version: '3.5'

# To Join any other app setup using another network, change name and set external = true

networks:

default:

name: elastic

external: false

# will contain all elasticsearch data.

volumes:

elasticsearch-data:

secrets:

elasticsearch.keystore:

file: ./secrets/keystore/elasticsearch.keystore

elasticsearch.service\_tokens:

file: ./secrets/service\_tokens

elastic.ca:

file: ./secrets/certs/ca/ca.crt

elasticsearch.certificate:

file: ./secrets/certs/elasticsearch/elasticsearch.crt

elasticsearch.key:

file: ./secrets/certs/elasticsearch/elasticsearch.key

kibana.certificate:

file: ./secrets/certs/kibana/kibana.crt

kibana.key:

file: ./secrets/certs/kibana/kibana.key

apm-server.certificate:

file: ./secrets/certs/apm-server/apm-server.crt

apm-server.key:

file: ./secrets/certs/apm-server/apm-server.key

services:

elasticsearch:

image: elastdocker/elasticsearch:${ELK\_VERSION}

build:

context: elasticsearch/

args:

ELK\_VERSION: ${ELK\_VERSION}

restart: unless-stopped

environment:

ELASTIC\_USERNAME: ${ELASTIC\_USERNAME}

ELASTIC\_PASSWORD: ${ELASTIC\_PASSWORD}

ELASTIC\_CLUSTER\_NAME: ${ELASTIC\_CLUSTER\_NAME}

ELASTIC\_NODE\_NAME: ${ELASTIC\_NODE\_NAME}

ELASTIC\_INIT\_MASTER\_NODE: ${ELASTIC\_INIT\_MASTER\_NODE}

ELASTIC\_DISCOVERY\_SEEDS: ${ELASTIC\_DISCOVERY\_SEEDS}

ELASTICSEARCH\_PORT: ${ELASTICSEARCH\_PORT}

ES\_JAVA\_OPTS: "-Xmx${ELASTICSEARCH\_HEAP} -Xms${ELASTICSEARCH\_HEAP} -Des.enforce.bootstrap.checks=true -Dlog4j2.formatMsgNoLookups=true"

bootstrap.memory\_lock: "true"

volumes:

- elasticsearch-data:/usr/share/elasticsearch/data

- ./elasticsearch/config/elasticsearch.yml:/usr/share/elasticsearch/config/elasticsearch.yml

- ./elasticsearch/config/log4j2.properties:/usr/share/elasticsearch/config/log4j2.properties

secrets:

- source: elasticsearch.keystore

target: /usr/share/elasticsearch/config/elasticsearch.keystore

- source: elasticsearch.service\_tokens

target: /usr/share/elasticsearch/config/service\_tokens

- source: elastic.ca

target: /usr/share/elasticsearch/config/certs/ca.crt

- source: elasticsearch.certificate

target: /usr/share/elasticsearch/config/certs/elasticsearch.crt

- source: elasticsearch.key

target: /usr/share/elasticsearch/config/certs/elasticsearch.key

ports:

- "${ELASTICSEARCH\_PORT}:${ELASTICSEARCH\_PORT}"

- "9300:9300"

ulimits:

memlock:

soft: -1

hard: -1

nofile:

soft: 200000

hard: 200000

healthcheck:

test: ["CMD", "sh", "-c", "curl -sf --insecure https://$ELASTIC\_USERNAME:$ELASTIC\_PASSWORD@localhost:$ELASTICSEARCH\_PORT/\_cat/health | grep -ioE 'green|yellow' || echo 'not green/yellow cluster status'"]

logstash:

image: elastdocker/logstash:${ELK\_VERSION}

build:

context: logstash/

args:

ELK\_VERSION: $ELK\_VERSION

restart: unless-stopped

volumes:

- ./logstash/config/logstash.yml:/usr/share/logstash/config/logstash.yml:ro

- ./logstash/config/pipelines.yml:/usr/share/logstash/config/pipelines.yml:ro

- ./logstash/pipeline:/usr/share/logstash/pipeline:ro

secrets:

- source: elastic.ca

target: /certs/ca.crt

environment:

ELASTIC\_USERNAME: ${ELASTIC\_USERNAME}

ELASTIC\_PASSWORD: ${ELASTIC\_PASSWORD}

ELASTICSEARCH\_HOST\_PORT: https://${ELASTICSEARCH\_HOST}:${ELASTICSEARCH\_PORT}

LS\_JAVA\_OPTS: "-Xmx${LOGSTASH\_HEAP} -Xms${LOGSTASH\_HEAP} -Dlog4j2.formatMsgNoLookups=true"

ports:

- "5044:5044"

- "9600:9600"

healthcheck:

test: ["CMD", "curl", "-s" ,"-XGET", "http://127.0.0.1:9600"]

kibana:

image: elastdocker/kibana:${ELK\_VERSION}

build:

context: kibana/

args:

ELK\_VERSION: $ELK\_VERSION

restart: unless-stopped

volumes:

- ./kibana/config/:/usr/share/kibana/config:ro

environment:

ELASTIC\_USERNAME: ${ELASTIC\_USERNAME}

ELASTIC\_PASSWORD: ${ELASTIC\_PASSWORD}

ELASTICSEARCH\_HOST\_PORT: https://${ELASTICSEARCH\_HOST}:${ELASTICSEARCH\_PORT}

KIBANA\_PORT: ${KIBANA\_PORT}

env\_file:

- ./secrets/.env.kibana.token

secrets:

- source: elastic.ca

target: /certs/ca.crt

- source: kibana.certificate

target: /certs/kibana.crt

- source: kibana.key

target: /certs/kibana.key

ports:

- "${KIBANA\_PORT}:${KIBANA\_PORT}"

apm-server:

image: elastdocker/apm-server:${ELK\_VERSION}

build:

context: apm-server/

args:

ELK\_VERSION: $ELK\_VERSION

restart: unless-stopped

ports:

- "8200:8200"

volumes:

- ./apm-server/config/apm-server.yml:/usr/share/apm-server/apm-server.yml:ro

environment:

ELASTIC\_USERNAME: ${ELASTIC\_USERNAME}

ELASTIC\_PASSWORD: ${ELASTIC\_PASSWORD}

ELASTICSEARCH\_HOST\_PORT: https://${ELASTICSEARCH\_HOST}:${ELASTICSEARCH\_PORT}

ELASTIC\_APM\_SECRET\_TOKEN: ${ELASTIC\_APM\_SECRET\_TOKEN}

secrets:

- source: elastic.ca

target: /certs/ca.crt

- source: apm-server.certificate

target: /certs/apm-server.crt

- source: apm-server.key

target: /certs/apm-server.key