1 ES权限管理配置

配置 TLS 和身份验证

第一件事是生成证书，通过这些证书便能允许节点安全地通信。您可以使用企业 CA 来完成这一步骤，但是在此演示中，我们将会使用一个名为 elasticsearch-certutil 的命令

执行 bin/elasticsearch-certutil cert -out config/elastic-certificates.p12 -pass ""

此时config目录下会出现以下文件

elastic-certificates.p12

elasticsearch.keystore

接下来，打开文件 配置config/elasticsearch.yml

注：其他节点需要将同一份elastic-certificates.p12复制到相应位置，然后修改配置

cluster.name: test-7.10.0

node.name: 7.10.0-node-1

path.data: /data1/7.10.0-esdata

path.logs: /app/estest/elasticsearch-7.10.0/logs

network.host: 10.145.134.4

http.port: 6600

transport.tcp.port: 6601

node.ingest: true

search.remote.connect: false

discovery.seed\_hosts: ["10.145.134.4", "10.145.134.5"]

cluster.initial\_master\_nodes: ["7.10.0-node-1", "7.10.0-node-2"]

gateway.recover\_after\_nodes: 3

xpack.monitoring.enabled: true

xpack.monitoring.elasticsearch.collection.enabled: true

xpack.monitoring.collection.enabled: true

xpack.security.enabled: true

xpack.security.transport.ssl.enabled: true

xpack.security.transport.ssl.verification\_mode: certificate

xpack.security.transport.ssl.keystore.path: elastic-certificates.p12

xpack.security.transport.ssl.truststore.path: elastic-certificates.p12

运行命令 bin/elasticsearch。这一可执行文件必须保持es运行，现在可以将此终端放在一边。

X-Pack安全性提供了内置的用户来帮助你启动和运行。elasticsearch-setup-password命令是首次设置内置用户密码的最简单方法。

内置用户有4个，如下：

elastic　　超级用户

Kibana　　用于连接并且和Elasticsearch通信的

logstash\_system 用于在Elasticsearch中存储监控信息

beats\_system 用于在Elasticsearch中存储监控信息

执行命令 bin/elasticsearch-setup-passwords auto。这将会为不同的内部堆栈用户生成随机密码。或者，您也可以跳过 auto 参数，改为通过 interactive 参数手动定义密码。请**记录以下密码**，我们很快就会再次用到这些密码。

（elasticsearch-setup-passwords命令只能运行一次，不能再运行第二次。此后如果你想修改密码的话，可以在Kibana UI界面中通Management --> Users 或者通过API的方式来修改）

Changed password for user apm\_system

PASSWORD apm\_system = [ password ]

Changed password for user kibana\_system

PASSWORD kibana\_system = [ password ]

Changed password for user kibana

PASSWORD kibana = **kibana**

Changed password for user logstash\_system

PASSWORD logstash\_system = [ password ]

Changed password for user beats\_system

PASSWORD beats\_system = [ password ]

Changed password for user remote\_monitoring\_user

PASSWORD remote\_monitoring\_user = [ password ]

Changed password for user elastic

PASSWORD elastic = **elastic**

验证效果使用浏览器访问http://10.145.134.4:6600



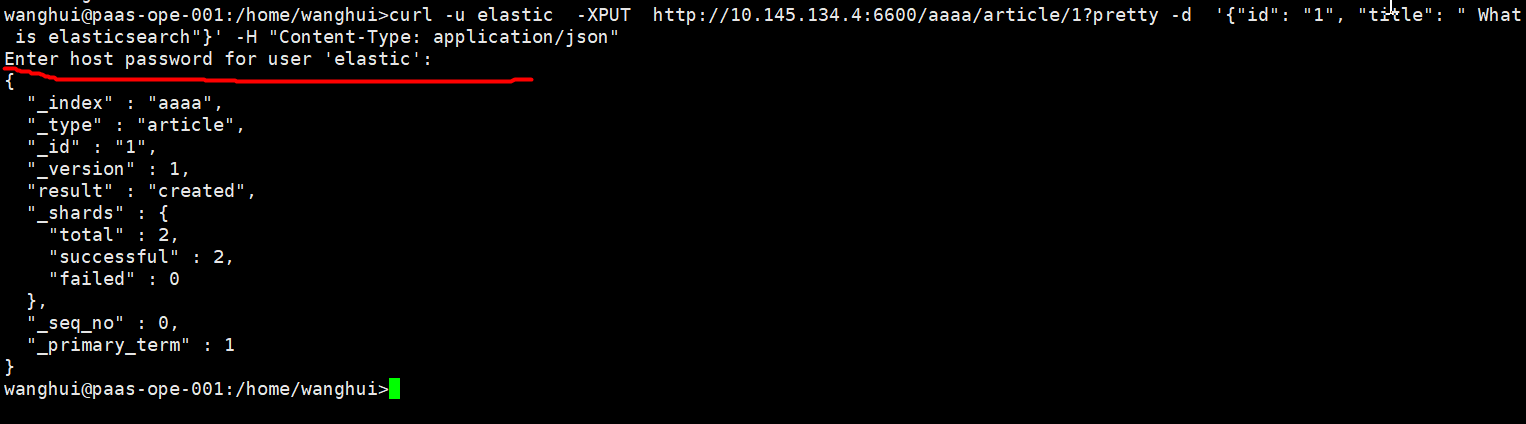
登录超级管理员账号elastic

用户：elastic

密码：elastic (可用自动生成的随机密码）

控制台端，此时可以通过：

curl -u elastic -XPUT http://10.145.134.4:6600/aaaa/article/1?pretty -d '{"id": "1", "title": " What is elasticsearch"}' -H "Content-Type: application/json" 命令来创建索引,此时终端会提示输入密码



2 Kibana 配置

#vim kibana.yml

server.port: 5710

server.host: "10.145.134.2"

elasticsearch.hosts: ["http://10.145.134.4:6600"]

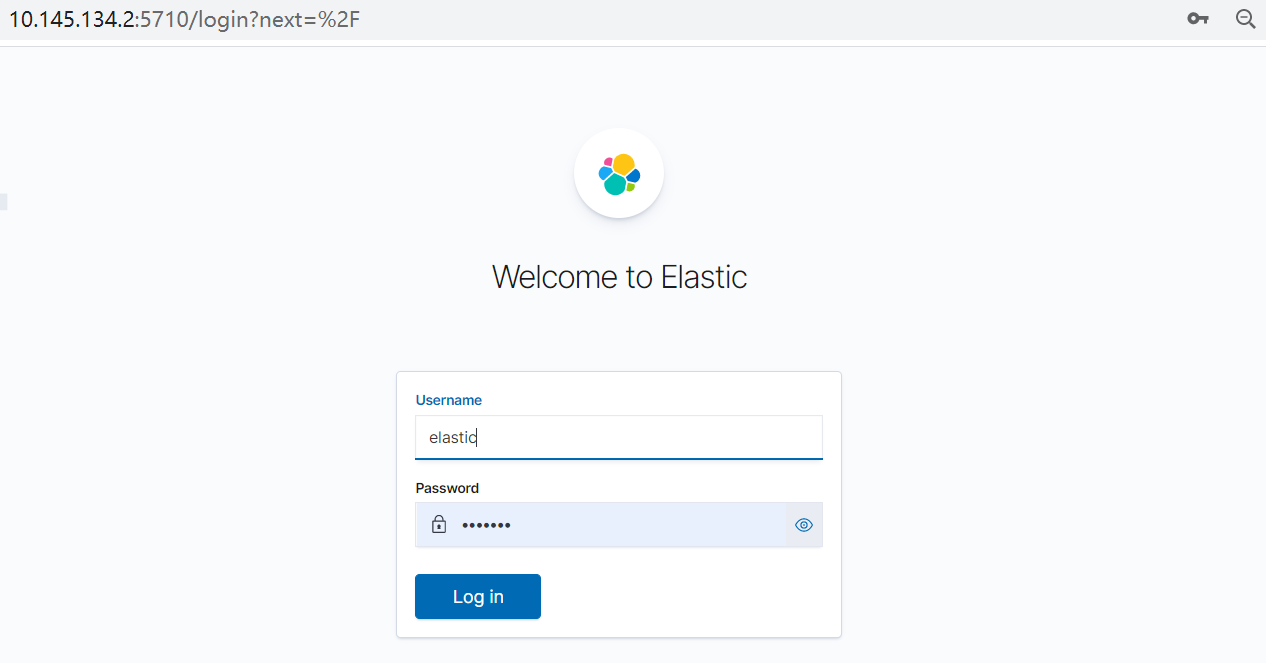
elasticsearch.username: "elastic"

elasticsearch.password: "elastic"

启动Kibana ：

nohup ./bin/kibana &

使用浏览器访问 http://10.145.134.2:5710

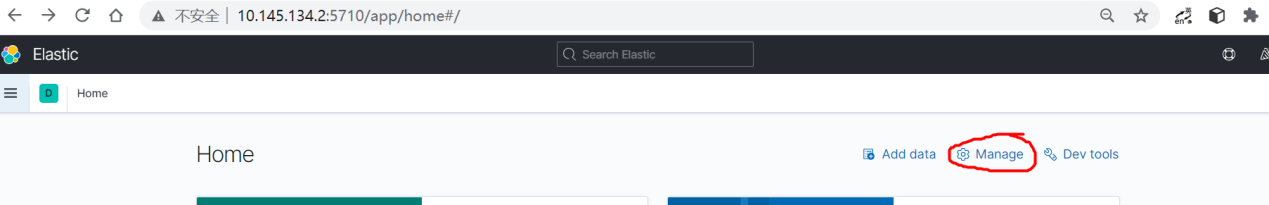


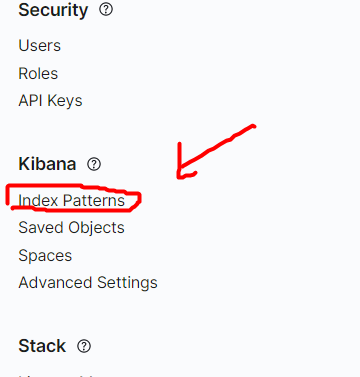
登录之前es创建的elastic用户（elastic是超级用户创建Role、user需要用elastic）

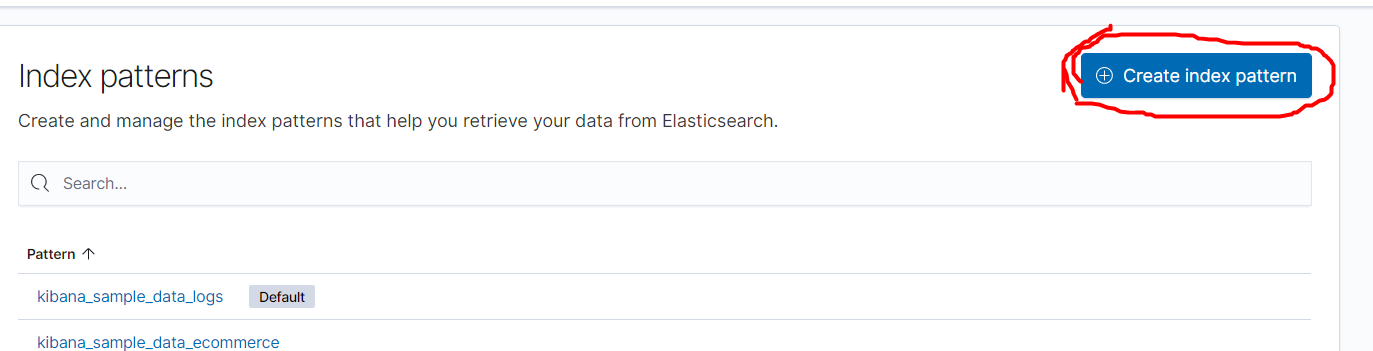
登陆后创建索引并进行user、Role的创建（按照顺序）

用户的权限划分： 创建Role,赋予权限，指定索引

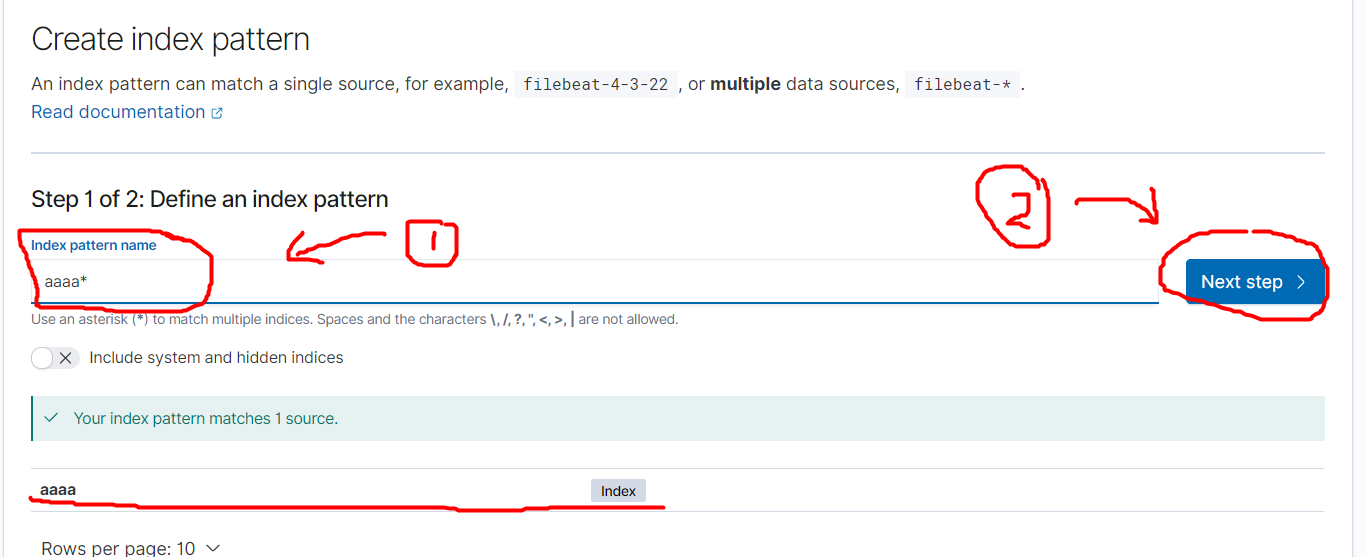
创建uesr,设置密码，赋予Role





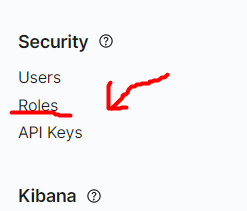


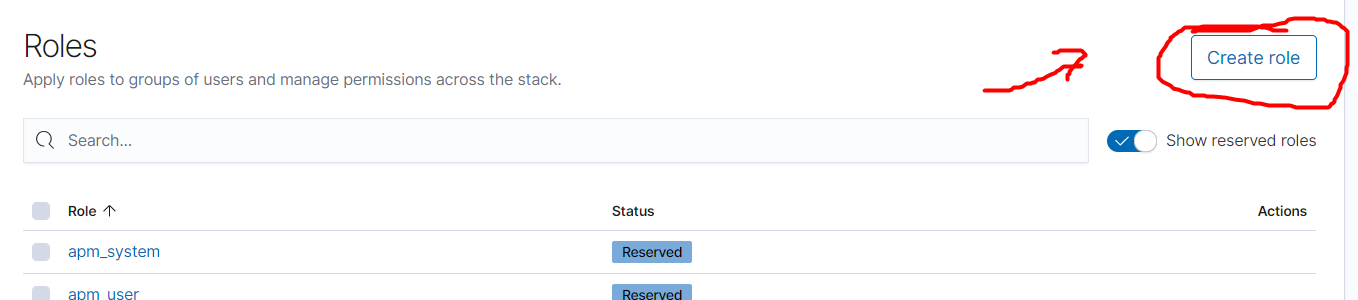
##搜到上文创建的aaaa索引





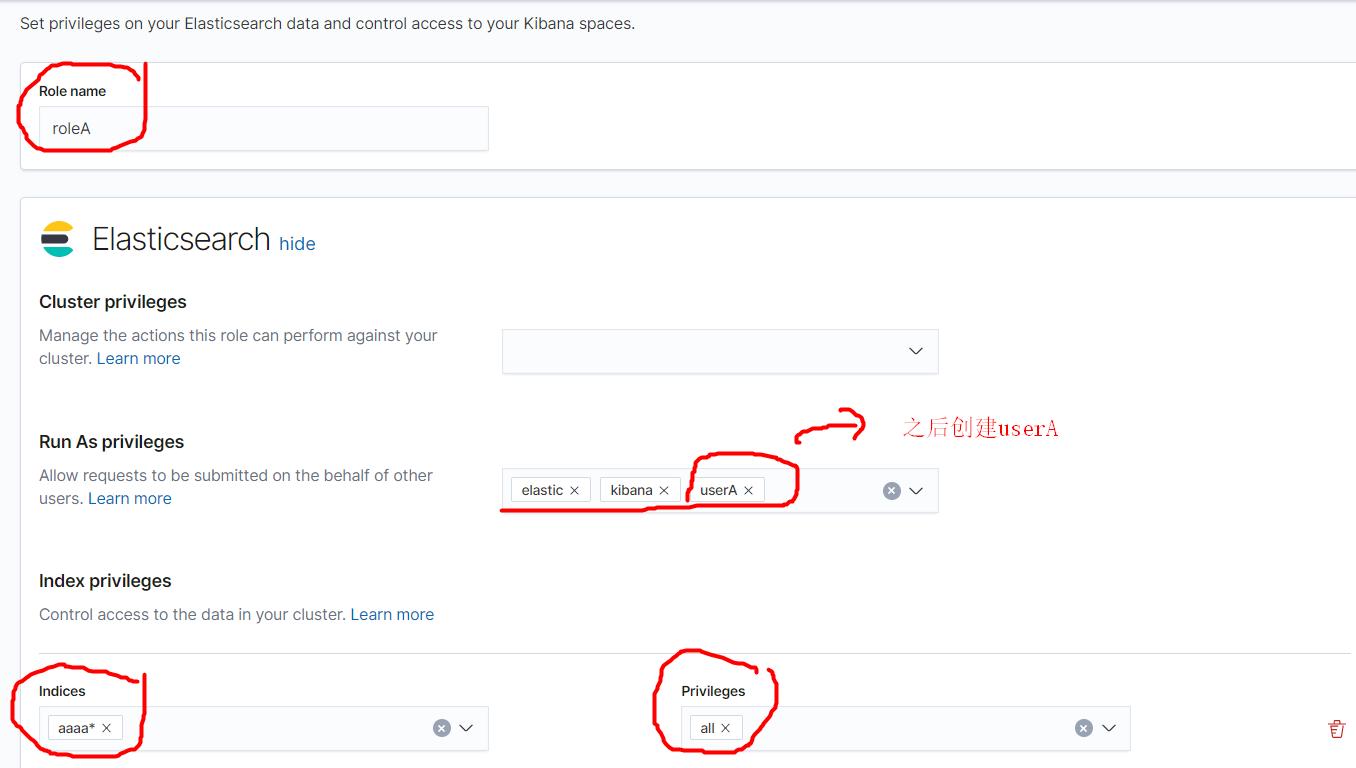
##创建Role,赋予权限，指定索引



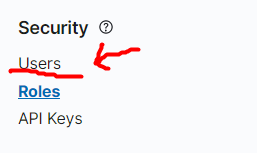


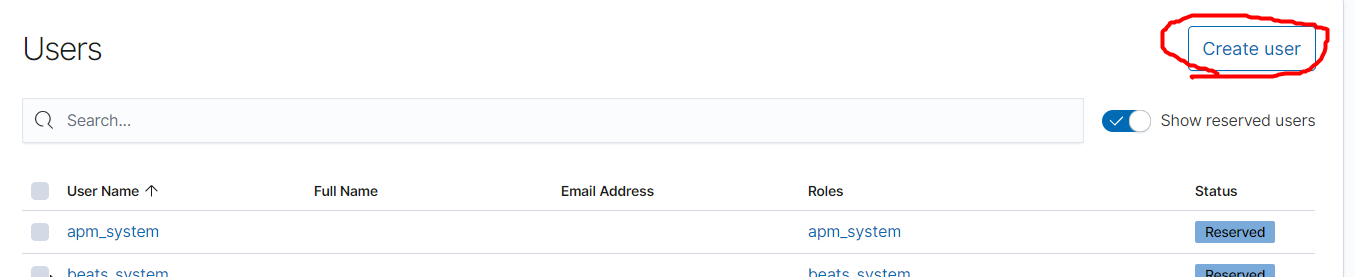
* 1. 设置用户只有权操作部分特定索引

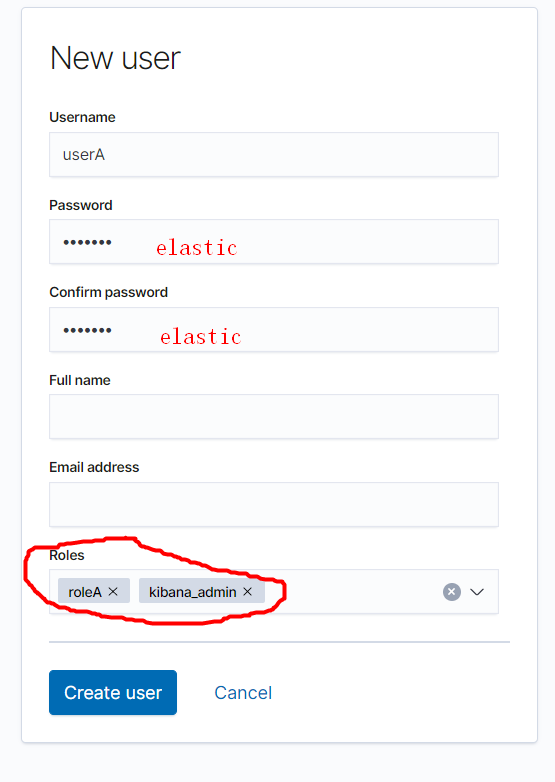
设置roleA只有权限操作aaaa\* 索引



##创建uesr,设置密码，赋予Role

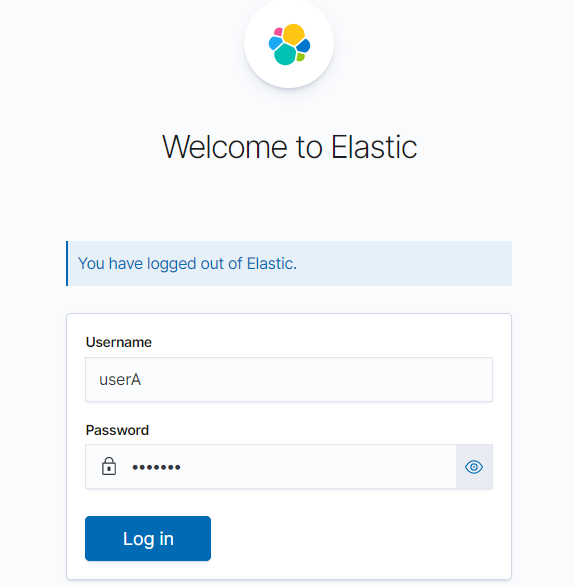




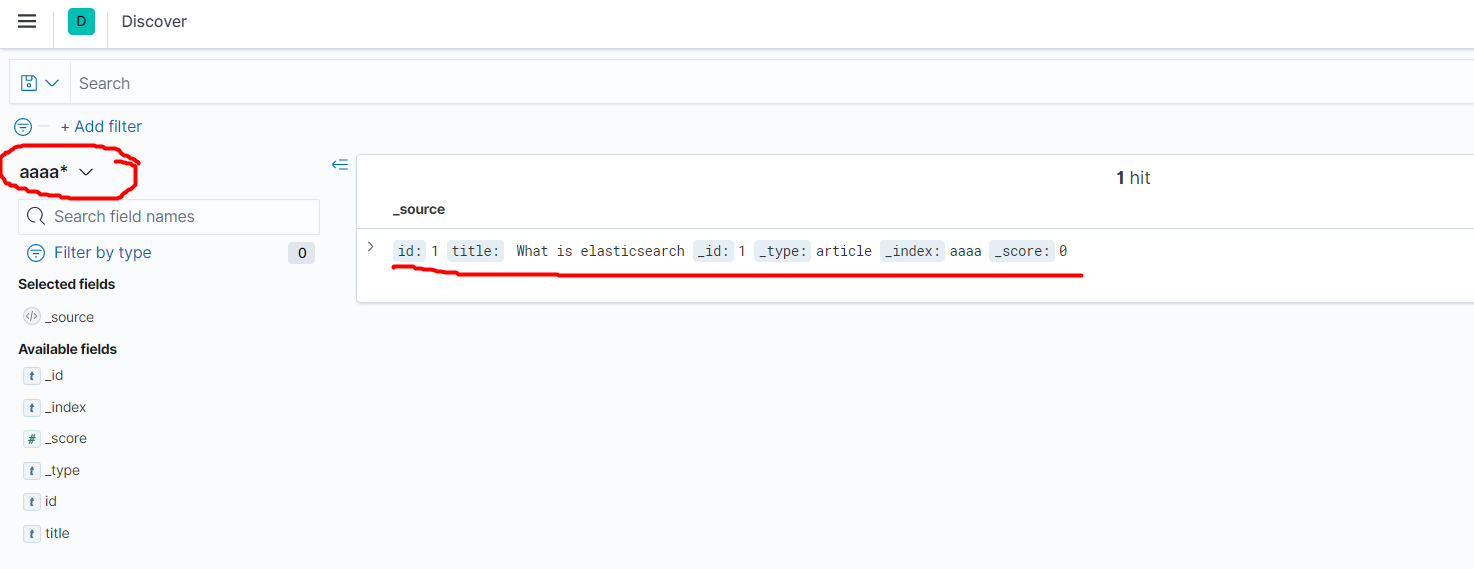


验证权限设置：

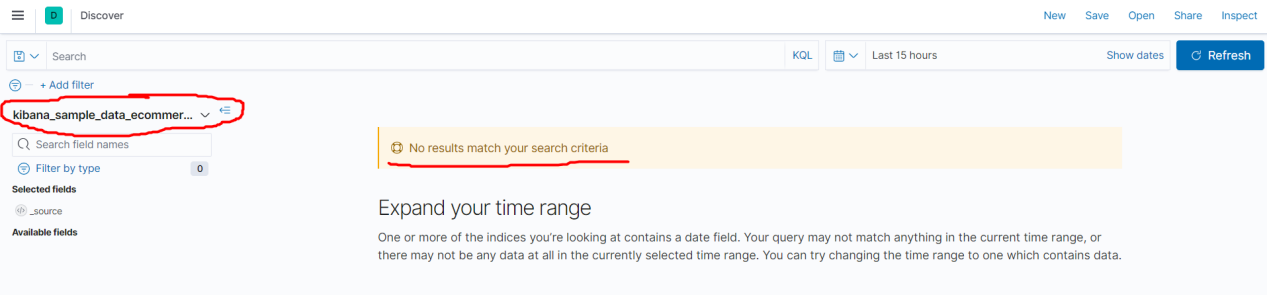
**用新创建的 设置了用户权限的userA登录**



##对于已经赋权的aaaa\*索引能够访问和搜索。

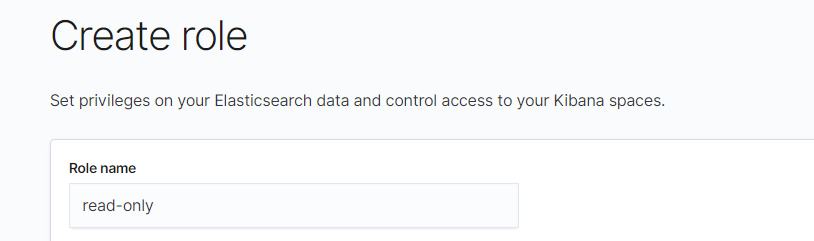


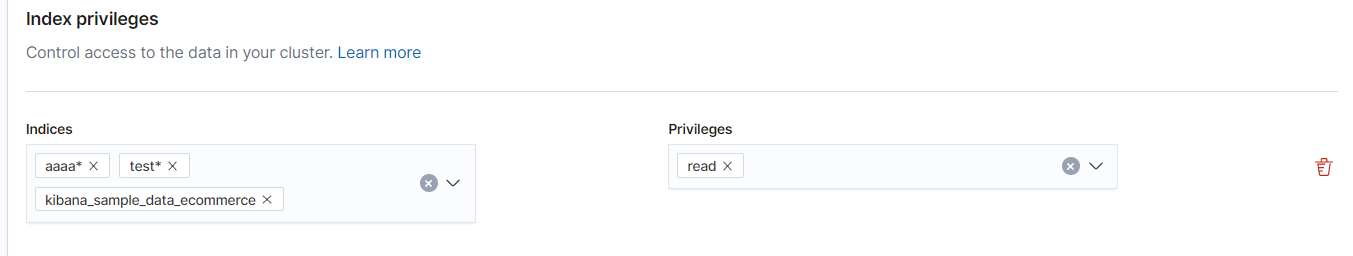
##对于其他未赋权的索引 则无法访问和搜索数据



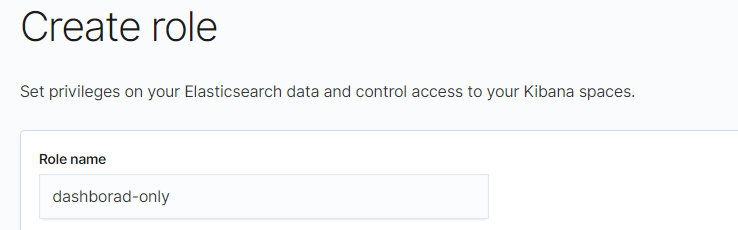
2.2 设置用户只有浏览图表的权限

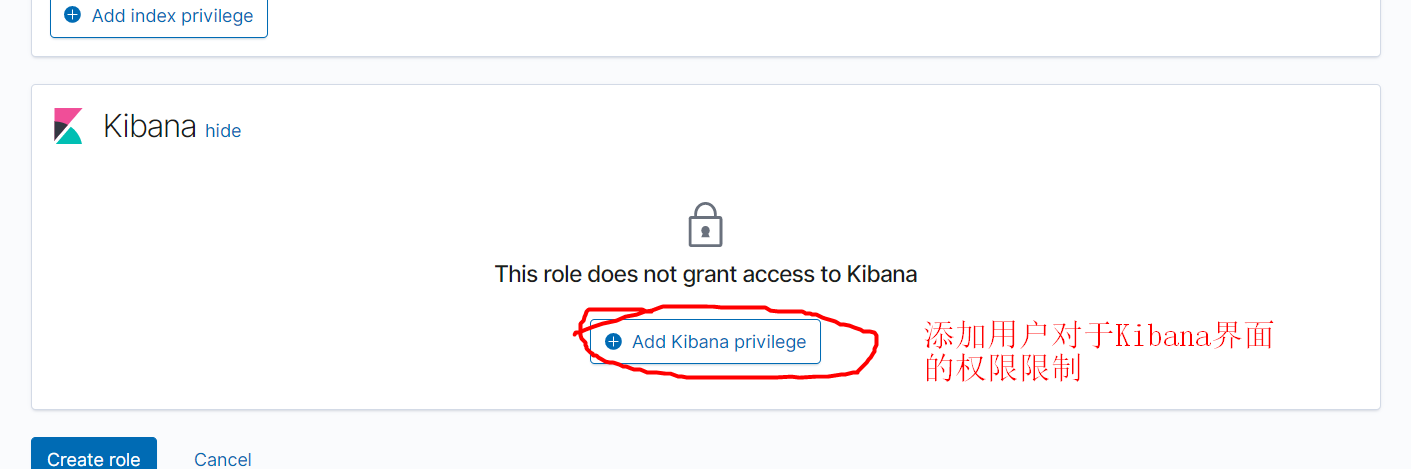
##创建表示只读权限的角色read-only

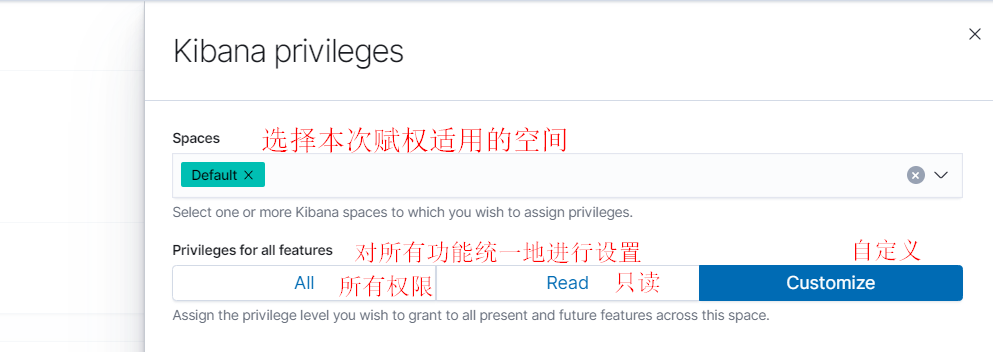


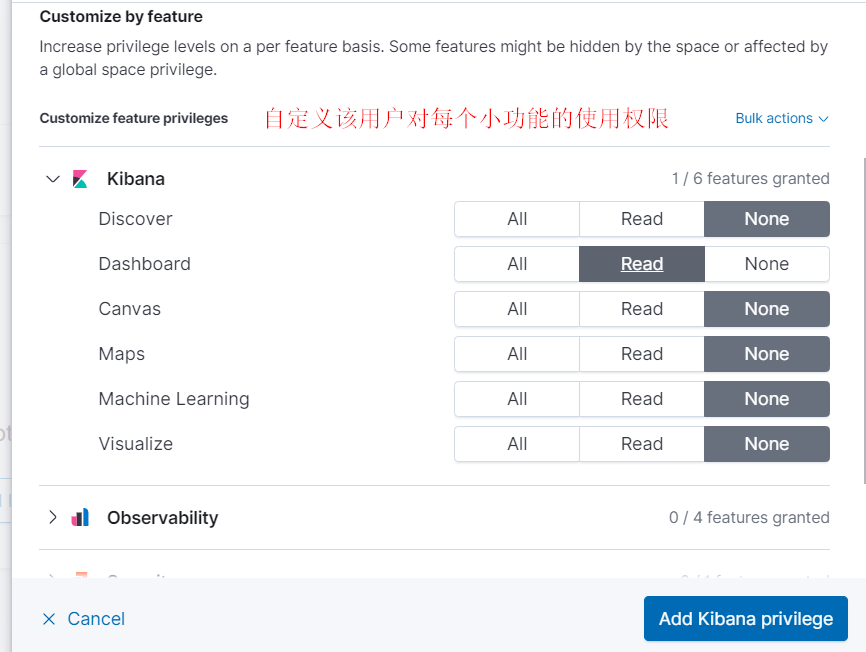


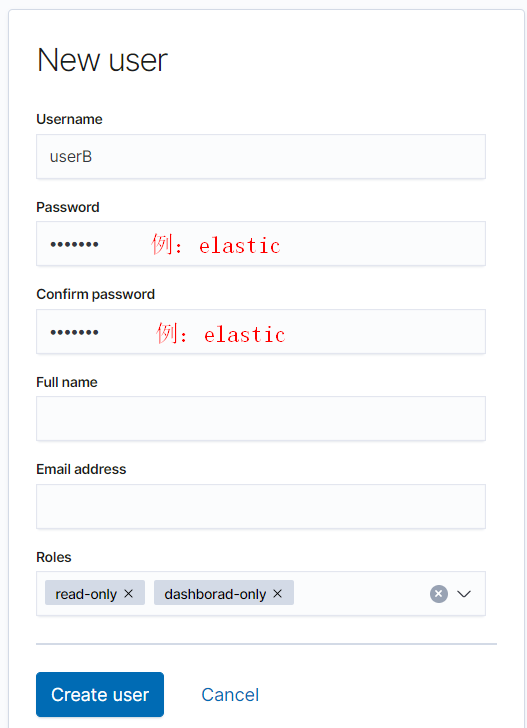
##创建表示权限只能看dashboard的角色dashboard-only



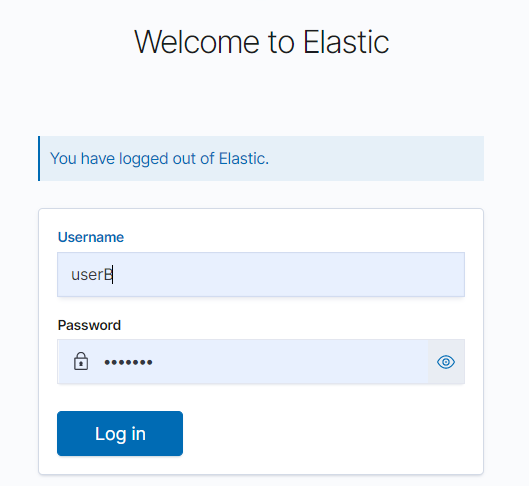




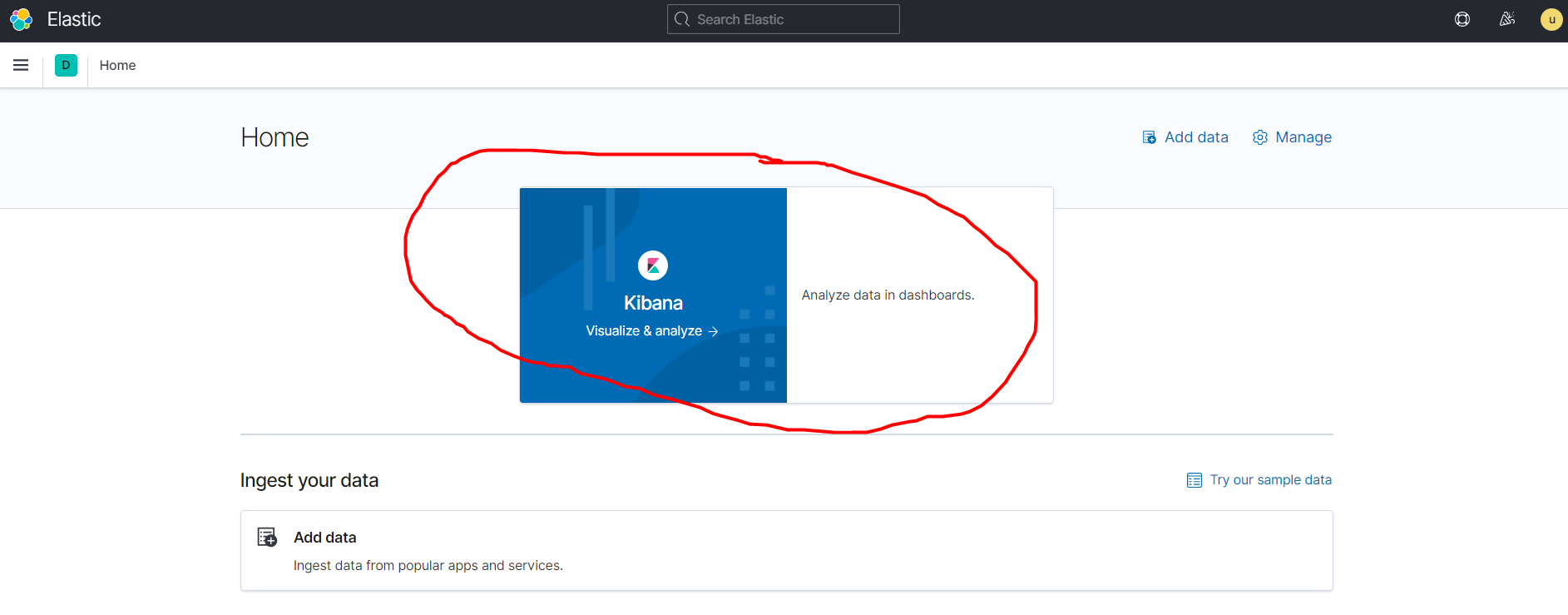




#用刚创的userB登录kibana

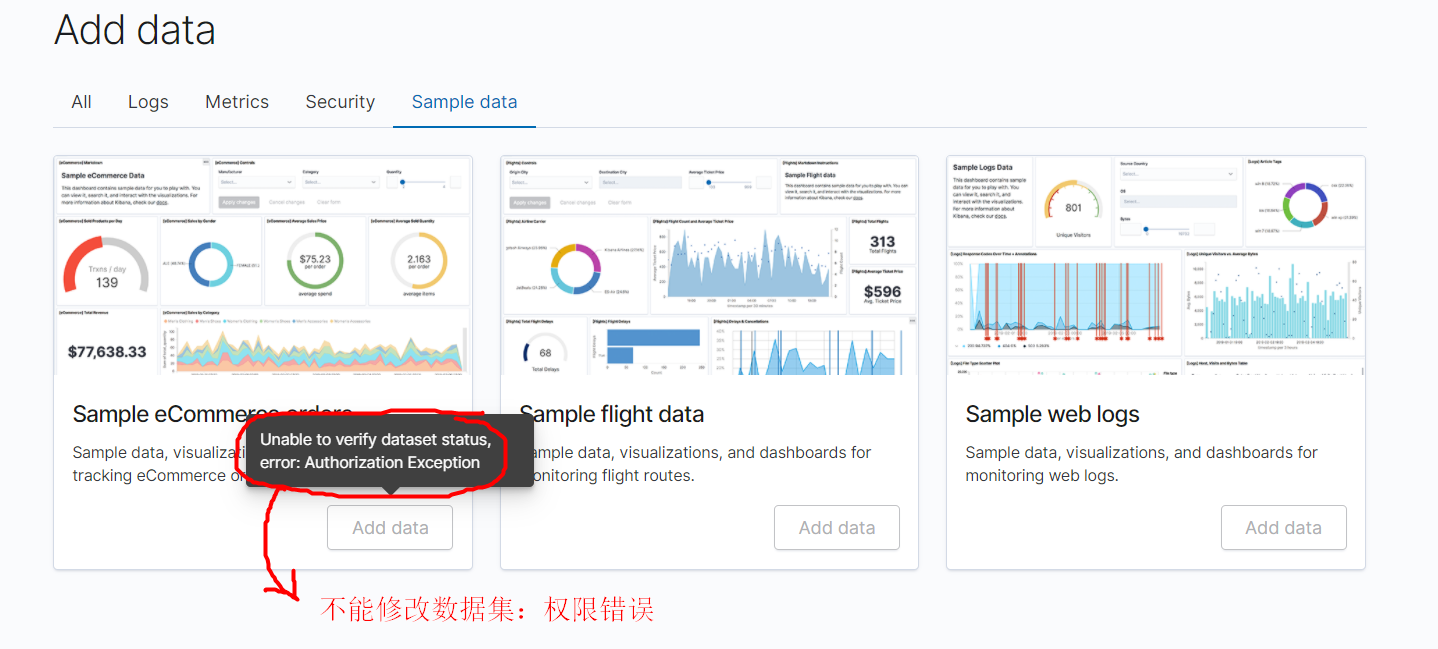


#可以看到页面中只有正中间一个dashboard可以看了



#除了浏览dashboard不能做其他操作，包括添加默认的示例数据

#未被授权的行为会报错：权限错误

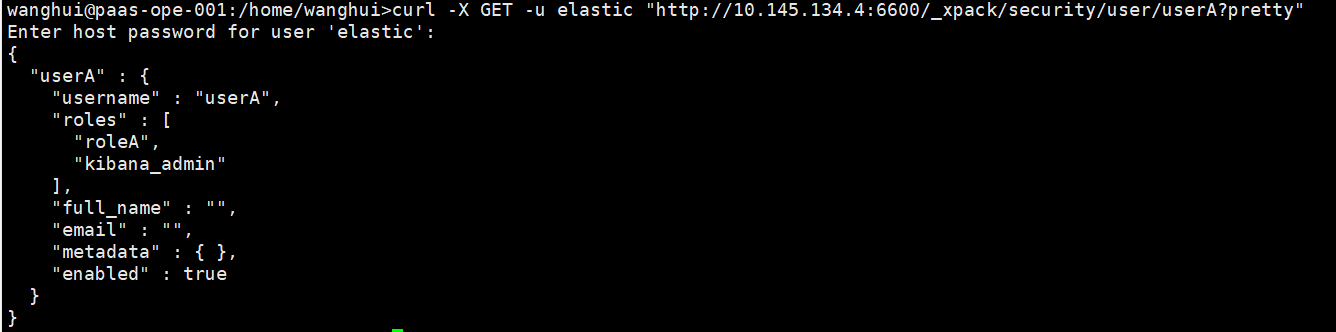


3 用户管理API操作

##用户管理

#查看指定用户的信息

curl -X GET -u elastic "http://10.145.134.4:6600/\_xpack/security/user/userA"



#修改用户密码

curl -X POST -u elastic "http://10.145.134.4:6600/\_xpack/security/user/userA/\_password" -H 'Content-Type: application/json' -d'

{

"password" : "elastic"

}

#禁用用户

curl -X PUT -u elastic "http://10.145.134.4:6600/\_xpack/security/user/userA/\_disable"

#启用用户

curl -X PUT -u elastic "http://10.145.134.4:6600/\_xpack/security/user/userA/\_enable"

#删除用户

curl -X DELETE -u elastic "http://10.145.134.4:6600/\_xpack/security/user/userA"

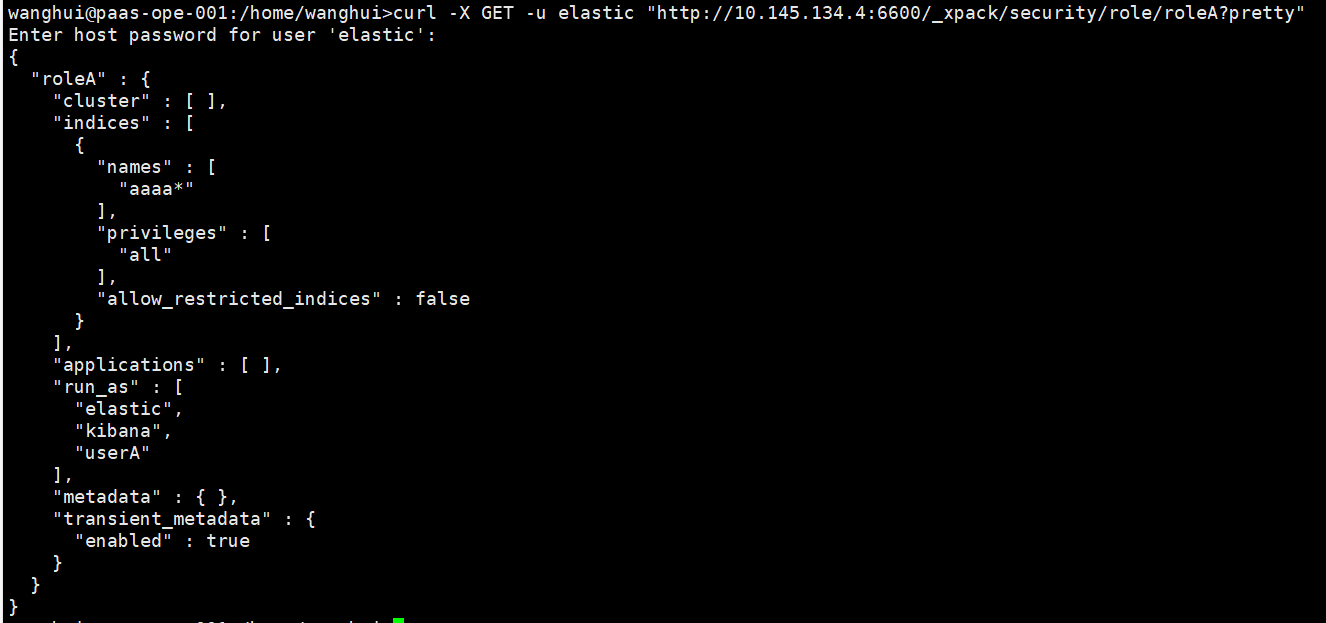
##角色管理

#查看所有角色的信息

curl -X GET -u elastic "http://10.145.134.4:6600/\_xpack/security/role?pretty"

#查看角色信息

curl -X GET -u elastic "http://10.145.134.4:6600/\_xpack/security/role/my\_admin\_role?pretty"



#删除角色

curl -X DELETE -u elastic "http://10.145.134.4:6600/\_xpack/security/role/my\_admin\_role"

#创建角色

curl -X POST -u elastic "http://10.145.134.4:6600/\_xpack/security/role/my\_admin\_role" -H 'Content-Type: application/json' -d'

{

"cluster": ["all"],

"indices": [

{

"names": [ "索引名" ],

"privileges": ["权限名"],

"field\_security" : { #可选

"grant" : [ "field名" ]

},

"query": "{\"match\": {\"title\": \"foo\"}}" #可选

}

],

"run\_as": [ "other\_user" ], #可选

"metadata" : { #可选

"version" : 1

}

}