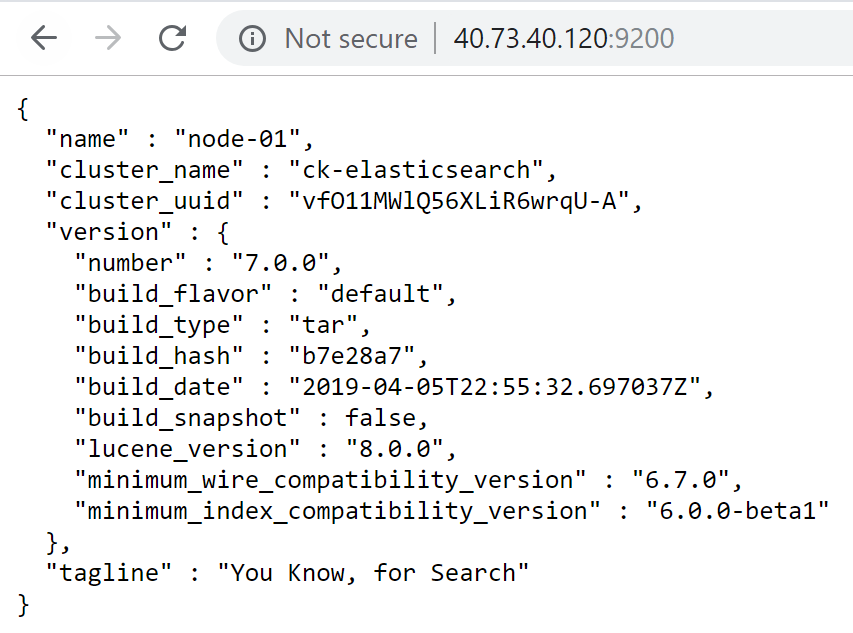
\* soft nofile 655350

\* hard nofile 655350

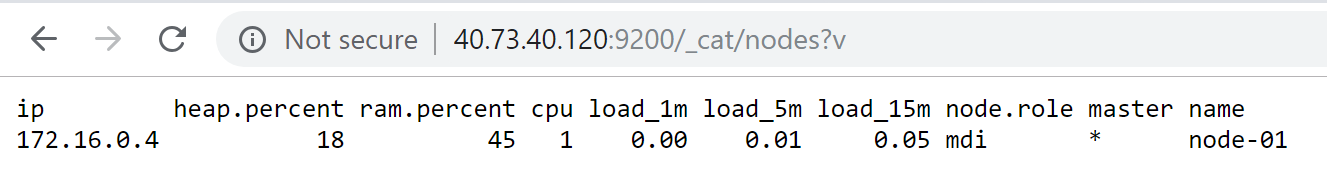
重新打开一个ssh终端，到 elasticsearch-7.0.0/bin 目录下，运行：**./elasticsearch -d**

到azure portal上，放开ElasticSearch所在虚机的9200端口的入站允许

使用浏览器或curl访问 <VM\_ip>:9200，应看到如下内容：



访问接口 <http://40.73.40.120:9200/_cat/nodes?v>，看到如下内容则ElasticSearch安装成功：



## Kibana：

下载Kibana 安装包：

Linux：<https://artifacts.elastic.co/downloads/kibana/kibana-7.0.0-linux-x86_64.tar.gz>

Windows：<https://artifacts.elastic.co/downloads/kibana/kibana-7.0.0-windows-x86_64.zip>

解压：tax -xzf kibana-7.0.0-linux-x86\_64.tar.gz

编辑ElasticSearch配置文件 kibana-7.0.0-linux-x86\_64/config/kibana.yml，修改以下参数内容（示例）：

server.port: 5601

server.host: "10.0.1.4"

elasticsearch.hosts: ["http://40.73.40.120:9200"]

kibana.index: ".kibana"

server.ssl.enabled: false

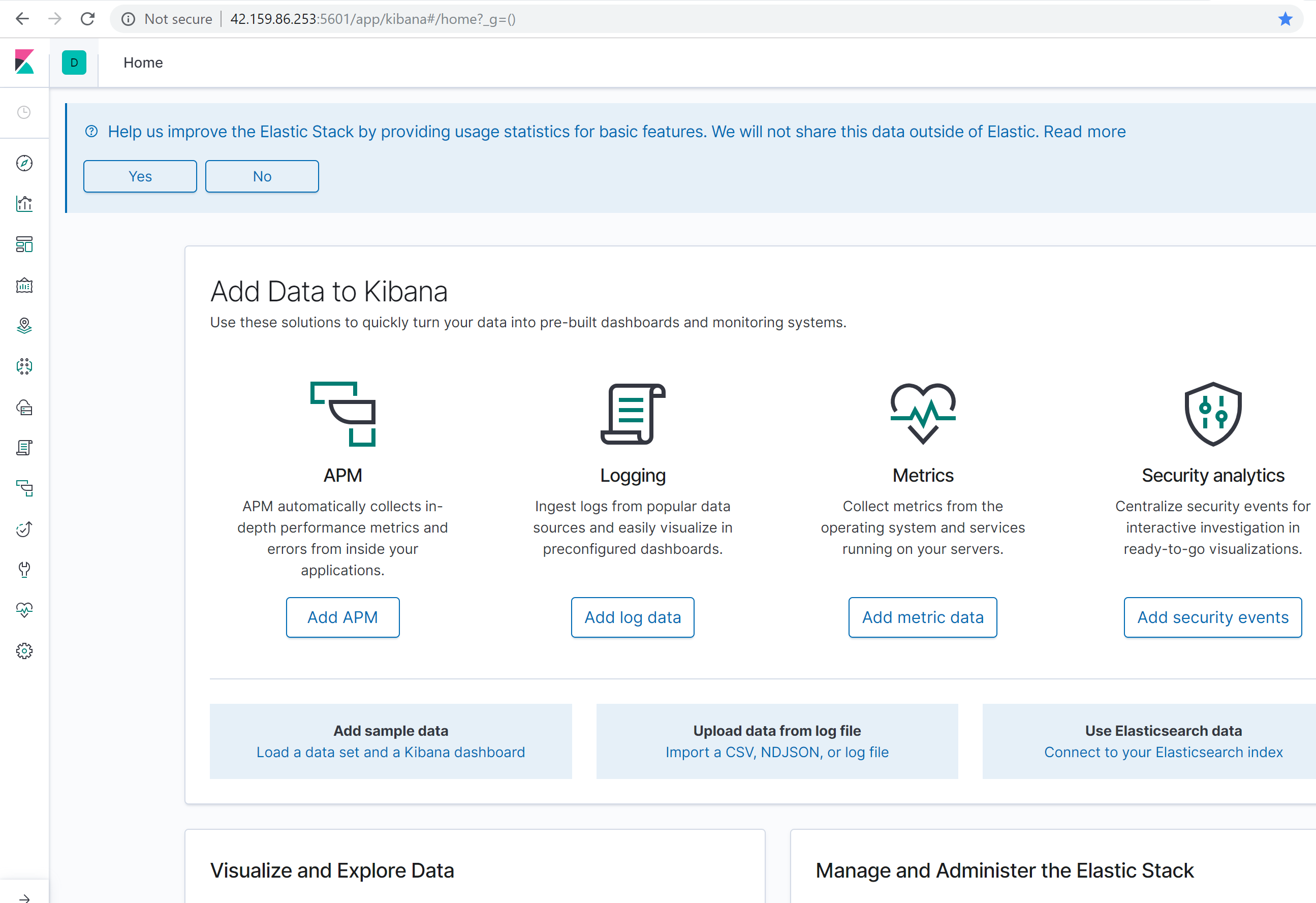
其中server.host的ip地址为VM的ifconfig得到的eth0地址（azure VM对外访问的nic内网地址）

elasticsearch.hosts为kibana可以访问到的ElasticSearch服务的地址和9200端口

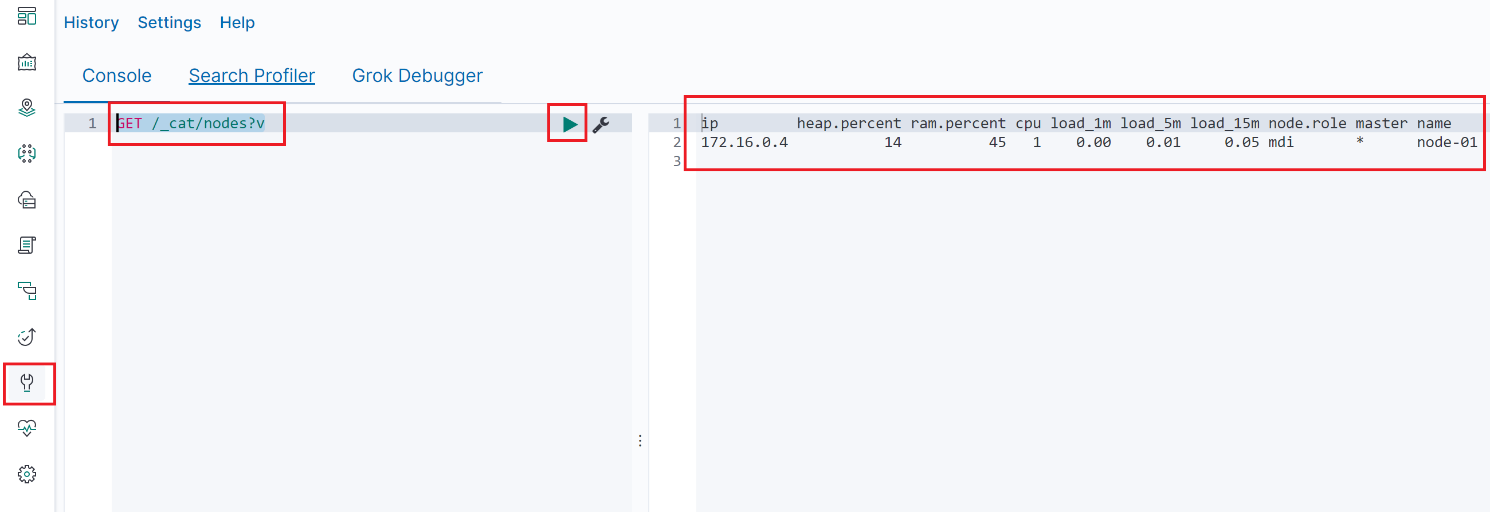
到azure portal上，放开Kibana所在虚机的5601端口的入站允许

到 kibana-7.0.0-linux-x86\_64/bin目录下，运行 **nohup ./kibana &**

使用浏览器或curl访问 <VM\_ip>:5601，看到如下内容则Kibana安装成功：



点击左侧工具栏的“Dev Tools”，执行API “GET /\_cat/nodes?v”，看到如下信息和以上ElasticSearch安装后执行API的内容一致，则Kibana和ElasticSearch集成完毕：



## Fluent-bit

执行命令创建dev namespace （namespace按实际需要命名）

kubectl create namespace dev

Git clone <https://github.com/fluent/fluent-bit-kubernetes-logging> 到本地

进入 fluent-bit-kubernetes-logging\ 目录，修改以下文件中的namespace值为dev：

fluent-bit-role-binding.yaml

fluent-bit-service-account.yaml

执行：

kubectl apply -f fluent-bit-service-account.yaml

kubectl apply -f fluent-bit-role.yaml

kubectl apply -f fluent-bit-role-binding.yaml

进入 fluent-bit-kubernetes-logging\output\elasticsearch\ 目录，根据实际情况修改文件fluent-bit-configmap.yaml 中以下highlight部分：

filter-kubernetes.conf: |

[FILTER]

Name kubernetes

Match kube.\*

Kube\_URL **https://ptfm-aks-u-ptfm-rsgp-uat-071a5d-3d6cbed2.hcp.chinaeast2.cx.prod.service.azk8s.cn:443**

Kube\_CA\_File /var/run/secrets/kubernetes.io/serviceaccount/ca.crt

Kube\_Token\_File /var/run/secrets/kubernetes.io/serviceaccount/token

Merge\_Log On

K8S-Logging.Parser On

K8S-Logging.Exclude Off

output-elasticsearch.conf: |

[OUTPUT]

Name es

Match \*

Host **42.159.86.253**

Port 9200

Logstash\_Format **On**

**Logstash\_Prefix aks-fluentbit**

Replace\_Dots On

Retry\_Limit False

进入 fluent-bit-kubernetes-logging\output\elasticsearch\ 目录，根据实际情况修改文件fluent-bit-ds.yaml 中以下highlight部分：

spec:

containers:

- name: fluent-bit

image: fluent/fluent-bit:1.0.6

imagePullPolicy: Always

ports:

- containerPort: 2020

env:

- name: FLUENT\_ELASTICSEARCH\_HOST

value: "**10.106.196.20**"

- name: FLUENT\_ELASTICSEARCH\_PORT

value: "9200"

volumeMounts:

- name: varlog

mountPath: /var/log

- name: varlibdockercontainers

mountPath: /var/lib/docker/containers

readOnly: true

- name: fluent-bit-config

mountPath: /fluent-bit/etc/

terminationGracePeriodSeconds: 10

执行命令：

kubectl apply -f fluent-bit-configmap.yaml

kubectl apply -f fluent-bit-ds.yaml

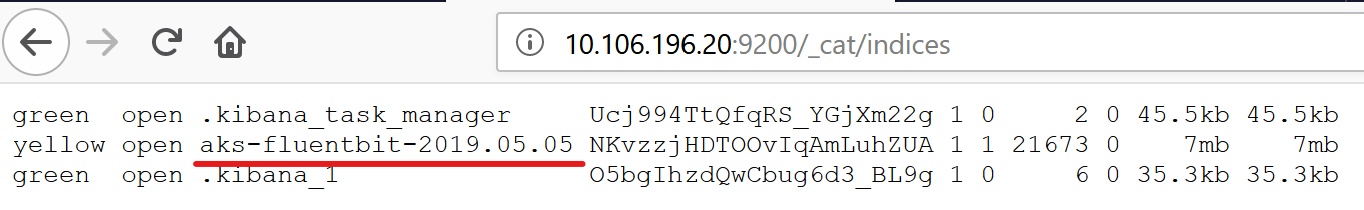
使用以下命令检查fluent-bit正常运行：

F:\AKS-env-setup>kubectl get daemonset -n dev

NAME DESIRED CURRENT READY UP-TO-DATE AVAILABLE NODE SELECTOR AGE

fluent-bit 3 3 3 3 3 <none> 52m

稍等片刻，到浏览器访问 <http://10.106.196.20:9200/_cat/indices>，得到以下输出：



其中 aks-fluentbit-2019.05.05 是fluentbit收集 AKS节点上的容器日志，在ElasticSearch中创建索引并上传日志内容。

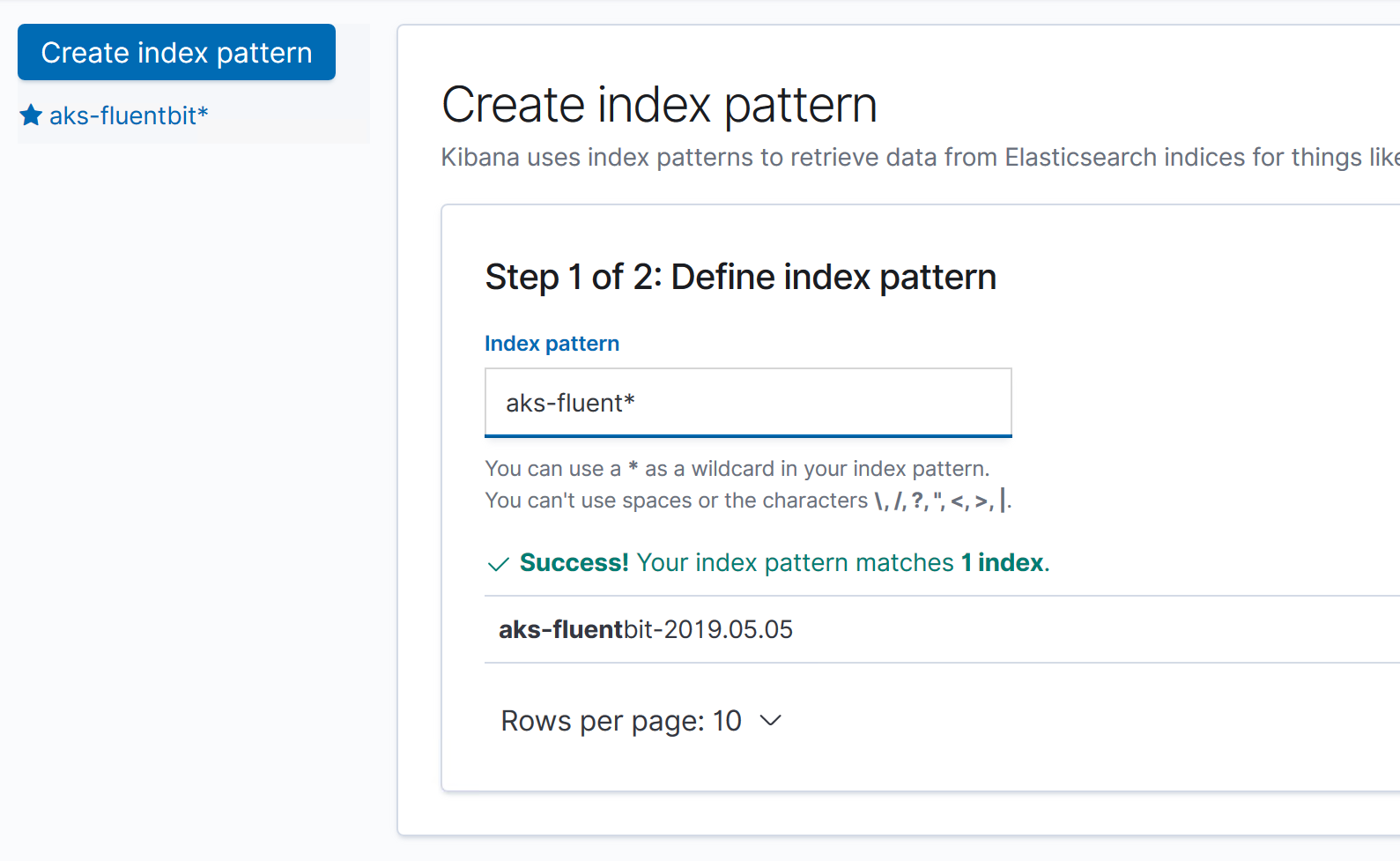
至此fluentbit配置完毕。

## 配置Kibana查询日志

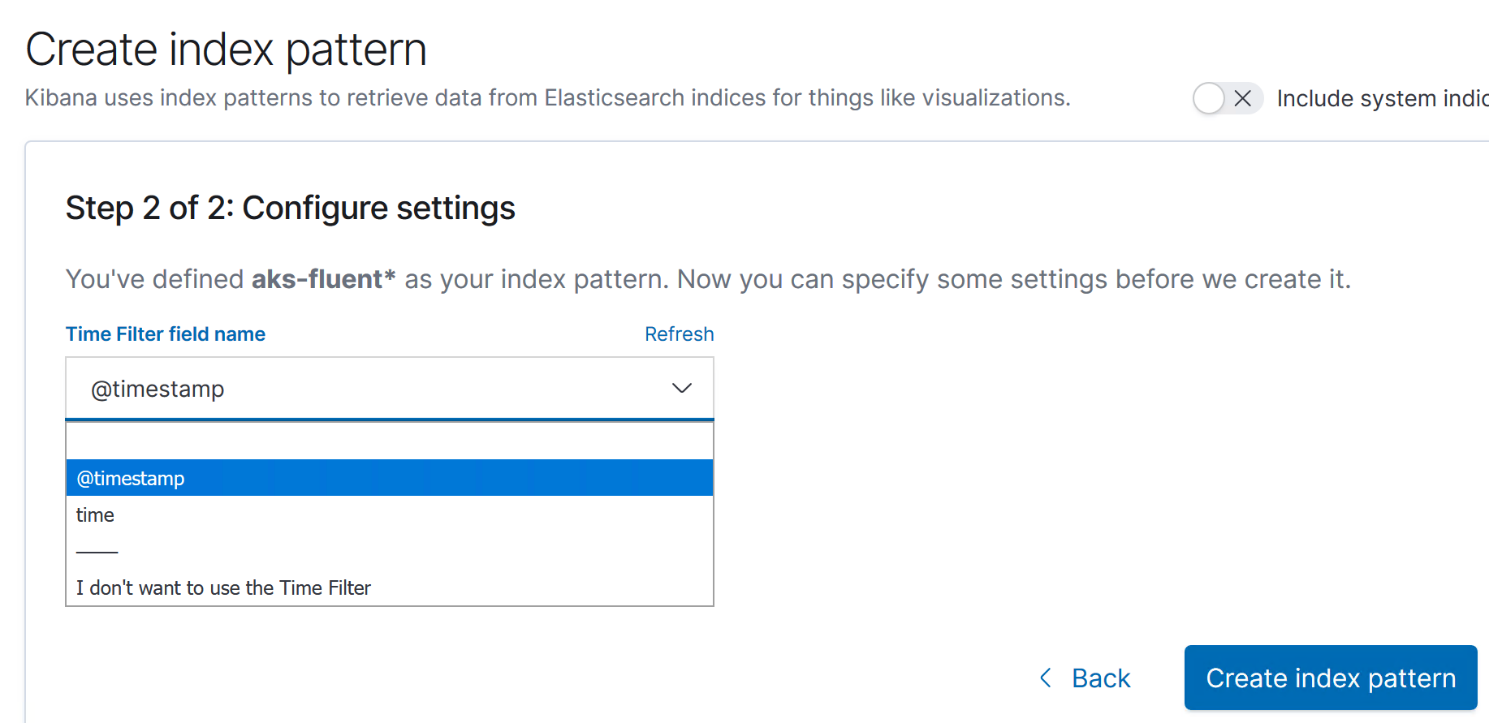
进入Kibana首页面板，选择左侧工具栏最下方Management（齿轮图标）

点击ElasticSearch -> Index Management，应能看到ElasticSearch中已有的索引

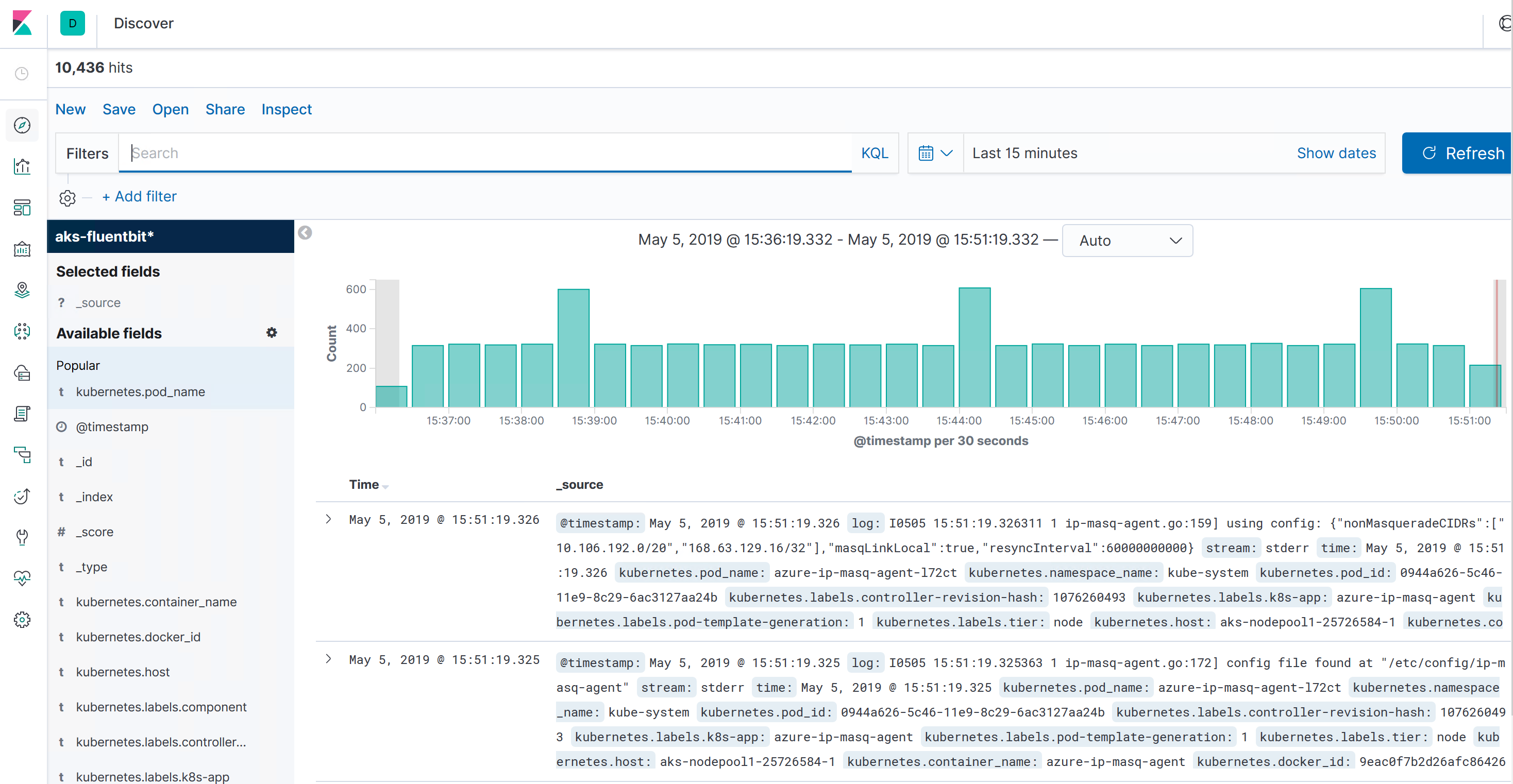
点击Kibana -> Index Patterns -> Create index pattern，创建索引模式，例如：



点击 Next step，进行配置后点击 Create index pattern，例如：



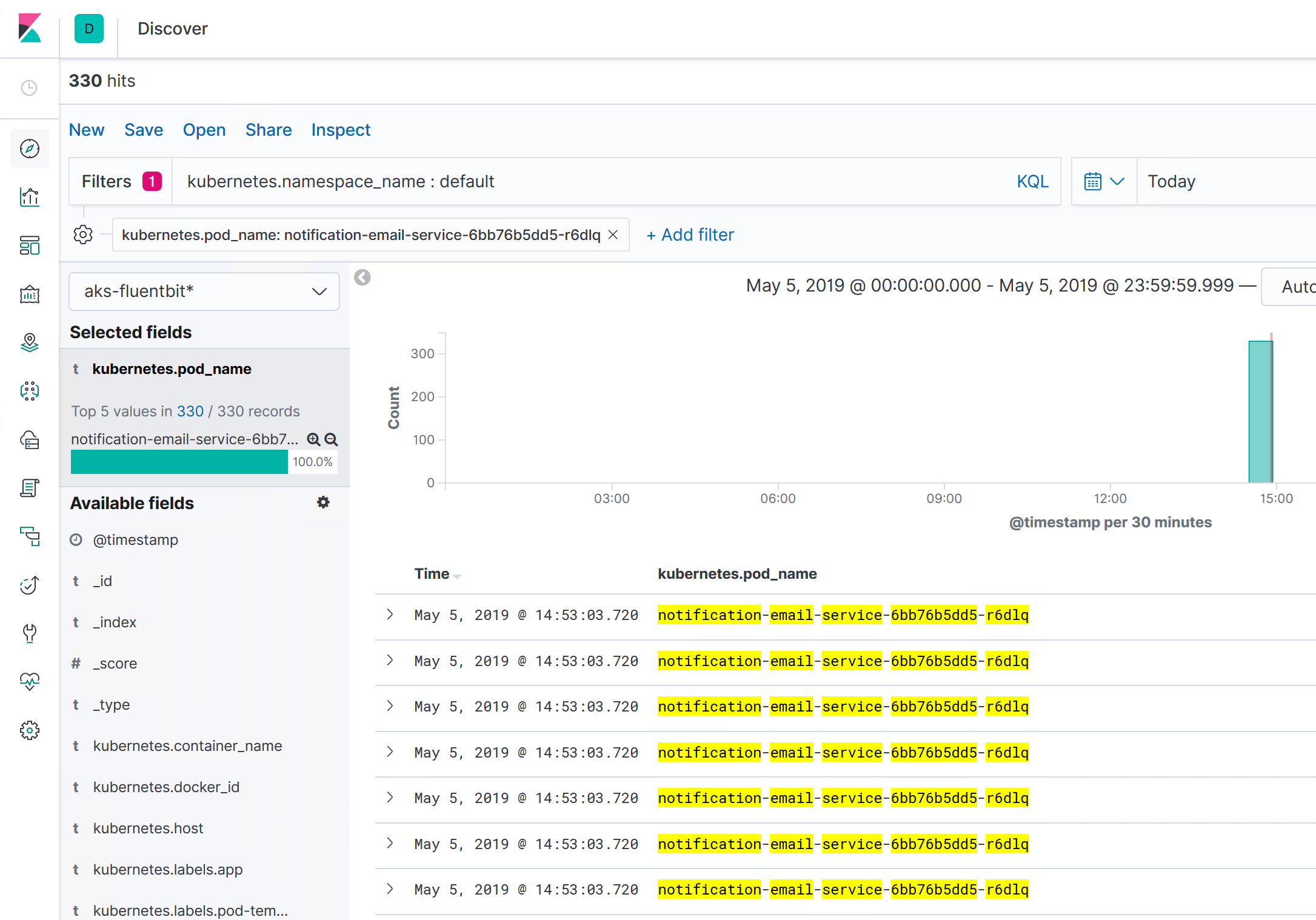
之后，到工具栏左侧选择Discover （最上方图标），选择对应的index pattern，Kibana将自动获取相应的日志记录，例如：



根据查询需要，设置过滤条件，缩小日志查询范围，可以使用的方式有：

* 在Filters中输入查询条件，例如 kubernetes.namespace\_name : default 为查询kubernetes中default namespace中的内容 （请查阅KQL查询语法）
* 在Available fields中移动鼠标到相应fields，点击add，相应的field会被添加到Selected fields，点击此field会显示所有的value，选择相应的取值进行过滤

例如：



过滤条件选择了：

default namespace

Kubernetes pod名字为notification-email-service-xxxxx