Elastalert 简介

- 前言
- 一、安装
  - 1.1 安装python3.6
  - 1.2 安装elastalert
- 二、使用
  - 2.1 配置config.yaml
  - 2.2 配置rule
  - 2.3 启动
- 三、实例
  - 3.1 示例一
  - 3.2 示例二
  - 3.3 示例三

前言

ElastAlert 是 Yelp 公司开源的一套用 Python2.6（调研过程中发现目前新版本只支持python3.6） 写的报警框架。属于后来 Elastic.co 公司出品的 Watcher 同类产品。官网地址见：http://elastalert.readthedocs.org/。下图展示了我们监控系统的组件及架构简图

一、安装

1.1 安装python3.6

测试过程中发现最新版本的elastalert只支持python3.6，所以首先需安装python3.6环境。此处使用anaconda安装。

```
wget http://172.16.56.39/Anaconda3/Anaconda3-5.2.0-Linux-x86_64.sh
bash Anaconda3-5.2.0-Linux-x86_64.sh
```

根据提示按ENTER

```
[root@ansible Anaconda3]# bash Anaconda3-5.2.0-Linux-x86_64.sh

Welcome to Anaconda3 5.2.0

In order to continue the installation process, please review the license
agreement.
Please, press ENTER to continue
>>>
```

根据提示输入yes,同意license agreement

```
pycrypto
    A collection of both secure hash functions (such as SHA256 and RIPEMD16

pyopenssl
    A thin Python wrapper around (a subset of) the OpenSSL library.

kerberos (krb5, non-Windows platforms)
    A network authentication protocol designed to provide strong authentica

cryptography
    A Python library which exposes cryptographic recipes and primitives.


Do you accept the license terms? [yes|no]
[no] >>>
```

如果直接ENTER键，用默认的安装路径（/root/anaconda3），建议此处使用/usr/local/anaconda3，直接输入

/usr/local/anaconda3，然后ENTER

```
Anaconda3 will now be installed into this location:
/root/anaconda3

  - Press ENTER to confirm the location
  - Press CTRL-C to abort the installation
  - Or specify a different location below

[/root/anaconda3] >>>
```

写入环境变量，直接输入yes

```
... ...
installing: zlib-1.2.8-3 ...
installing: anaconda-4.4.0-np112py27_0 ...
installing: conda-4.3.21-py27_0 ...
installing: conda-env-2.6.0-0 ...
Python 2.7.13 :: Continuum Analytics, Inc.
creating default environment...
installation finished.
Do you wish the installer to prepend the Anaconda2 install location
to PATH in your /root/.bashrc ? [yes|no]
[no] >>> yes
```

1.2 安装elastalert

```
#
yum install gcc libffi-devel python-devel openssl-devel python-setuptools
#
git clone https://github.com/Yelp/elastalert.git
/usr/local/anaconda3/bin/pip install -r requirements.txt
/usr/local/anaconda3/bin/pip install "setuptools>=11.3"
/usr/local/anaconda3/bin/python setup.py install

# /usr/local/anaconda3/bin/pip install elastalert
Elasticsearch 5.0+:
pip install "elasticsearch==5.5.3"
Elasticsearch 2.X:
pip install "elasticsearch<3.0.0"
```

【注】：因为目前我们使用的elasticsearch版本为5.6.16，所以经测试此处如果python elasticsearch库版本安装7.0版本的话，会无法使用，所以建议安装为elasticsearch==5.5.3

elastalert /usr/local/anaconda3/bin/

```
[root@es-5 supervisor]# ll /usr/local/anaconda3/bin/elastalert*
-rwxr-xr-x 1 root root 239 11月  1 13:34 /usr/local/anaconda3/bin/elastalert
-rwxr-xr-x 1 root root 241 11月  1 13:34 /usr/local/anaconda3/bin/elastalert-create-index
-rwxr-xr-x 1 root root 245 11月  1 13:34 /usr/local/anaconda3/bin/elastalert-rule-from-kibana
-rwxr-xr-x 1 root root 238 11月  1 13:34 /usr/local/anaconda3/bin/elastalert-test-rule
```

```
elastalert-create-index  ESelastalert_status
elastalert-test-rule     rule
elastalert-rule-from-kibana   Kibana3Filters
```

二、使用

2.1 配置config.yaml

创建配置文件，在elastalert 目录里边有一个config.yaml.example 文件，我们通过copy 一份之后修改成自己需要的配置

```
cp  config.yaml.example config.yaml
vi config.yaml
```

```
#elastalert elastalert
# This is the folder that contains the rule yaml files
# Any .yaml file will be loaded as a rule
rules_folder: /usr/local/elastalert/example_rules

#Elastalert elasticsearch
# How often ElastAlert will query Elasticsearch
# The unit can be anything from weeks to seconds
run_every:
  #seconds1
  minutes: 1
  #hours1
  #days1
  #weeks1

#
# ElastAlert will buffer results from the most recent
# period of time, in case some log sources are not in real time
buffer_time:
  minutes: 15

#Elasticsearch ip
# The Elasticsearch hostname for metadata writeback
# Note that every rule can have its own Elasticsearch host
es_host: 172.16.63.5

#Elasticsearch
# The Elasticsearch port
es_port: 9200

#TLS
# Connect with TLS to Elasticsearch
#use_ssl: True

#TLS
# Verify TLS certificates
#verify_certs: True

#Elasticsearch
# Option basic-auth username and password for Elasticsearch
#es_username: someusername
#es_password: somepassword

#
# Use SSL authentication with client certificates client_cert must be
# a pem file containing both cert and key for client
#verify_certs: True
```

```
#ca_certs: /path/to/cacert.pem
#client_cert: /path/to/client_cert.pem
#client_key: /path/to/client_key.key

#elastalert esindex
# The index on es_host which is used for metadata storage
# This can be a unmapped index, but it is recommended that you run
# elastalert-create-index to set a mapping
writeback_index: elastalert_status

#alert
# If an alert fails for some reason, ElastAlert will retry
```

```
# sending the alert until this time period has elapsed
alert_time_limit:
  days: 2
```

配置过后，首先执行如下命令在elasticsearch中创建elastalert的日志索引

```
/usr/local/anaconda3/bin/elastalert-create-index --config
/etc/elastalert/config.yaml
```

elastalert-create-index ElastAlert会将执行记录存放到一个ES索引中，该命令就是用来创建这个索引的，默认情况下，索引名叫elastalert_status。其中有4个_type，都有自己的@timestamp字段，所以同样也可以用kibana来查看这个索引的日志记录情况。

2.2 配置rule

```
cd example_rules/
cp example_frequency.yaml my_rule.yaml
vi my_rule.yaml
```

```
# Alert when the rate of events exceeds a threshold
#Elasticsearch
# (Optional)
# Elasticsearch host
es_host: 192.168.115.65

#Elasticsearch
# (Optional)
# Elasticsearch port
es_port: 9200

#ssl
# (OptionaL) Connect with SSL to Elasticsearch
#use_ssl: True

#elasticsearch
# (Optional) basic-auth username and password for Elasticsearch
#es_username: someusername
#es_password: somepassword

#rule name
# (Required)
# Rule name, must be unique
name: xx-xx-alert

# anyblacklistwhitelistchangefrequencyspikeflatlinenew_termcardinality
any
```

```
blacklistcompare_key blacklist
whitelistcompare_keywhitelist
changequery_keycompare_key timeframe
frequency query_keytimeframe num_events
spikequery_keytimeframespike_heightspike_type- up,down,both
threshold_refthreshold_cur
flatlinetimeframe threshold
new_termfieldsterms_window_size(30)terms_size (50)
cardinality query_keytimeframecardinality_field max_cardinality
min_cardinality
# (Required)
# Type of alert.
# the frequency rule type alerts when num_events events occur with
timeframe time
#frequency query_keytimeframe num_events
type: frequency

#index kibana indexindex*
# (Required)
# Index to search, wildcard supported
index: es-nginx*,winlogbeat*

#
# (Required, frequency specific)
# Alert when this many documents matching the query occur within a
timeframe
num_events: 5

#45
# (Required, frequency specific)
# num_events must occur within this amount of time to trigger an alert
timeframe:
  minutes: 4

#messageelasticsearch query AND&OR
# (Required)
# A list of Elasticsearch filters used for find events
# These filters are joined with AND and nested in a filtered query
# For more info:
http://www.elasticsearch.org/guide/en/elasticsearch/reference/current/quer
y-dsl.html
filter:
- query:
    query_string:
      query: "message:   OR Error"
#
#alert
https://elastalert.readthedocs.io/en/latest/recipes/adding_rules.html#writ
ingrules
https://elastalert.readthedocs.io/en/latest/recipes/adding_alerts.html#wri
tingalerts

Command
```

```
Email
JIRA
OpsGenie
SNS
HipChat
Slack
Telegram
Debug
Stomp
# (Required)
# The alert is use when a match is found
alert:
- "email"
#alert_text
alert_text: "Ref Log http://192.168.115.65"
#smtp server
smtp_host: smtp.126.com
#smtp
smtp_port: 25
#userpassword
smtp_auth_file: /usr/local/elastalert/example_rules/smtp_auth_file.yaml
email_reply_to:test@126.com
from_addr: test@126.com

#,
# (required, email specific)
# a list of email addresses to send alerts to
```

```
email:
- "test@126.com"
- "test1@126.com"
```

接下来我们需要配置smtp认证文件了，touch 一个配置文件

```
vi smtp_auth_file.yaml
#
user: "test"
password: "test@12345"
```

然后通过elastalert-test-rule 测试一下我们写的rule 是否有问题

```
/usr/local/anaconda3/bin/elastalert-test-rule my_rule.yaml --config
/etc/elastalert/config.yaml
```

测试结果如下，如果有问题会提示问题，如果没有问题就会告诉你successfully。

```
[root@kube-worker-01 rules]# /usr/local/anaconda3/bin/elastalert-test-rule ips_error_frequency.yaml --config /etc/elastalert/config.yaml
/usr/local/anaconda3/lib/python3.6/site-packages/requests/__init__.py:80: RequestsDependencyWarning: urllib3 (1.25.6) or chardet (3.0.4) doesn't match a supported version!
  RequestsDependencyWarning)
INFO:elastalert:Note: In debug mode, alerts will be logged to console but NOT actually sent.
           To send them but remain verbose, use --verbose instead.
Didn't get any results.
INFO:elastalert:Note: In debug mode, alerts will be logged to console but NOT actually sent.
                To send them but remain verbose, use --verbose instead.
1 rules loaded
INFO:apscheduler.scheduler:Adding job tentatively -- it will be properly scheduled when the scheduler starts
INFO:elastalert:Queried rule ips-error frequency rule from 2019-11-01 16:08 CST to 2019-11-01 16:18 CST: 0 / 0 hits
INFO:elastalert:Queried rule ips-error frequency rule from 2019-11-01 16:18 CST to 2019-11-01 16:18 CST: 0 / 0 hits

Would have written the following documents to writeback index (default is elastalert_status):

elastalert_status - {'rule_name': 'ips-error frequency rule', 'endtime': datetime.datetime(2019, 11, 1, 8, 18, 34, 447982, tzinfo=tzutc()), 'starttime': datetime.datetime(201
9, 11, 1, 8, 8, 28, 447982, tzinfo=tzutc()), 'matches': 0, 'hits': 0, '@timestamp': datetime.datetime(2019, 11, 1, 8, 18, 34, 574079, tzinfo=tzutc()), 'time_taken': 0.0529339
3135070801}

[root@kube-worker-01 rules]#
```

2.3 启动

配置检查成功之后，就可以把程序启动起来

1. 如果只匹配一个rule启动，可采用如下方式：

```
/usr/local/anaconda3/bin/python -m elastalert.elastalert --verbose --rule
my_rule.yaml  --config /etc/elastalert/config.yaml
```

2. 如果config.yaml文件中rules_folder配置项配置的目录下面有多个rule规则，同时需要匹配，可采用如下方式启动

```
/usr/local/anaconda3/bin/python -m elastalert.elastalert --config
/etc/elastalert/config.yaml --verbose
```

3. 最后测试没问题之后，我们可以使用supervisor来管理elastalert

```
yum install -y supervisor
```

配置elastalert program

```
vim /etc/supervisord.d/elastalert.ini
[program:elastalert]
command=/usr/local/anaconda3/bin/python -m elastalert.elastalert --config
/etc/elastalert/config.yaml --verbose
autostart=true
autorestart=true
stdout_logfile=/var/log/supervisor/elastalert.log
stderr_logfile=/var/log/supervisor/elastalert_error.log
stopsignal=INT
stderr_logfile_maxbytes=5MB
stdout_logfile_maxbytes=5MB
```

启动服务

```
supervisorctl start elastalert
```

三、实例

3.1 示例一

监控index为ips-* 的日志，10m 中Severity: CRITICAL 的日志大于1条就发送告警邮件

rules文件如下：

```
# Alert when the rate of events exceeds a threshold

# (Optional)
# Elasticsearch host
# es_host: elasticsearch.example.com

# (Optional)
# Elasticsearch port
# es_port: 14900

# (OptionaL) Connect with SSL to Elasticsearch
#use_ssl: True

# (Optional) basic-auth username and password for Elasticsearch
#es_username: someusername
#es_password: somepassword

# (Required)
```

```
# Rule name, must be unique
name: ips-error frequency rule

# (Required)
# Type of alert.
# the frequency rule type alerts when num_events events occur with
timeframe time
type: frequency

aggregation:
#  "* * * * *" means: run as the "run_every" in config.yaml
  schedule: "* * * * *"
aggregate_by_match_time: true

# (Required)
# Index to search, wildcard supported
index: ips-*

# (Required, frequency specific)
# Alert when this many documents matching the query occur within a
timeframe
num_events: 1

# (Required, frequency specific)
# num_events must occur within this amount of time to trigger an alert
timeframe:
  #hours: 4
  minutes: 10

# (Required)
# A list of Elasticsearch filters used for find events
# These filters are joined with AND and nested in a filtered query
# For more info:
http://www.elasticsearch.org/guide/en/elasticsearch/reference/current/quer
y-dsl.html
filter:
#- term:
#    some_field: "some_value"
#
#- bool:
#    #
#    must:
#      - match:
#          stream: "stderr"
    #
#    must_not:
#      - match:
#          stackTrace: "org.apache.catalina.connector.ClientAbortException:
java.io.IOException: Broken pipe"
#      - match:
#          message: "[SUCCESS]"
- query:
    query_string:
```

```
      #query: "Severity: CRITICAL or MEDIUM or HIGH "
      query: "Severity: CRITICAL"

##  realert
## 5  name
#query_key:
#  - name
#
## 5
#realert:
#  minutes: 5
#
##  realert
##  5 -> 10 -> 20 -> 40 -> 60
##  realert
#exponential_realert:
#  hours: 1
#
##  name
#aggregation_key: stream
#
##  name  message
#summary_table_fields:
#  - stream
#  - message


#mail
smtp_host: smtp.exmail.qq.com
smtp_port: 25
smtp_auth_file: /etc/elastalert/smtp_auth.yaml
#
email_reply_to: wangruihua@cechealth.cn
###
from_addr:  alarm@cechealth.cn
# (Required)
# The alert is use when a match is found
alert:
- "email"

email_format: html

# (required, email specific)
# a list of email addresses to send alerts to
email:
- "zhujiawen@cechealth.cn"
- "zhaoyuansheng@cechealth.cn"
- "wangruihua@cechealth.cn"

alert_subject: "[IPS-FIRING:{}] 10minutesipsSeverity: {} [{}],"
alert_subject_args:
  - Severity
  - Severity
```

```
  - num_hits

alert_text_type: alert_text_only
#
alert_text: |
  <div style='display:block;background-color: red;padding:
10px;border-radius: 5px;color: white;font-weight: bold;'>
    <p>{}</p>
  </div>
  <br>
  <a
href='http://172.16.63.5:5601/app/kibana#/discover?_g=(refreshInterval:(di
splay:Off,pause:!f,value:0),time:(from:now-15m,mode:quick,to:now))&_a=(col
umns:!(_source),filters:!(('$state':(store:appState),meta:(alias:!n,disabl
ed:!f,index:AW3toPFNKwXdc6wRiHnf,key:Severity,negate:!f,type:phrase,value:
CRITICAL),query:(match:(Severity:(query:CRITICAL,type:phrase))))),index:AW
3toPFNKwXdc6wRiHnf,interval:auto,query:(match_all:()),sort:!('@timestamp',
desc))' target='_blank' style='padding: 8px 16px;background-color:
#46bc99;text-decoration:none;color: white;border-radius: 5px;'>Click to
Kibana</a>
  <br>
  <h3></h3>
  <table>
    <tr>
      <td style='padding:5px;text-align: right;font-weight:
bold;border-radius: 5px;background-color: #eef;'>@timestamp:</td>
      <td style='padding:5px;border-radius: 5px;background-color:
#eef;'>{}</td></tr>
    <tr>
      <td style='padding:5px;text-align: right;font-weight:
bold;border-radius: 5px;background-color: #eef;'>_index:</td>
      <td style='padding:5px;border-radius: 5px;background-color:
#eef;'>{}</td></tr>
    <tr>
      <td style='padding:5px;text-align: right;font-weight:
bold;border-radius: 5px;background-color: #eef;'>AttackName:</td>
      <td style='padding:5px;border-radius: 5px;background-color:
#eef;'>{}</td></tr>
    <tr>
      <td style='padding:5px;text-align: right;font-weight:
bold;border-radius: 5px;background-color: #eef;'>DstIPAddr:</td>
      <td style='padding:5px;border-radius: 5px;background-color:
#eef;'>{}</td></tr>
    <tr>
      <td style='padding:5px;text-align: right;font-weight:
bold;border-radius: 5px;background-color: #eef;'>DstPort:</td>
      <td style='padding:5px;border-radius: 5px;background-color:
#eef;'>{}</td></tr>
    <tr>
      <td style='padding:5px;text-align: right;font-weight:
bold;border-radius: 5px;background-color: #eef;'>Severity:</td>
      <td style='padding:5px;border-radius: 5px;background-color:
#eef;'>{}</td></tr>
```

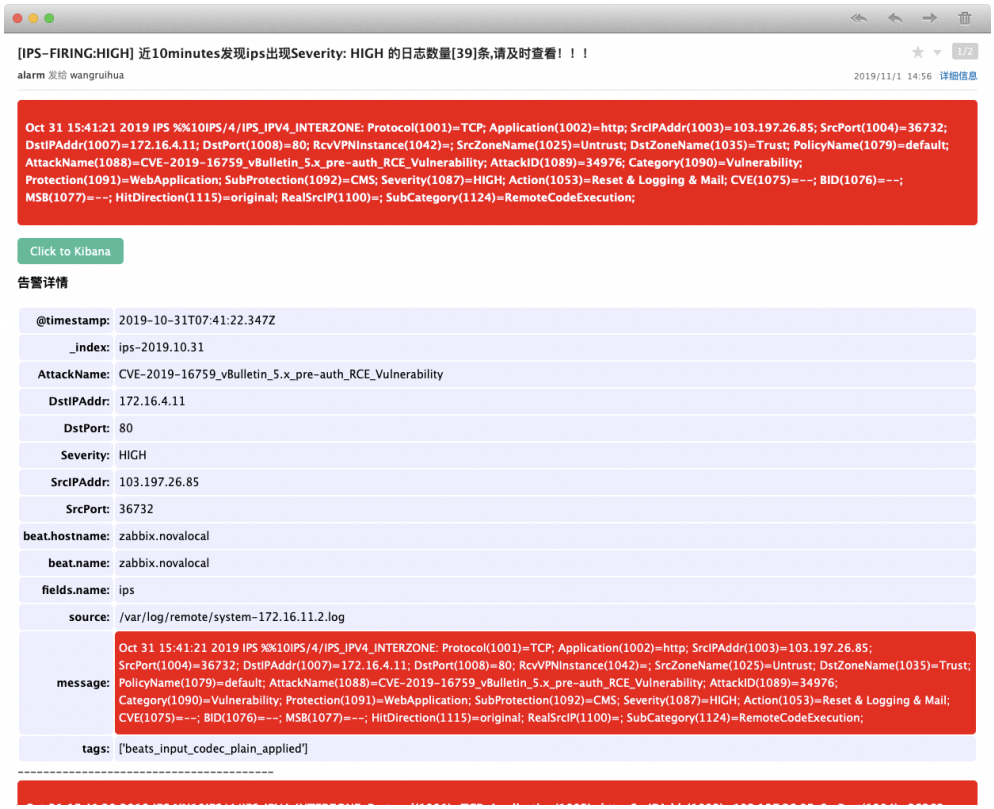
```
    <tr>
      <td style='padding:5px;text-align: right;font-weight:
bold;border-radius: 5px;background-color: #eef;'>SrcIPAddr:</td>
      <td style='padding:5px;border-radius: 5px;background-color:
#eef;'>{}</td></tr>
    <tr>
      <td style='padding:5px;text-align: right;font-weight:
bold;border-radius: 5px;background-color: #eef;'>SrcPort:</td>
      <td style='padding:5px;border-radius: 5px;background-color:
#eef;'>{}</td></tr>
    <tr>
      <td style='padding:5px;text-align: right;font-weight:
bold;border-radius: 5px;background-color: #eef;'>beat.hostname:</td>
      <td style='padding:5px;border-radius: 5px;background-color:
#eef;'>{}</td></tr>
    <tr>
      <td style='padding:5px;text-align: right;font-weight:
bold;border-radius: 5px;background-color: #eef;'>beat.name:</td>
      <td style='padding:5px;border-radius: 5px;background-color:
#eef;'>{}</td></tr>
    <tr>
      <td style='padding:5px;text-align: right;font-weight:
bold;border-radius: 5px;background-color: #eef;'>fields.name:</td>
      <td style='padding:5px;border-radius: 5px;background-color:
#eef;'>{}</td></tr>
    <tr>
      <td style='padding:5px;text-align: right;font-weight:
bold;border-radius: 5px;background-color: #eef;'>source:</td>
      <td style='padding:5px;border-radius: 5px;background-color:
#eef;'>{}</td></tr>
    <tr>
      <td style='padding:5px;text-align: right;font-weight:
bold;border-radius: 5px;background-color: #eef;'>message:</td>
      <td style='padding:10px 5px;border-radius: 5px;background-color:
red;color: white;'>{}</td></tr>
    <tr>
      <td style='padding:5px;text-align: right;font-weight:
bold;border-radius: 5px;background-color: #eef;'>tags:</td>
      <td style='padding:5px;border-radius: 5px;background-color:
#eef;'>{}</td></tr>
  </table>
 # area_textsprintf
alert_text_args:
 - message
 - "@timestamp"
 - _index
 - AttackName
 - DstIPAddr
 - DstPort
 - Severity
 - SrcIPAddr
 - SrcPort
 - beat.hostname
```

```
  - beat.name
  - fields.name
  - source
  - message
  - tags

#alert_text: |
#  ({})web
#  ### {}
#  > : {}
#  > timestamp: {} + 08:00
#  > attacker's ip: {}
#  > request: {}
#  > message: {}
#  > UA: {}
#  >>> {}
#
#alert_text_args:
#  - host
#  - num_hits
#  - time
#  - "@timestamp"
#  - remote_addr
#  - request
```

```
#   - message
#   - http_user_agent
#   - source
```

告警邮件如下图：



3.2 示例二

五分钟内流量总和超过200M就发邮件

```
es_host: 172.16.63.5
es_port: 9200
run_every:
  minutes: 5

name: nanjing_flow
type: metric_aggregation
index: n-xxx-*
buffer_time:
  minutes: 5

metric_agg_key: body_bytes_sent
metric_agg_type: sum
max_threshold: 209715200
use_run_every_query_size: true

alert_text_type: alert_text_only
alert_subject: "Alter nanjing 200M"
alert_text: |
  : {0} B
  kibana url: http://xxxxx

alert_text_args:
  - metric_body_bytes_sent_sum

smtp_host: smtp.qq.com
smtp_port: 25
smtp_auth_file: /opt/elastalert/rule_templates/smtp_auth_file.yaml
from_addr: "xxxx@qq.com"
alert:
- "email"
email:
- "xxxx@qq.com"
```

3.3 示例三

nginx例子，对后端请求超过3秒的发送邮件。需要对特定的接口，比如认证接口过滤(不计算在内)

```
es_host: 172.16.63.5
es_port: 9200
run_every:
  seconds: 30
name: xxx_reponse_time
index: n-xxx-*
type: whitelist
compare_key: "request"
ignore_null: true
whitelist:
```

```
  - /index.html
  - /siteapp/ecsAuthentication/hasAuthentication

type: frequency
num_events: 1
timeframe:
    seconds: 30
filter:
- query_string:
   query: "upstream_response_time: >3 "

alert_text_type: alert_text_only
alert_subject: "Alter {0} 3"
alert_subject_args:
- _index

html_table_title: "<h2>This is a heading</h2>"
alert_text: |
  timestamp: {0}
  request_method: {1}
  request: {2}
  request_body: {3}
  request_time: {4} s
  upstream_response_time: {5} s
  body_bytes_sent: {6} B
  status: {7}
  remote_addr: {8}
  http_x_forwarded_for: {9}
  upstream_addr: {10}
  agent: {11}

alert_text_args:
  - timestamp
  - request_method
  - request
  - request_body
  - request_time
  - upstream_response_time
  - body_bytes_sent
  - status
  - remote_addr
  - http_x_forwarded_for
  - upstream_addr
  - agent

smtp_host: smtp.qq.com
smtp_port: 25
smtp_auth_file: /opt/elastalert/rule_templates/smtp_auth_file.yaml
from_addr: "xxx@qq.com"
alert:
```

```
- "email"
email:
- "xxxxx@qq.com"
```