

2 Hướng dẫn triển khai ElastAlert

Triển khai **Elastalert-server** sử dụng docker

Yêu cầu:

- Cài đặt docker engine
- Custom source code tải từ github (cho phép sử dụng supervisord chạy nhiều profile thay thì elastalert-server mặc định sử dụng profile duy nhất tại `/opt/elastalert-server/elastalert.yaml`)

✓ Mục tiêu: Tạo elastalert-server cung cấp:

- elastalert2 thực hiện 3 folder rules với thời gian khác nhau bao gồm 1m, 5m và 1day
- API tương tác với rules tại `http://<server>:3030`

Cài đặt docker

Thực hiện truy cập (SSH) vào server `192.168.68.139` thực hiện update và cài đặt docker engine

- Cài đặt apt repository

```
1 // Add Docker's official GPG key:
2 $ sudo apt-get update
3 $ sudo apt-get install ca-certificates curl gnupg
4 $ sudo install -m 0755 -d /etc/apt/keyrings
5 $ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /etc/apt/keyrings/docker.gpg
6 $ sudo chmod a+r /etc/apt/keyrings/docker.gpg
7
8 // Add the repository to Apt sources:
9 $ echo \
10 "deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.gpg] https://download.docker.com/li
11 $(. /etc/os-release && echo "$VERSION_CODENAME") stable" | \
12 sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
13 $ sudo apt-get update
```

- Cài đặt docker engine

```
1 $ sudo apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compose-plugin
```

- Thêm user hiện tại vào group docker nếu cần

```
1 $ sudo groupadd docker
2 $ sudo usermod -aG docker $USER
```

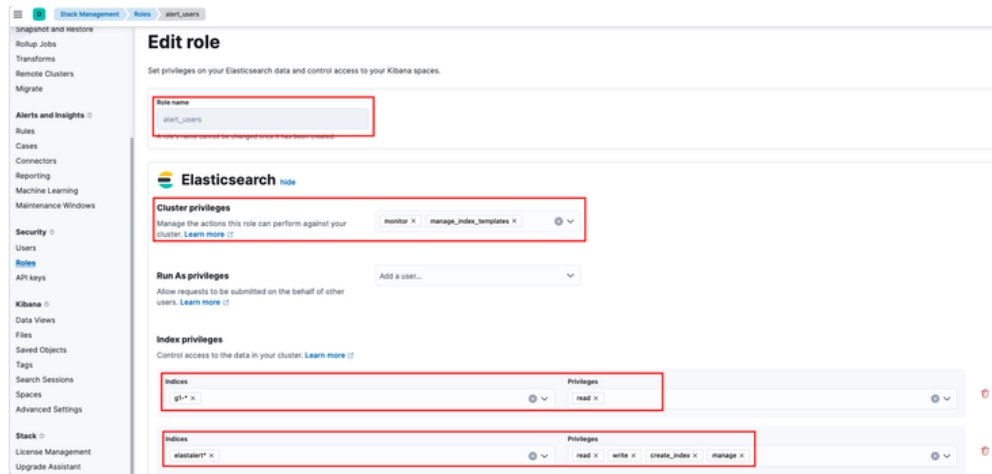
- Thiết lập tự động start docker.service

```
1 $ sudo systemctl enable docker
2 $ sudo systemctl enable containerd
3
4 $ sudo systemctl start docker
```

Chuẩn bị credentials và file cấu hình

- Tạo tài khoản (sử dụng kibana)

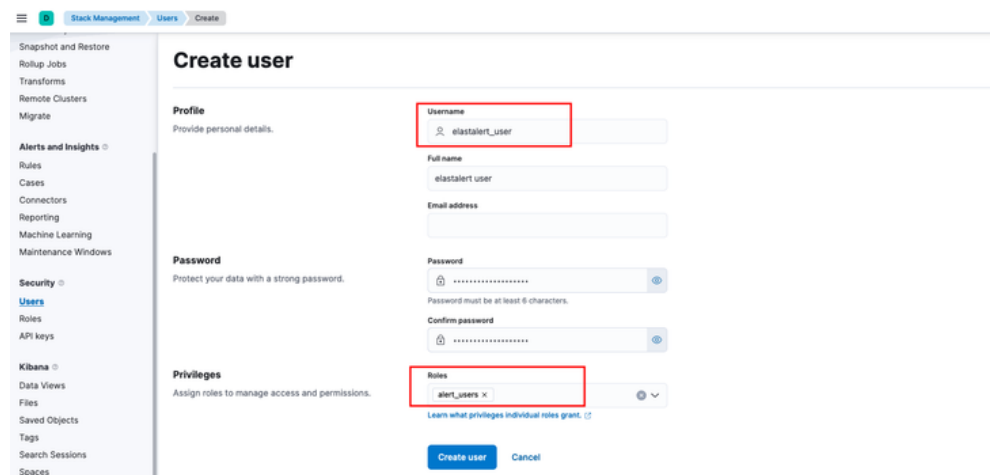
Để tạo tài khoản `elastalert_user` đầu tiên login vào giao diện kibana (sử dụng tài khoản `elastic`) và tạo role theo thao tác như sau: Truy cập `menu` → `Stack Management` → `Roles` → chọn button `Create role` sau đó nhập thông tin về role như hình:



Lưu ý cần chọn đúng index privileges để elastalert user có quyền đọc cũng như tạo các indices phù hợp

Nhấn **Create role** để hoàn tất tạo role `alert_users`

Tiếp theo tạo user với role `elastalert_user` như sau: Truy cập *menu* → **Stack Management** → **Users** → chọn button **Create user**



Nhấn **Create user** để hoàn tất tạo user `elastalert_user`

- Tạo và tải certificate từ es-node

Tiếp theo đối với certificate cho connection giữa kibana đến ES Cluster và giữa client đến server kibana chỉ cần tải certificate ở es-node01 đã tạo trước

Sử dụng `scp` để tải các certs trên es-node 01 như sau:

```
1 # cd /opt/elastalert-server-g1
2 # mkdir certs/
3 # scp -P 16623 <user>@192.168.68.141:/usr/share/elasticsearch/certs/ca/ca.crt certs/
4 # scp -P 16623 <user>@192.168.68.141:/usr/share/elasticsearch/certs/kibana/kibana.crt certs/
5 # scp -P 16623 <user>@192.168.68.141:/usr/share/elasticsearch/certs/kibana/kibana.key certs/
6 # chmod -R 755 certs/
```

Lưu ý thay đổi user truy cập SSH, elastalert-server đang cài đặt cùng với kibana server

Clone `elastalert-server` từ github tại [GitHub - johnsusek/elastalert-server](https://github.com/johnsusek/elastalert-server)

```
1 $ pwd
2 /home/g1admin
```

```
3 $ git clone https://github.com/johnsusek/elastalert-server
4 $ sudo cp elastalert-server /opt/elastalert-server-g1
5 $ sudo chown -R g1admin:g1admin /opt/elastalert-server-g1
```

⚠ Do nhu cầu cần chạy nhiều config profile đối với elastalert do đó cần thay đổi source code của elastalert-server để cho phép chạy nhiều elastalert profile sử dụng supervisord

Sau khi có được thư mục elastalert-server-g1 thay đổi thông tin của file `/opt/elastalert-server-g1/src/controllers/process/index.js`

```
1 $ pwd
2 /opt/elastalert-server-g1
3 $ cat > src/controllers/process/index.js << EOF
4 import { spawn, spawnSync } from 'child_process';
5 import config from '../../../common/config';
6 import Logger from '../../../common/logger';
7 import { Status } from '../../../common/status';
8
9 let logger = new Logger('ProcessController');
10
11 export default class ProcessController {
12
13   constructor() {
14     this._elastalertPath = config.get('elastalertPath');
15     this._writebackIndex = config.get('writeback_index');
16     this._onExitCallbacks = [];
17     this._status = Status.IDLE;
18
19     /**
20      * @type {ChildProcess}
21      * @private
22      */
23     this._process = null;
24   }
25
26   onExit(onExitCallback) {
27     this._onExitCallbacks.push(onExitCallback);
28   }
29
30   get status() {
31     return this._status;
32   }
33
34   /**
35    * Start ElastAlert if it isn't already running.
36    */
37   start() {
38     // Do not do anything if ElastAlert is already running
39     if (this._process !== null) {
40       logger.warn('ElastAlert is already running!');
41       return;
42     }
43
44     // Start ElastAlert from the directory specified in the config
45     logger.info('Starting ElastAlert');
46     this._status = Status.STARTING;
47
48     // Create ElastAlert index if it doesn't exist yet
49     logger.info('Creating index');
```

```

50     var indexCreate = spawnSync('python3', ['-m', 'elastalert.create_index', '--index', this._writebackIndex, '
51         cwd: this._elastalertPath
52     });
53
54     // Redirect stdin/stderr to logger
55     if (indexCreate.stdout && indexCreate.stdout.toString() !== '') {
56         logger.info(indexCreate.stdout.toString());
57     }
58     if (indexCreate.stderr && indexCreate.stderr.toString() !== '') {
59         logger.error(indexCreate.stderr.toString());
60     }
61
62     // Set listeners for index create exit
63     if (indexCreate.status === 0) {
64         logger.info(`Index create exited with code ${indexCreate.status}`);
65     } else {
66         logger.error(`Index create exited with code ${indexCreate.status}`);
67         logger.warn('ElastAlert will start but might not be able to save its data!');
68     }
69     logger.info("Start supervisord");
70
71     this._process = spawn('/usr/bin/supervisord', ['-c', '/etc/supervisor/supervisor.conf']);
72
73     logger.info(`Started Supervisord (PID: ${this._process.pid})`);
74     this._status = Status.READY;
75
76
77     // Redirect stdin/stderr to logger
78     this._process.stdout.on('data', (data) => {
79         logger.info(data.toString());
80     });
81     this._process.stderr.on('data', (data) => {
82         logger.error(data.toString());
83     });
84
85     // Set listeners for ElastAlert exit
86     this._process.on('exit', (code) => {
87         if (code === 0) {
88             logger.info(`ElastAlert exited with code ${code}`);
89             this._status = Status.IDLE;
90         } else {
91             logger.error(`ElastAlert exited with code ${code}`);
92             this._status = Status.ERROR;
93         }
94         this._process = null;
95
96         this._onExitCallbacks.map(function(onExitCallback) {
97             if (onExitCallback !== null) {
98                 onExitCallback();
99             }
100         });
101     });
102
103     // Set listener for ElastAlert error
104     this._process.on('error', (err) => {
105         logger.error(`ElastAlert error: ${err.toString()}`);
106         this._status = Status.ERROR;
107         this._process = null;

```

```

108     });
109 }
110
111 /**
112  * Stop ElastAlert if it is running.
113  */
114 stop() {
115     if (this._process !== null) {
116         // Stop ElastAlert
117         logger.info(`Stopping ElastAlert (PID: ${this._process.pid})`);
118         this._status = Status.CLOSING;
119         this._process.kill('SIGINT');
120     } else {
121         // Do not do anything if ElastAlert is not running
122         logger.info('ElastAlert is not running');
123     }
124 }
125 }
126
127 EOF

```

Trước khi build lại image mới cần edit lại `Dockerfile` như sau

```

1  $ pwd
2  /opt/elastalert-server-g1
3  $ cat > Dockerfile << EOF
4  FROM python:3.11-alpine3.18 as ea2
5  ARG ELASTALERT_VERSION=2.14.0
6  ENV ELASTALERT_VERSION=${ELASTALERT_VERSION}
7  ARG ELASTALERT_URL=https://github.com/jertel/elastalert2/archive/refs/tags/${ELASTALERT_VERSION}.zip
8  ENV ELASTALERT_URL=${ELASTALERT_URL}
9  ENV ELASTALERT_HOME /opt/elastalert
10
11 WORKDIR /opt
12
13 RUN apk add --update --no-cache wget && \
14     apk add --update --no-cache supervisor && \
15     wget -O elastalert.zip "${ELASTALERT_URL}" && \
16     unzip elastalert.zip && \
17     rm elastalert.zip && \
18     mv e* "${ELASTALERT_HOME}"
19
20 FROM node:16.20.2-alpine3.18 as install
21 ENV PATH /home/node/.local/bin:$PATH
22
23 RUN apk add --update --no-cache \
24     ca-certificates \
25     cargo \
26     curl \
27     gcc \
28     libffi-dev \
29     libmagic \
30     make \
31     musl-dev \
32     openssl \
33     openssl-dev \
34     py3-pip \
35     py3-wheel \
36     python3 \

```

```
37     python3-dev \
38     tzdata \
39     supervisor
40
41 COPY --from=ea2 /opt/elastalert /opt/elastalert
42
43 WORKDIR /opt/elastalert-server
44 COPY . /opt/elastalert-server
45
46 RUN npm install --omit=dev --quiet
47
48 RUN pip3 install --no-cache-dir --upgrade pip==23.3.1
49
50 USER node
51
52 WORKDIR /opt/elastalert
53
54 RUN pip3 install --no-cache-dir cryptography --user
55 RUN pip3 install --no-cache-dir -r requirements.txt --user
56
57 FROM node:16.20.2-alpine3.18
58 LABEL maintainer="John Susek <john@johnsolo.net>"
59 ENV TZ Etc/UTC
60 ENV PATH /home/node/.local/bin:$PATH
61
62 RUN apk add --update --no-cache \
63     ca-certificates \
64     cargo \
65     curl \
66     gcc \
67     libffi-dev \
68     libmagic \
69     make \
70     musl-dev \
71     openssl \
72     openssl-dev \
73     py3-pip \
74     python3 \
75     python3-dev \
76     tzdata \
77     supervisor
78
79 COPY --from=install /opt/elastalert /opt/elastalert
80 COPY --from=install /home/node/.local/lib/python3.11/site-packages /home/node/.local/lib/python3.11/site-packages
81
82 WORKDIR /opt/elastalert-server
83
84 COPY --from=install /opt/elastalert-server ./
85
86 COPY config/elastalert.yaml /opt/elastalert/config.yaml
87 COPY config/config.json config/config.json
88 COPY rule_templates/ /opt/elastalert/rule_templates
89 COPY elastalert_modules/ /opt/elastalert/elastalert_modules
90
91 # Add default rules directory
92 # Set permission as unprivileged user (1000:1000), compatible with Kubernetes
93 RUN mkdir -p /opt/elastalert/rules/ /opt/elastalert/server_data/tests/ \
94     && chown -R node:node /opt
```

```

95
96 RUN mkdir -p /var/log/supervisord \
97     && chown -R node:node /var/log/supervisord
98
99 USER node
100
101 EXPOSE 3030
102
103 WORKDIR /opt/elastalert-server
104
105 ENTRYPOINT ["npm", "start"]
106 EOF

```

Tiếp theo cần build lại docker image

```

1 $ pwd
2 /opt/elastalert-server-g1
3 $ sudo docker build -t elasalert-server-g1:v1 .
4 $ sudo docker image ls
5
6 REPOSITORY          TAG          IMAGE ID       CREATED        SIZE
7 elastalert-server-g1 v1           33ba49089b81  2 weeks ago   1.57GB
8 praecoapp/elastalert-server latest       cd61b812f782  8 weeks ago   1.62GB

```

Sau khi có được image elastalert-server mới, cần chuẩn bị một số file cấu hình cơ bản bao gồm `config.json` `rules` `supervisord` như sau

File `config.json`

```

1 $ cd /opt/elastalert-server-g1
2 $ cat > /opt/elastalert-server-g1/config/config.json << EOF
3 {
4   "appName": "elastalert-server-g1",
5   "port": 3030,
6   "wsport": 3333,
7   "elastalertPath": "/opt/elastalert",
8   "verbose": true,
9   "es_debug": false,
10  "debug": false,
11  "rulesPath": {
12    "relative": true,
13    "path": "/rules"
14  },
15  "templatesPath": {
16    "relative": true,
17    "path": "/rule_templates"
18  },
19  "dataPath": {
20    "relative": true,
21    "path": "/server_data"
22  },
23  "es_host": "192.168.68.141",
24  "es_port": 9200,
25  "es_username": "elastalert_user",
26  "es_password": "ssXbvLhMUjW6Hpahidden",
27  "opensearch_flg": false,
28  "opensearch2_flg": false,
29  "es_ssl": true,
30  "es_ca_certs": "/opt/elastalert/certs/ca.crt",
31  "es_client_cert": "/opt/elastalert/certs/kibana.crt",
32  "es_client_key": "/opt/elastalert/certs/kibana.key",

```

```
33     "writeback_index": "elastalert_g1_status"
34 }
35 EOF
```

ⓘ Lưu ý thay đổi thông tin phù hợp bao gồm: `es_host` `es_port` `es_username` `es_password` `es_ca_certs` `es_client_cert` `es_client_key` và `writeback_index`

Folder `rules` và `supervisord`

```
1 $ pwd
2 /opt/elastalert-server-g1
3 $ mkdir rules
4 $ mkdir rules/1d_rules
5 $ mkdir rules/5m_rules
6 $ mkdir rules/1m_rules
7 $ mkdir supervisord
```

File `supervisord.conf`


```
1 $ pwd
2 /opt/elastalert-server-g1
3 $ cat > config/supervisord.conf << EOF
4 [supervisord]
5 nodaemon=true
6 logfile=/var/log/supervisord/supervisord.log
7 logfile_maxbytes=20MB
8
9 [program:elastalert-run1m]
10 directory=/opt/elastalert
11 command=python3 -m elastalert.elastalert --config /opt/elastalert/supervisord/elastalert-1m.yaml --verbose
12 process_name=elastalert-run1m
13 autorestart=true
14 startsecs=15
15 stopsignal=INT
16 stopasgroup=true
17 killasgroup=true
18 stdout_logfile=/var/log/supervisord/elastalert-run1m_stdout.log
19 stdout_logfile_maxbytes=15MB
20 stderr_logfile=/var/log/supervisord/elastalert-run1m_stderr.log
21 stderr_logfile_maxbytes=15MB
22
23 [program:elastalert-run5m]
24 directory=/opt/elastalert
25 command =python3 -m elastalert.elastalert --config /opt/elastalert/supervisord/elastalert-5m.yaml --verbose
26 process_name=elastalert-run5m
27 autorestart=true
28 startsecs=15
29 stopsignal=INT
30 stopasgroup=true
31 killasgroup=true
32 stdout_logfile=/var/log/supervisord/elastalert-run5m_stdout.log
33 stdout_logfile_maxbytes=15MB
34 stderr_logfile=/var/log/supervisord/elastalert-run5m_stderr.log
35 stderr_logfile_maxbytes=15MB
36
37 [program:elastalert-run1day]
38 directory=/opt/elastalert
39 command =python3 -m elastalert.elastalert --config /opt/elastalert/supervisord/elastalert-1d.yaml --verbose
```



```

40 process_name=elastalert-run1day
41 autorestart=true
42 startsecs=15
43 stopsignal=INT
44 stopasgroup=true
45 killasgroup=true
46 stdout_logfile=/var/log/supervisord/elastalert-run1d_stdout.log
47 stdout_logfile_maxbytes=15MB
48 stderr_logfile=/var/log/supervisord/elastalert-run1d_stderr.log
49 stderr_logfile_maxbytes=15MB
50 EOF

```

 Lưu ý config supervisord trên tạo 3 process thực hiện chạy trên 3 profile. Có thể remove option `--verbose` tại command để không ghi output ra file log

File cấu hình cho elastalert với các profile chạy theo schedule tương ứng như config supervisord.conf

```

1 $ pwd
2 /opt/elastalert-server-g1
3 $ cat > supervisord/elastalert-1m.yaml << EOF
4 # The elasticsearch hostname for metadata writeback
5 # Note that every rule can have its own elasticsearch host
6 es_host: 192.168.68.141
7
8 # The elasticsearch port
9 es_port: 9200
10
11 # This is the folder that contains the rule yaml files
12 # Any .yaml file will be loaded as a rule
13 rules_folder: rules/1m_rules
14
15 # How often ElastAlert will query elasticsearch
16 # The unit can be anything from weeks to seconds
17 run_every:
18   minutes: 1
19
20 # ElastAlert will buffer results from the most recent
21 # period of time, in case some log sources are not in real time
22 buffer_time:
23   minutes: 1
24
25 # Optional URL prefix for elasticsearch
26 #es_url_prefix: elasticsearch
27
28 # Connect with TLS to elasticsearch
29 use_ssl: True
30
31 # Verify TLS certificates
32 # verify_certs: false
33
34 # GET request with body is the default option for Elasticsearch.
35 # If it fails for some reason, you can pass 'GET', 'POST' or 'source'.
36 # See http://elasticsearch-py.readthedocs.io/en/master/connection.html?highlight=send_get_body_as#transport
37 # for details
38 #es_send_get_body_as: GET
39
40 # Option basic-auth username and password for elasticsearch
41 es_username: elastalert_user

```

```

42 es_password: ssXbvLhMUjW6Hpahidden
43
44 # Use SSL authentication with client certificates client_cert must be
45 # a pem file containing both cert and key for client
46 ca_certs: /opt/elastalert/certs/ca.crt
47 client_cert: /opt/elastalert/certs/kibana.crt
48 client_key: /opt/elastalert/certs/kibana.key
49
50 # The index on es_host which is used for metadata storage
51 # This can be a unmapped index, but it is recommended that you run
52 # elastalert-create-index to set a mapping
53 writeback_index: elastalert_g1_status
54
55 # If an alert fails for some reason, ElastAlert will retry
56 # sending the alert until this time period has elapsed
57 alert_time_limit:
58     days: 2
59
60 skip_invalid: True
61 EOF

```

```

1 $ pwd
2 /opt/elastalert-server-g1
3 $ cat > supervisord/elastalert-5m.yaml << EOF
4 # The elasticsearch hostname for metadata writeback
5 # Note that every rule can have its own elasticsearch host
6 es_host: 192.168.68.141
7
8 # The elasticsearch port
9 es_port: 9200
10
11 # This is the folder that contains the rule yaml files
12 # Any .yaml file will be loaded as a rule
13 rules_folder: rules/5m_rules
14
15 # How often ElastAlert will query elasticsearch
16 # The unit can be anything from weeks to seconds
17 run_every:
18     minutes: 5
19
20 # ElastAlert will buffer results from the most recent
21 # period of time, in case some log sources are not in real time
22 buffer_time:
23     minutes: 5
24
25 # Optional URL prefix for elasticsearch
26 #es_url_prefix: elasticsearch
27
28 # Connect with TLS to elasticsearch
29 use_ssl: True
30
31 # Verify TLS certificates
32 # verify_certs: false
33
34 # GET request with body is the default option for Elasticsearch.
35 # If it fails for some reason, you can pass 'GET', 'POST' or 'source'.
36 # See http://elasticsearch-py.readthedocs.io/en/master/connection.html?highlight=send\_get\_body\_as#transport
37 # for details

```

```

38 #es_send_get_body_as: GET
39
40 # Option basic-auth username and password for elasticsearch
41 es_username: elastalert_user
42 es_password: ssXbvLhMUjW6Hpahidden
43
44 # Use SSL authentication with client certificates client_cert must be
45 # a pem file containing both cert and key for client
46 ca_certs: /opt/elastalert/certs/ca.crt
47 client_cert: /opt/elastalert/certs/kibana.crt
48 client_key: /opt/elastalert/certs/kibana.key
49
50 # The index on es_host which is used for metadata storage
51 # This can be a unmapped index, but it is recommended that you run
52 # elastalert-create-index to set a mapping
53 writeback_index: elastalert_g1_status
54
55 # If an alert fails for some reason, ElastAlert will retry
56 # sending the alert until this time period has elapsed
57 alert_time_limit:
58     days: 2
59
60 skip_invalid: True
61 EOF

```

```


1 $ pwd
2 /opt/elastalert-server-g1
3 $ cat > supervisord/elastalert-1d.yaml << EOF
4 # The elasticsearch hostname for metadata writeback
5 # Note that every rule can have its own elasticsearch host
6 es_host: 192.168.68.141
7
8 # The elasticsearch port
9 es_port: 9200
10
11 # This is the folder that contains the rule yaml files
12 # Any .yaml file will be loaded as a rule
13 rules_folder: rules/1d_rules
14
15 # How often ElastAlert will query elasticsearch
16 # The unit can be anything from weeks to seconds
17 run_every:
18     days: 1
19
20 # ElastAlert will buffer results from the most recent
21 # period of time, in case some log sources are not in real time
22 buffer_time:
23     days: 1
24
25 # Optional URL prefix for elasticsearch
26 #es_url_prefix: elasticsearch
27
28 # Connect with TLS to elasticsearch
29 use_ssl: True
30
31 # Verify TLS certificates
32 # verify_certs: false
33

```

```

34 # GET request with body is the default option for Elasticsearch.
35 # If it fails for some reason, you can pass 'GET', 'POST' or 'source'.
36 # See http://elasticsearch-py.readthedocs.io/en/master/connection.html?highlight=send_get_body_as#transport
37 # for details
38 #es_send_get_body_as: GET
39
40 # Option basic-auth username and password for elasticsearch
41 es_username: elastalert_user
42 es_password: ssXbvLhMUjW6Hpahidden
43
44 # Use SSL authentication with client certificates client_cert must be
45 # a pem file containing both cert and key for client
46 ca_certs: /opt/elastalert/certs/ca.crt
47 client_cert: /opt/elastalert/certs/kibana.crt
48 client_key: /opt/elastalert/certs/kibana.key
49
50 # The index on es_host which is used for metadata storage
51 # This can be a unmapped index, but it is recommended that you run
52 # elastalert-create-index to set a mapping
53 writeback_index: elastalert_g1_status
54
55 # If an alert fails for some reason, ElastAlert will retry
56 # sending the alert until this time period has elapsed
57 alert_time_limit:
58     days: 2
59
60 skip_invalid: True
61 EOF

```

 Lưu ý thay đổi thông tin phù hợp gồm `es_host` `es_port` `rules_folder` `es_username` `es_password` `ca_certs` `client_cert` `client_key` và `writeback_index`

 Tùy vào nhu cầu chạy schedule sẽ tạo số lượng config supervisord, elastalert profile khác nhau

Start docker elastalert-server

Để thực hiện thành công start elastalert-server cần kiểm tra lại các config mapping như sau

| Host | Container |
|--------------------------------------------------|-------------------------------------------|
| /opt/elastalert-server-g1/supervisord | /opt/elastalert/supervisor |
| /opt/elastalert-server-g1/config/supervisor.conf | /etc/supervisor/supervisor.conf |
| /opt/elastalert-server-g1/config/config.json | /opt/elastalert-server/config/config.json |
| /opt/elastalert-server-g1/rules | /opt/elastalert/rules |
| /opt/elastalert-server-g1/rule_templates | /opt/elastalert/rule_templates |
| /opt/elastalert-server-g1/certs | /opt/elastalert/certs |
| /opt/elastalert-server-g1/hosts | /etc/hosts |

Tiếp theo thực hiện start docker như sau

```

1 $ sudo docker run -d -p 3030:3030 -p 3333:3333 \
2   -v /opt/elastalert-server-g1/supervisord:/opt/elastalert/supervisord \

```

```

3 -v /opt/elastalert-server-g1/config/config.json:/opt/elastalert-server/config/config.json \
4 -v /opt/elastalert-server-g1/config/supervisor.conf:/etc/supervisor/supervisor.conf \
5 -v /opt/elastalert-server-g1/rules:/opt/elastalert/rules \
6 -v /opt/elastalert-server-g1/elastalert_modules:/opt/elastalert/elastalert_modules \
7 -v /opt/elastalert-server-g1/rule_templates:/opt/elastalert/rule_templates \
8 -v /opt/elastalert-server-g1/certs:/opt/elastalert/certs \
9 -v /opt/elastalert-server-g1/hosts:/etc/hosts \
10 --net="host" \
11 --name elastalert-g1-v1 elastalert-server-g1:v1
12
13 $ sudo docker container ls
14 CONTAINER ID   IMAGE                                COMMAND                  CREATED          STATUS          PORTS          NAMES
15 1733273cedbf   elastalert-server-g1:v1             "npm start"             2 weeks ago     Up 2 weeks     elastalert-g1-v1

```

✓ Kết quả nhận được

```

1 $ curl 192.168.68.139:3030 ; echo
2 {"name":"elastalert-server","port":3030,"version":"20231017"}
3 $ curl 192.168.68.139:3030/status ; echo
4 {"status":"READY"}
5 $ curl 192.168.68.139:3030/rules ; echo
6 {"directories":["1d_rules","1m_rules","5m_rules"],"rules":[]}

```