文档标题：Di-Logger常见问题

创建时间：2023-4-21

撰写人：何亮亮、周佳佳、李宝平

目录

[1.安装升级常见问题 3](#_Toc133338431)

[2.数据收集常见问题 3](#_Toc133338432)

[3.自定义菜单常见问题 7](#_Toc133338433)

[4.登录相关常见问题 11](#_Toc133338434)

[5.权限相关常见问题 12](#_Toc133338435)

[6.自监控相关问题 12](#_Toc133338436)

[7.搜索分析相关问题 13](#_Toc133338437)

[8.仪表板相关问题 14](#_Toc133338438)

[9.邮件服务器相关问题 15](#_Toc133338439)

[10.存储使用相关问题 15](#_Toc133338440)

[11.通用配置 16](#_Toc133338441)

[12.离线报告相关 26](#_Toc133338442)

# 1.安装升级常见问题

1. **Q：如何控制Di-Logger内存使用。**

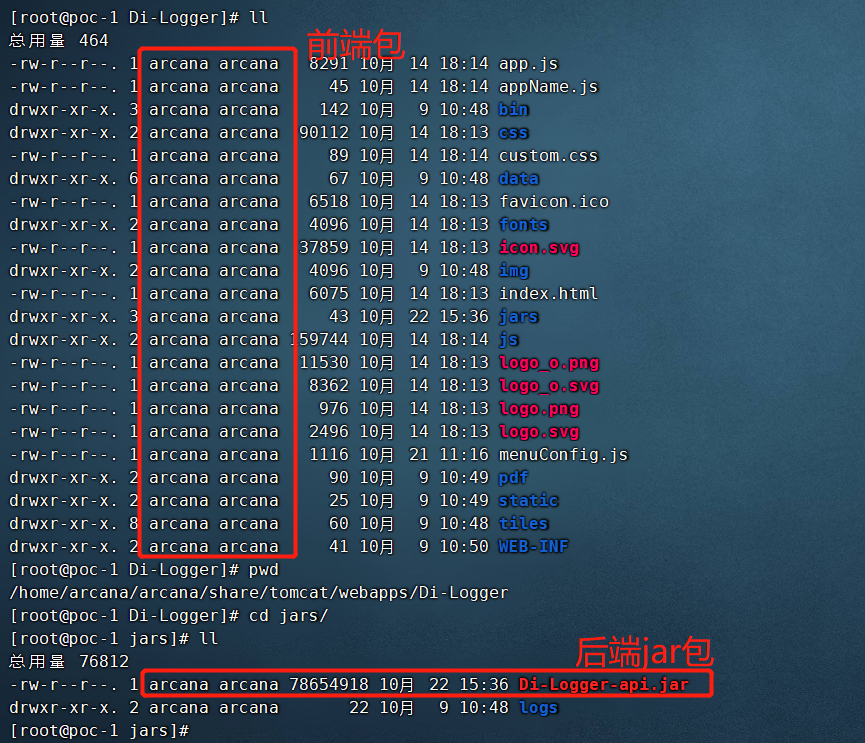
A：Di-Logger安装目录下app.js中修改参数配置： "executeEnv": "-Dserver.port=7082 **-Xms1024m -Xmx1024m**"。

以上为Di-Logger默认配置，若部署环境高于16G可以将1024配置为更高参数值，如2048。

1. **Q：升级后登录Di-Logger会显示空白页面。**

A：一般由于Di-Logger启动失败引起。可以首先排查是否因root权限启动导致，操作如下：

1. 在Di-Logger安装路径下查看权限：



1. 若root权限执行chown -R arcana:arcana /arcana/share/tomcat/webapps/Di-Logger/。

若非权限问题，请联系技术支持人员。

# 2.数据收集常见问题

1. **Q：数据收集只进入Edge，但未进入ARCANA平台，无法进行搜索。**

A：当ARCANA分析节点和Edge节点部署在不同服务器上，需要进入/opt/arcana/conf修改两个文件的配置：

1. server.properties

kafkaServer = *ip*:9092

1. kafka.properties

advertised.listeners=PLAINTEXT://*ip*:9092

*ip*，为arcana分析节点的IP地址。

1. **Q：怎么实现秒级的收数？**

A：需要手动更改后端的logstashID.conf配置

编辑$ARCANA\_HOME/etc/logstash/logstashID.conf(617fa6e87afb8e2f68f6c07a.conf)文件中的schedule参数：

schedule = "\*/15 \* \* \* \* \*" # 每15秒收一次数据

1. **Q：Logstash数据如何转发到外部UDP/TCP端口？**

A：更改配置文件/opt/arcana-edge/etc/logstash/下的两个配置文件kafka.conf和output.conf。**kafka.conf** input{ kafka{ bootstrap\_servers => ["localhost:9093"] group\_id => "arcana" client\_id => "test" auto\_offset\_reset => "latest" consumer\_threads => 1 decorate\_events => false topics=>["input-topic-60bdefb52abc8d09ab1fe458"] } } filter{ mutate{ rename => {"@timestamp" => "timestamp"} remove\_field => ["@version"] add\_field => { "processor" => "LogstashJSON"} add\_field => { "source" => "java"} add\_field => { "index" => "tcp"} } } output { kafka { bootstrap\_servers => "localhost:9092" codec => "json" topic\_id => "forward-topic" } stdout{ codec => rubydebug } }**output.conf** input{ kafka{ bootstrap\_servers => ["localhost:9093"] group\_id => "arcana" client\_id => "test" auto\_offset\_reset => "latest" consumer\_threads => 1 decorate\_events => false topics=>["input-topic-60bdefb52abc8d09ab1fe458"] } } filter{ mutate{ rename => {"@timestamp" => "timestamp"} remove\_field => ["@version"] add\_field => { "processor" => "LogstashJSON"} add\_field => { "source" => "java"} add\_field => { "index" => "tcp"} } } output { kafka { bootstrap\_servers => "localhost:9092" codec => "json" topic\_id => "forward-topic" }stdout{ codec => rubydebug }}

1. **Q：Filebeat如何配置udp/tcp转发到edge?**

/opt/filebeat-7.11.2-linux-x86\_64/filebeat.ymlfilebeat.inputs: - type: log enabled: true encoding: utf-8 #tags: ["log"] exclude\_files: ['.gz$'] paths: - /opt/ftp\_files/ali\_file/\*/\*.log - /opt/ftp\_files/huawei/yun\_cloud/logs/broker\*.log\* output.logstash: # The Logstash hosts hosts: ["localhost:5044"] 启动 ./filebeat -e -c filebeat.yml -d "publish" 或nohup ./bin/filebeat -f config/test.conf >>/FILEBEAT\_HOME/logs/filebeat.log &

# 3.自定义菜单常见问题

1. **Q：如何配置自定义菜单。**

A：请见《用户手册》附录。

1. **Q：自定义菜单配置后，点击菜单，出现登录页面。**

A：登录使用的是域名，但menuConfig.js中的URL配置的是IP，或者登录使用的是IP，但URL配置的是域名导致的。建议自定义菜单配置URL不用加上IP或域名，直接配置成“/dashboard/manage/preview/xxxxxx”

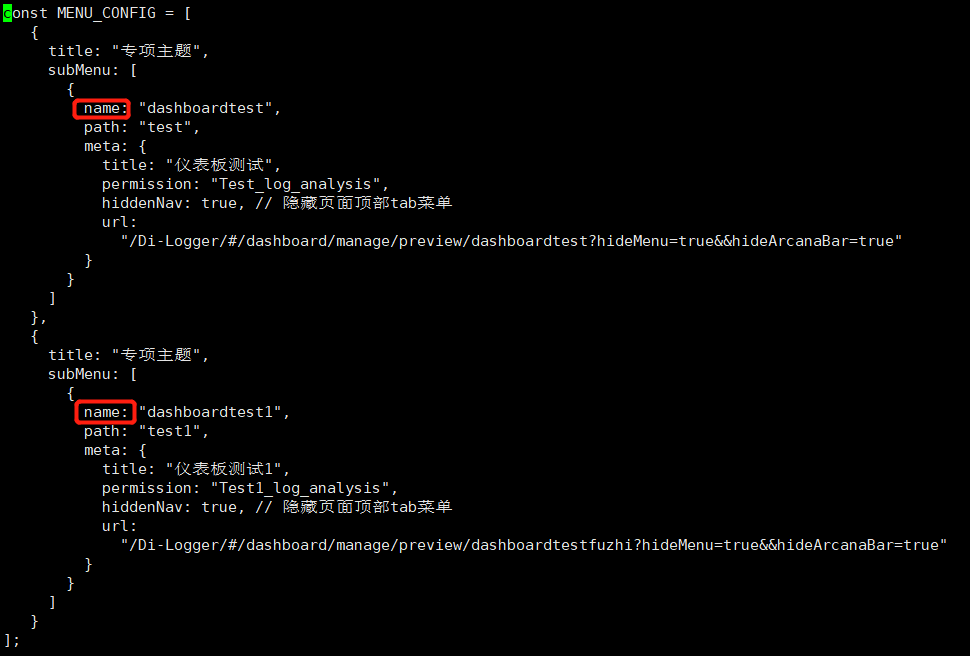
1. **Q：按步骤配置自定义菜单，reload Di-Logger后，自定义菜单未显示。**

A：首先检查配置文件是否正确，通常遇到的错误有：

* 逗号输入法；
* permission错误，若自定义菜单无需权限管控，默认为no\_need\_authority；
* 一级菜单title设置为非系统自带菜单名称；
* 若以上都检查无误，可清空浏览器缓存，再登录查看。

1. **Q：若同一个一级菜单下配置多个自定义菜单，点击切换，无法正常切换。**

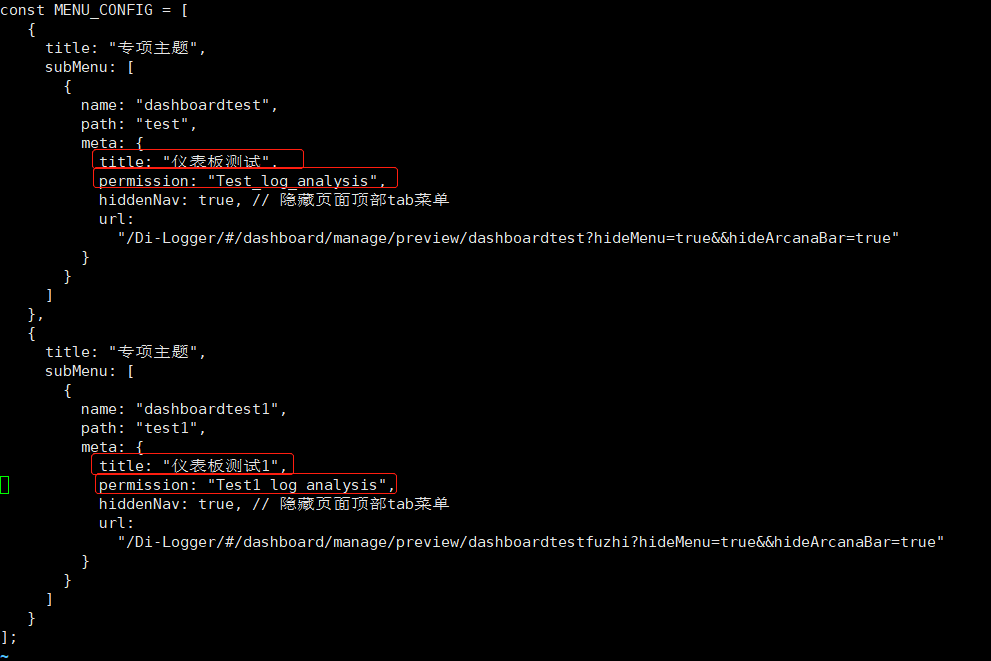
A：menuConfig.js中的name需要设置不同。



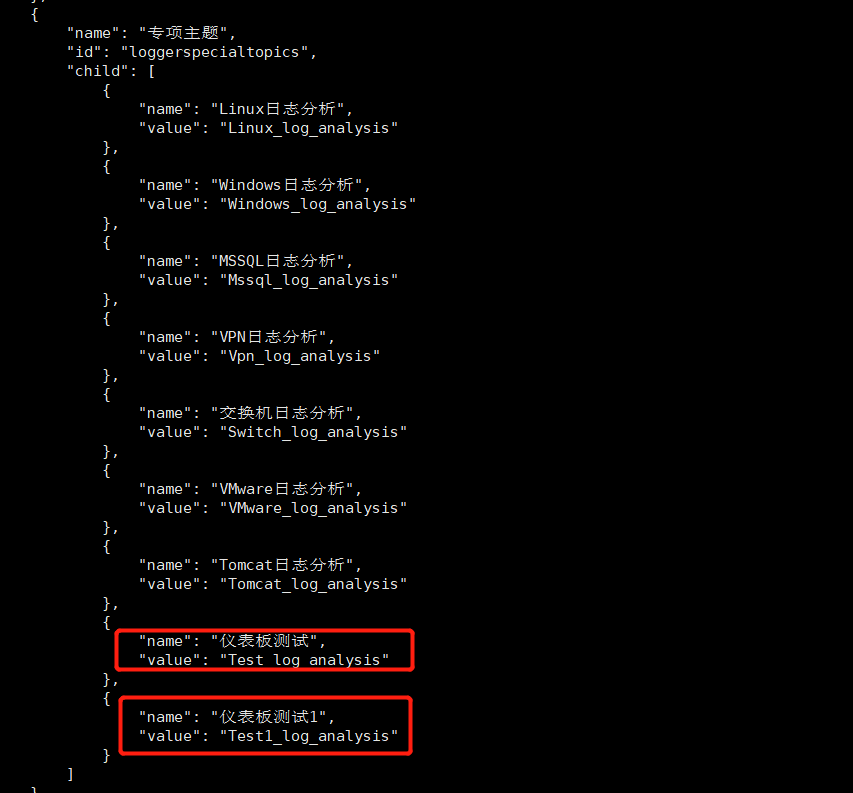
1. **Q：若将用户组某一级菜单下的二级菜单都隐藏，该一级菜单也会隐藏，导致一级菜单下的自定义菜单也无法显示。**

A：默认自定义菜单不受用户组权限管控，可以通过配置增加自定义菜单受用户组权限管控。以下以专项主题下增加两个自定义菜单为例，配置步骤如下：

1. 配置menuConfig.js文件，设置自定义二级菜单。

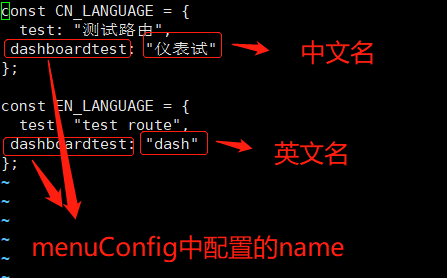


1. 配置app.js文件。配置自定义菜单受用户组管控。需要注意app.js中的name与menuConfig.js中的title一致，app.js中的value与menuConfig.js中的permission一致。app.js中的name和value请保证唯一性



1. 进入/arcana/bin路径，执行./arcana restart重启arcana。
2. 通过用户组权限控制自定义菜单展示。
3. **Q：点击自定义菜单后，菜单标题出现“menus.”的前缀。**

A：此问题是因为中英文翻译导致，需要在/arcana/share/tomcat/webapps/Di-Logger/menusLanguage.js配置中英文菜单名称。



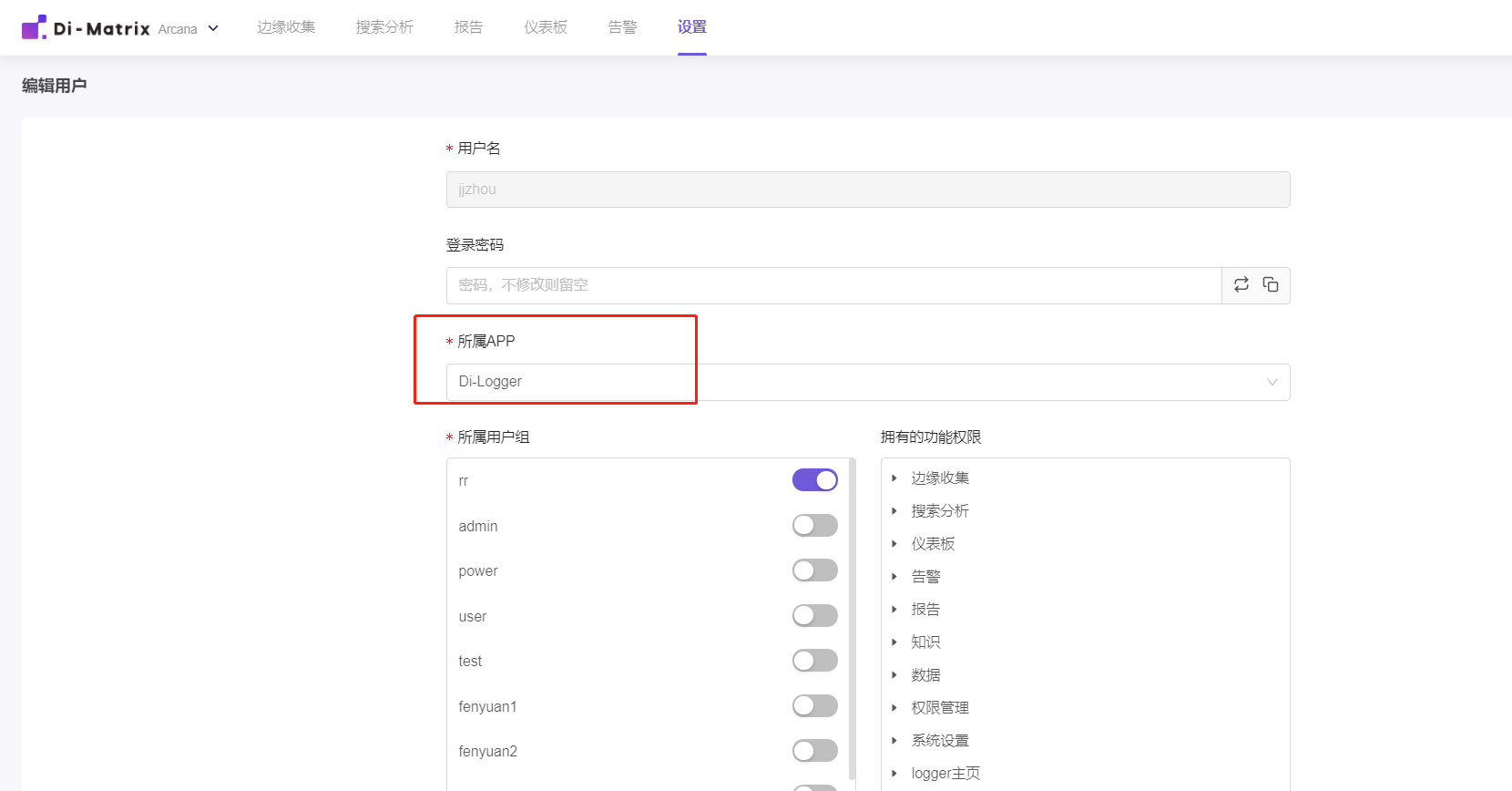
**7、Q：配置某dashboard作为自定义二级菜单后，点击自定义菜单后，提示“Error 404”。**

**A：**Di-Logger V3.0.2209及以上版本配置自定义菜单，其menuConfig.js中的URL配置需要加上“/Di-Logger/#/”前缀，例如“/Di-Logger/#/dashboard/manage/preview/test?hideMenu=true”。

# 4.登录相关常见问题

1. **Q：用户登录后如何直接进入Di-Logger首页。**

A：在设置>用户页面，新建用户或者编辑该用户时，配置所属APP为Di-Logger即可。



1. **Q：为什么登录页面有时是英文有时是中文。**

A：默认首页登录是中文。当登录成功后，在页面上切换为英文，退出登录后，登录页面会显示英文。若清空了浏览器缓存则登录页面会恢复成中文。

# 5.权限相关常见问题

1. **Q：配置Di-Logger权限后，提示“Error 401，对不起~你没有权限访问该页面，你可以点击按钮返回首页”。**

A：在配置Di-Logger的权限时，Di-Logger V2.0版本和Di-Logger V 2.1版本的首页权限必须勾选。

# 6.自监控相关问题

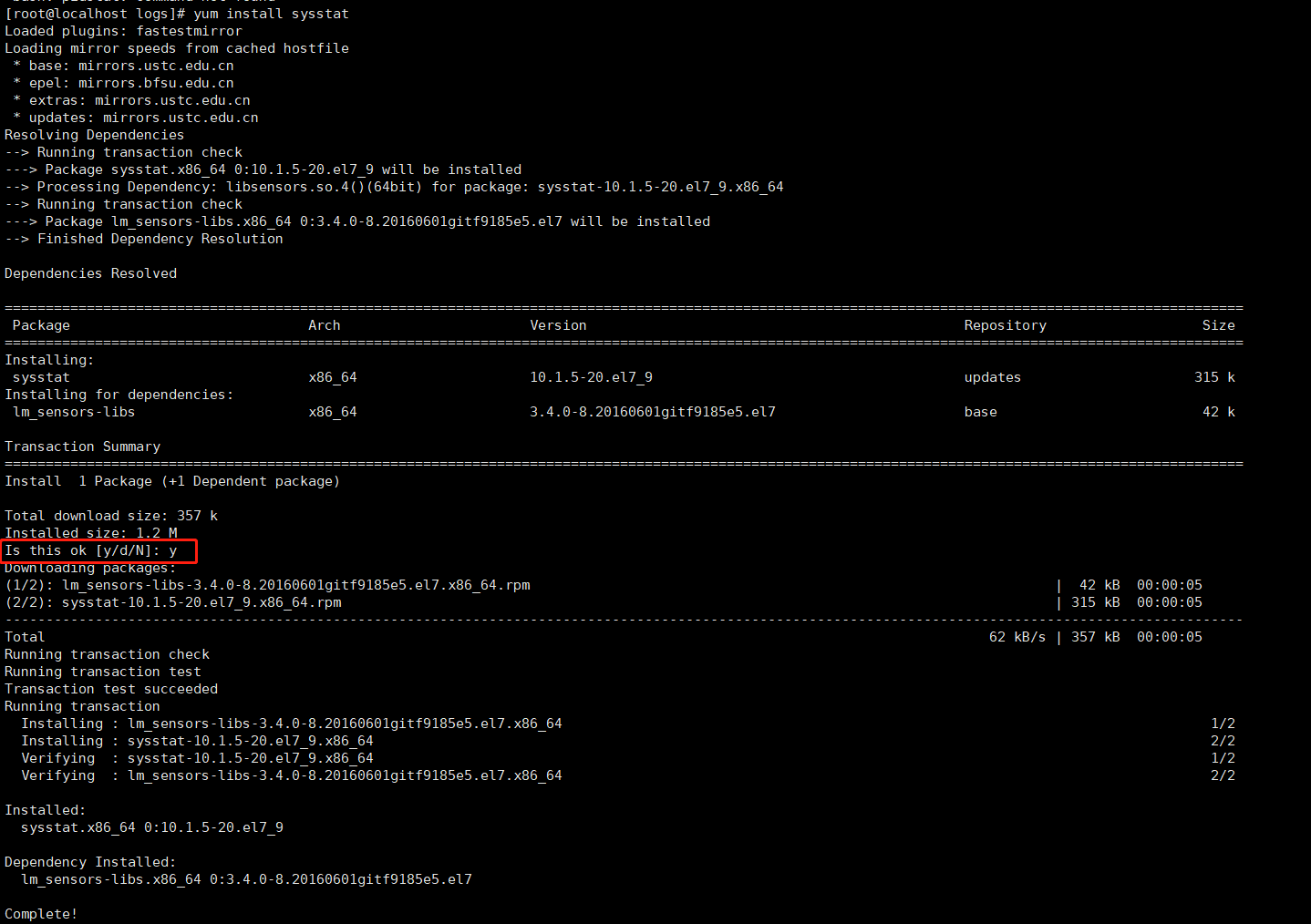
**1、Q：若组件自监控或服务器自监控出现面板提示“查询错误!”。**

A：

1. 在后台任意路径下输入pidstat，若提示如下操作2)。

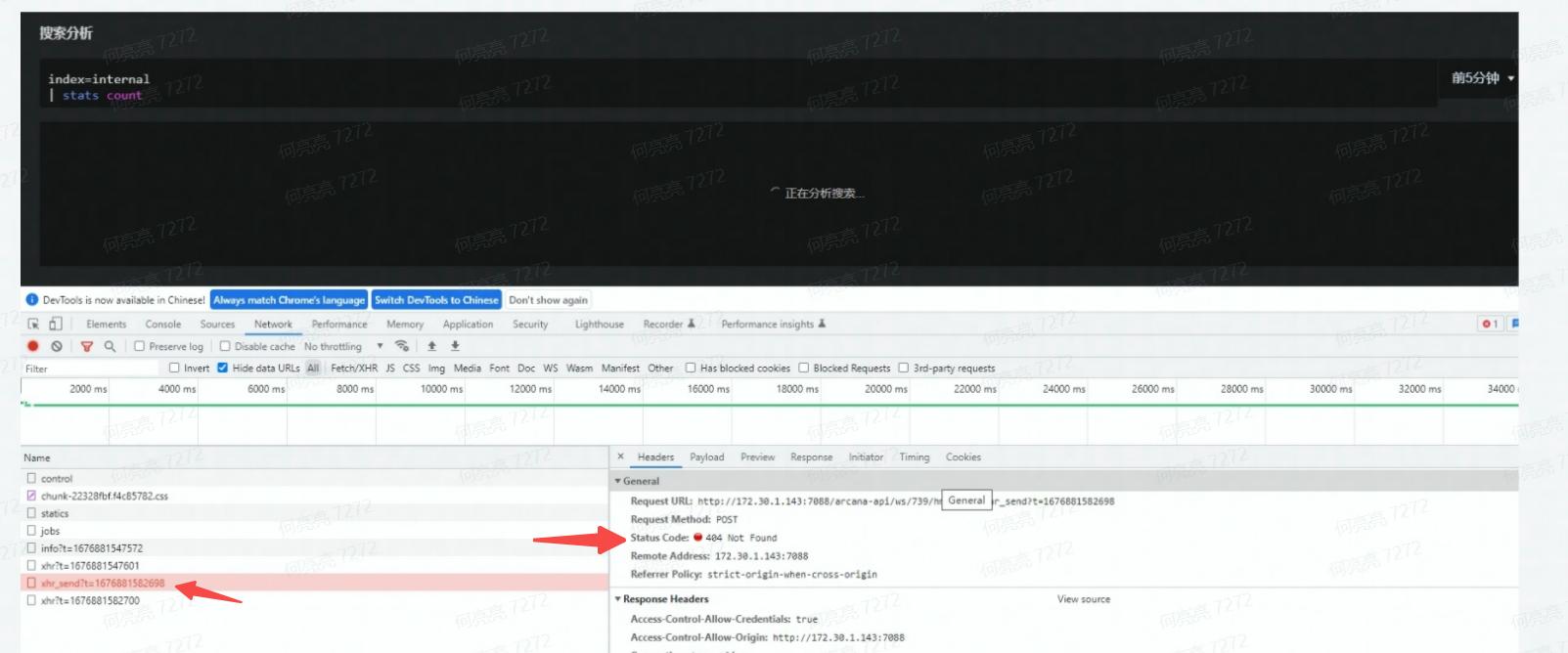


1. 输入命令yum install sysstat，在提示“Is this ok [y/d/N]:”，输入y即可。

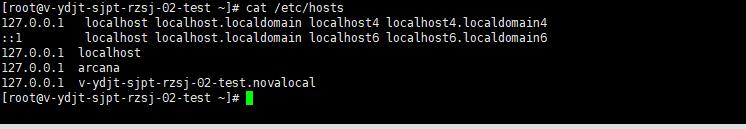


# 7.搜索分析相关问题

**1、Q：在搜索页面查询SPL，有时能出来结果，但大部分时候出不来结果。经排查，搜索的SPL在var/jobs下有产生结果，但websocket去查询结果会报404。**



A：修改etc/hosts配置，添加127.0.0.1 arcana



1. **Q：V3.0.2212版本如何设置\_time时间精确到毫秒级。**

A：进入/arcana/share/tomcat/webapps/Di-Logger中修改config.js中的参数searchMillisecond：



searchMillisecond：false，\_time精确到秒

searchMillisecond：true，\_time精确到毫秒秒

# 8.仪表板相关问题

**1、Q：如何隐藏仪表板的工具栏？**

A：1）点击指定dashboard所在行的编辑按钮，点击“文本编辑方式”。在最后一个“}”前添加下面红框中的内容：



1. 编辑完成后，点击“保存”。

**2、Q：如何复制仪表板配置？**

A：1）点击指定dashboard所在行的编辑按钮，点击“文本编辑方式”，复制需要的配置。

2）点击需要复制到的dashboard所在行的编辑按钮，点击“文本编辑方式”，将刚才复制的内容粘贴进去。

3）点击“保存”，即可看到复制好的配置。

# 9.邮件服务器相关问题

**1、Q:邮件服务器配置后，离线报告、告警等不能正常使用邮件发送。**

A：可能存在原因包括邮件服务器配置错误，网络不通等。检查步骤：

1. 配置正确的SMTP邮件服务器地址。常见服务器，

企业微信：[SMTP.EXMAIL.QQ.COM](http://SMTP.EXMAIL.QQ.COM) 端口：25，465（SSL）

飞书：[smtp.feishu.cn](http://smtp.feishu.cn) 端口：25，465（SSL）

1. 用户名密码需正确配置。密码一般均为授权码，通过对应邮件管理页面进行生成。
2. 检查网络是否通畅。需要能ping通对应的服务器，能telnet对应的端口。例如：

ping [smtp.exmail.qq.com](http://smtp.exmail.qq.com)

telnet [smtp.exmail.qq.com](http://smtp.exmail.qq.com) 465

# 10.存储使用相关问题

**1、ES默认配置了85%的水位线，如果ES所在磁盘文件系统使用率（可以通过df -Th查看）超过85%，则会导致ES锁住，只能读不能写。**

**2、Q：如果文件系统使用率快到85%了，怎么办？**

A：有以下几种解决方法：

1. 联系客户进行磁盘扩容
2. 对历史文件进行清理。
   1. 日志文件：$ARCANA\_PATH/logs下的压缩文件与日志文件。建议至少保留最近3天的日志，供后续问题排查。
   2. 历史安装文件：历史升级安装文件可以迁移走，也可以删除。按实际情况而定。
3. 缩短kafka数据过期时效。
   1. $ARCANA\_PATH/conf/kafka.properties 文件下 缩短log.retention.hours的值。比如原来是72，可以改成24。表示kafka内的数据只保留24小时。
   2. $ARCANA\_Edge\_PATH/conf/kafka.properties 文件下 缩短log.retention.hours的值。
4. 在与客户确认后，删除历史索引数据。
   1. 正常情况下，通过设置索引管理里的过期时间，让系统在晚上自动清理。
   2. 【慎用】紧急情况下，可以直接删除历史索引文件。注：此方法会直接删除单个索引历史文件，不能控制时间；一定不能删除最后一个分片。

操作步骤：

* 1. 先查询分片具体名字 curl <http://127.0.0.1:9200/_cat/indices?v>
  2. 针对有多个分片的索引数据，删除历史分片。比如上述命令可以看到winlog\_01有3个分片dm-winlog\_01-000001，dm-winlog\_01-000002，dm-winlog\_01-000003，可以考虑将第一个分片进行删除：curl -XDELETE <http://127.0.0.1:9200/dm-winlog_01-000001>

# 11.通用配置

**1、Q：如何隐藏用户？**

A：在ARCANA版本为3.1.1及以上版本中做如下操作

1. 以root用户登录ARCANA所在的服务器。
2. 创建/opt/arcana/share/tomcat/webapps/system/local/ROOT目录。
3. 将custom.css文件上传至/opt/arcana/share/tomcat/webapps/system/local/ROOT目录下。

说明：custom.css文件的获取方式请联系内部工作人员。

1. 执行以下命令，重启ARCANA。

cd /opt/arcana/bin

su - arcana //ES场景下需要切换至arcana用户执行。

./arcana restart

**2、Q：如何配置正则文件解析规则**

A：当使用内置解析引擎ArcanaRegexModel时，需要手动将解析规则上传。

1. 使用root用户登录ARCANA后台，进入/opt/arcana/etc/transform目录下。

cd /opt/arcana/etc/transform

说明：/opt/arcana为ARCANA的安装目录，请根据实际部署环境。

1. 执行以下命令，创建文件。文件名请根据实际情况配置，这边以regex.conf为例。

touch regex.conf

1. 执行以下命令，在文件中存入regex规则。

vim regex.conf

[log]

EXTRACT-fields =\[(?<DATE>[^]+)\s+(?<TIME>[^]+)\)](?<user>.+)访问服务IP\:(?<ip>\d+\.\d+\.\d+\.\d+)

EVAL-\_time =DATE+""+TIME

EVAL-\_time =string\_to\_date(\_time,'yyyy-MM-DD HH:mm:ss')

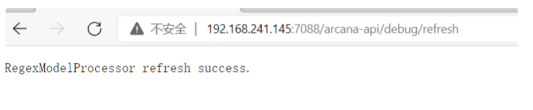
EVAL-DATE=nil

EVAL-DATE=nil

1. 配置完成后，登录192.168.241.145:7088/arcana-api/debug/refresh（单独的正则接口）。

192.168.241.145为安装arcana的服务器IP，请根据实际修改。

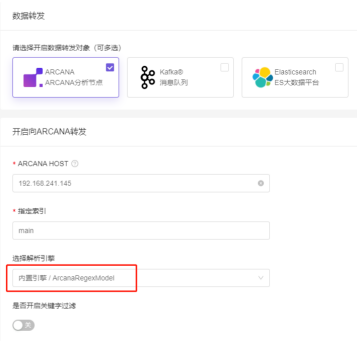
页面显示如下所示，说明规则上传成功：



1. arcana配置收数实例---日志文件，指定文件路径并且来源标签必须是：xxx.log。

xxx是conf文件的名称，log是解析规则的[log]开头标签。





**3、Q：如何修改日志保留天数**

**A：**1）修改ARCANA端

vim /opt/arcana/conf/kafka.properties

log.retention.hours=168

2）修改ARCANA Edge端

vim /opt/arcana/conf/kafka.properties

log.retention.hours=168

**4、Q：如何配置KafKa SSL认证。**

**生成加密证书**

1. 在/etc/hosts文件中自定义一个hosts名。

说明：kafka2.0.x开始，将ssl.endpoint.identification.algorithm默认设置为HTTPS，即：需要验证主机名，所以我们这里可以先配置一个主机名。 追注：不论是否需要主机名验证，都推荐配置一下hosts。

1. 创建证书保存目录。

mkdir -p /usr/ca/{root,server,client,trust}

/usr/ca/为证书存放路径，请根据实际需要配置。

1. 生成server.keystore.jks文件(即生成服务端的keystore文件)。

keytool -keystore /usr/ca/server/server.keystore.jks -alias ds-kafka-single -validity 365 -genkey -keypass DiMatrix -keyalg RSA -dname "CN=kafka-single,OU=aspire,O=aspire,L=beijing,S=beijing,C=cn" -storepass DiMatrix -ext SAN=DNS:kafka-single

说明：加粗字体分别为证书地址，指定别名条目的密码，hosts配置的主机名，指定密钥库的密码，hosts配置的主机名。

keytool相关指令说明：

|  |  |
| --- | --- |
| **指定** | **含义** |
| -alias | 别名 |
| -keystore | 指定密钥库的名称(就像数据库一样的证书库，可以有很多个证书，cacerts这个文件是jre自带的， 也可以使用其它文件名字，如果没有这个文件名字，它会创建这样一个) |
| -storepass | 指定密钥库的密码 |
| -keypass | 指定别名条目的密码 |
| -list | 显示密钥库中的证书信息 |
| -v | 显示密钥库中的证书详细信息 |
| -export | 将别名指定的证书导出到文件 |
| -file | 参数指定导出到文件的文件名 |
| -delete 删除密钥库中某条目 | 删除密钥库中某条目 |
| -import | 将已签名数字证书导入密钥库 |
| -keypasswd | 修改密钥库中指定条目口令 |
| -dname | 指定证书拥有者信息。其中，CN=名字与姓氏/域名,OU=组织单位名称,O=组织名称,L=城市或区域名称,ST=州或省份名称,C=单位的两字母国家代码 |
| -keyalg | 指定密钥的算法 |
| -validity | 指定创建的证书有效期多少天 |
| -keysize | 指定密钥长度 |

1. 生成CA认证证书(为了保证整个证书的安全性，需要使用CA进行证书的签名保证)。

利用Linux自带的OpenSSL创建即可。

openssl req -new -x509 -keyout /usr/ca/root/ca-key -out /usr/ca/root/ca-cert -days 365 -passout pass:DiMatrix -subj "/C=cn/ST=beijing/L=beijing/O=aspire/OU=aspire/CN=kafka-single"

1. 通过CA证书创建一个客户端信任证书。

keytool -keystore /usr/ca/trust/client.truststore.jks -alias CARoot -import -file /usr/ca/root/ca-cert -storepass DiMatrix

1. 通过CA证书创建一个服务端器端信任证书。

keytool -keystore /usr/ca/trust/server.truststore.jks -alias CARoot -import -file /usr/ca/root/ca-cert -storepass DiMatrix

1. 服务器证书的签名处理。
2. 导出服务器端证书server.cert-file。

keytool -keystore /usr/ca/server/server.keystore.jks -alias ds-kafka-single -certreq -file /usr/ca/server/server.cert-file -storepass DiMatrix

1. 用CA给服务器端证书进行签名处理。

openssl x509 -req -CA /usr/ca/root/ca-cert -CAkey /usr/ca/root/ca-key -in /usr/ca/server/server.cert-file -out /usr/ca/server/server.cert-signed -days 365 -CAcreateserial -passin pass:DiMatrix

1. 将CA证书导入到服务器端keystore。

keytool -keystore /usr/ca/server/server.keystore.jks -alias CARoot -import -file /usr/ca/root/ca-cert -storepass DiMatrix

1. 将已签名的服务器证书导入到服务器keystore。

keytool -keystore /usr/ca/server/server.keystore.jks -alias ds-kafka-single -import -file /usr/ca/server/server.cert-signed -storepass DiMatrix

参考 https://blog.csdn.net/justry\_deng/article/details/88383081

**Kafka服务配置**

修改kafka安装目录下config目录下的server.properties文件。

####################### Server Basics #############################

**# SSL认证配置**

**# 如果配置了SSL认证，那么原来的port和advertised.listeners可以注释掉了**

**#如果broker之间通讯未启用SSL listeners=PLAINTEXT://host.name:port,SSL://host.name:port**

**#PLAINTEXT和SSL端口是必须要配置。**

**listeners=SSL://kafka-single:9095**

**advertised.listeners=SSL://kafka-single:9095**

**ssl.key.password=ds1994**

**ssl.keystore.type=JKS**

**ssl.keystore.location=/usr/ca/server/server.keystore.jks**

**ssl.keystore.password=ds1994**

**ssl.truststore.type=JKS**

**ssl.truststore.location=/usr/ca/trust/server.truststore.jks**

**ssl.truststore.password=ds1994**

**#“required”=>客户端身份验证是必需的，“requested”=>客户端身份验证请求，客户端没有证书仍然可以连接。使用“requested”是纸老虎，因为它提供了一种虚假的安全感，错误的配置客户端仍将连接成功。**

**#如果开启客户端身份验证，则客户端必须配置keystore**

**#ssl.client.auth=required**

**ssl.client.auth=none**

**ssl.enabled.protocols=TLSv1.2,TLSv1.1,TLSv1**

**# kafka2.0.x开始，将ssl.endpoint.identification.algorithm设置为了HTTPS，即:需要验证主机名**

**# 如果不需要验证主机名，那么可以这么设置 ssl.endpoint.identification.algorithm=即可**

**#ssl.endpoint.identification.algorithm=HTTPS**

**ssl.endpoint.identification.algorithm=**

**# 设置内部访问也用SSL，默认值为security.inter.broker.protocol=PLAINTEXT**

**security.inter.broker.protocol=SSL**

**#security.inter.broker.protocol=PLAINTEXT**

注：如果设置的内部broker的通讯协议PLAINTEXT，那么监听PLAINTEXT的时候就需要作相应的配置 listeners=PLAINTEXT://host.name:port,SSL://host.name:port。

**重启Kafka**

# 前台启动kafka

bin/kafka-server-start.sh config/server.properties

使用Linux自带的openssl测试一下，验证我们配置的ssl有效

openssl s\_client -debug -connect kafka

**5、Q：如何配置Beats相关证书**

为了保证应用日志数据的传输安全，我们可以使用SSL相互身份验证来保护Filebeat和Logstash之间的连接。 这可以确保Filebeat仅将加密数据发送到受信任的Logstash服务器，并确保Logstash服务器仅从受信任的Filebeat客户端接收数据。 下面就讲述一下配置Filebeat与Logstash之间进行加密通信的方法。全文是在CentOS7上基于Elastic 7.5.0技术栈所验证的。

需要一个自签的CA证书，以及使用该CA证书签署的两份数据证书。一份是给Logstash作为server端验证自己身份时使用，一份是提供给Filebeat客户端验证自己身份使用。 直接利用的Elasticsearch随安装包提供的数字证书工具elasticsearch-certutil来制作需要的证书。

官方地址：

https://www.elastic.co/guide/en/elasticsearch/reference/7.5/certutil.html

**制作自签的CA证书**

在Linux下，进入到Elasticsearch程序的部署家目录中，执行以下命令可以生成一份自签的CA证书：

**./bin/elasticsearch-certutil ca**

使用默认输出文件名elastic-stack-ca.p12，并为证书设置访问口令。

根据证书文件导出一份CA公钥文件，用于后续各应用配置文件中引用CA公钥时使用：

**openssl pkcs12 -clcerts -nokeys -in elastic-stack-ca.p12 -out ca.pem**

或

**.bin/elasticsearch-certutil cert --ca elastic-stack-ca.p12 # elastic-certificates.p12 openssl pkcs12 -in elastic-certificates.p12 -cacerts -nokeys -out out ca.pem**

**制作Logstash使用的数字证书**

Logstash服务在启用SSL加密通信支持时，会有一个特殊的问题。因为Logstash在底层是通过集成了Netty来提供的对外服务端口，而Netty在支持数字证书这一功能上面，有一个局限性，即Netty仅支持使用PKCS#8的密钥格式。

1. 使用openssl转换出一份PKCS#8格式的密钥文件，即logstash.p8。

**openssl pkcs8 -in logstash.key -topk8 -nocrypt -out logstash.p8**

1. 生成logstash数字证书。

**./bin/elasticsearch-certutil cert --ca elastic-stack-ca.p12 --name logstash --dns logstash --ip 192.168.1.235 --pem --out logstash.zip**

1. 解压logstash的证书至logstash的安装路径下或放至统一证书路径，如/etc/certificate/logstash下。

unzip -n logstash.zip -d /etc/certificate/logstash

**制作Filebeats使用的数字证书**

./bin/elasticsearch-certutil cert --ca elastic-stack-ca.p12 --name filebeat --dns filebeat --ip 192.168.75.20 --pem --out filebeat.zip unzip filebeat.zip

解压filebeat的证书至filebeat的安装路径下或放至统一的证书路径，如/etc/certificate/filebeat下

<!--#解压后会各有一个.key和.crt后缀的文件-->

unzip  -n  filebeat.zip  -d /etc/certificate/filebeat

**Beats配置SSL**

filebeat.yml文件配置output，如

output.logstash:

    hosts: ["192.168.1.235:5000"]

        ssl.certificate\_authorities: ["/etc/certificate/ca.pem"]

        ssl.certificate: ["/etc/certificate/filebeat.crt"]

        ssl.key: ["/etc/certificate/filebeat.key"]

logstash配置接收filebeat接收日志时使用SSL

input {

    beats {

            ssl => true

            ssl\_key => "/etc/certificate/logstash/logstash.p8"

            ssl\_certificate => "/etc/certificate/logstash/logstash.crt"

            ssl\_verify\_mode => "force\_peer"

            ssl\_certificate\_authorities => ["/etc/certificate/logstash/ca.pem"]

            port => "9000"

            add\_field => {

                    "[fields][forward\_server]" => "localhost:9092"

                    "[fields][forward\_type]" => "arcana"

                    "[fields][edge\_id]" => "607d3730084b1b633115c518"

                    "[fields][store\_name]" => "beat-ssl"

                    "[fields][source]" => "datasource"

                    "[fields][type]" => "beats"

            }

         }

     }

# 12.离线报告相关

**1、Q：模板中表格中的正文文字段落未缩进，生成的报告却显示首行缩进，导致报告排版出现问题；**

A：WPS与word不完全兼容，若使用WPS编辑模板或从别的WPS文档中复制内容放A入word模板，会导致生成的报告格式出现问题。建议禁止使用WPS打开/编辑，禁止从WPS文档中复制内容粘贴入Word；

**2、Q：word模板排版不规范**

现象：

* 某个表格一行占了好几页，仔细排查发现单元格内某个文字字体大小达到了3000多字号大小；
* 生成的报告中出现空白页，检查发现是下章节有段前分页；
* 表格排版错乱有的单元格会文字缩进，有的没有，字体不统一；
* 表格页码错乱

A：建议：

* 规范排版，尽量使用模板进行复制修改；
* 多刷格式，保持所有表格、标题、正文格式一致；
* 段前不分页、断后不分页；
* 表格或图片不超过页边距；

**3、Q重名文件上传不生效，**上传与历史文件同名的新报告模板，发现生成的报告内容并不是新报告模板的内容；

A：建议：上传报告模板文件名须有命名规范进行区分，如文件名后加上-日期等；

**4、Q：饼图若有搜索数据为0，则饼状图会空白不显示任何内容；**

A：建议如果易出现结果全部为0，建议使用曲线图或柱状图，不使用饼状图。