Configure Elasticsearch/Kibana for TLS and Authentication

Create Certificates

1. Shutdown elk-cluster
2. Add certificate path to elasticsearch config in docker-compose.yml

```
services:
  elasticsearch:
    volumes:
      - ./certs:/certs
```

3. Start elasticsearch

```
$ docker-compose up -d elasticsearch
```

4. Create certificates

```
$ docker exec -it elk-test-elasticsearch bash

[root@elk-test-elasticsearch elasticsearch]# bin/elasticsearch-certutil
ca -pem -ca-dn "cn=Elastic Stack CA"
This tool assists you in the generation of X.509 certificates and
certificate
signing requests for use with SSL/TLS in the Elastic stack.

[ ... ]

If you elect to generate PEM format certificates (the -pem option),
then the output will
be a zip file containing individual files for the CA certificate and
private key

Please enter the desired output file [elastic-stack-ca.zip]:
[root@elk-test-elasticsearch elasticsearch]# unzip -d /certs/ elastic-
stack-ca.zip
Archive:  elastic-stack-ca.zip
   creating: /certs/ca/
  inflating: /certs/ca/ca.crt
  inflating: /certs/ca/ca.key

[root@elk-test-elasticsearch elasticsearch]# bin/elasticsearch-certutil
http

## Elasticsearch HTTP Certificate Utility

[ ... ]
```

```
## Do you wish to generate a Certificate Signing Request (CSR)?

[ ... ]

Generate a CSR? [y/N]n

## Do you have an existing Certificate Authority (CA) key-pair that you
wish to use to sign your certificate?

[ ... ]

Use an existing CA? [y/N]y

## What is the path to your CA?

CA Path: /certs/ca/ca.crt

## What is the path to your CA key?

/certs/ca/ca.crt appears to be a PEM formatted certificate file.
In order to use it for signing we also need access to the private key
that corresponds to that certificate.

CA Key: /certs/ca/ca.key

For how long should your certificate be valid? [5y]

[ ... ]

Generate a certificate per node? [y/N]y

## What is the name of node #1?

This name will be used as part of the certificate file name, and as a
descriptive name within the certificate.

You can use any descriptive name that you like, but we recommend using
the name
of the Elasticsearch node.

node #1 name: elk-test-elasticsearch

## Which hostnames will be used to connect to elk-test-elasticsearch?

[ ... ]

Enter all the hostnames that you need, one per line.
When you are done, press <ENTER> once more to move on to the next step.

elk-test-elasticsearch
```

```
You entered the following hostnames.

 - elk-test-elasticsearch

Is this correct [Y/n]y

## Which IP addresses will be used to connect to elk-test-
elasticsearch?

Enter all the IP addresses that you need, one per line.
When you are done, press <ENTER> once more to move on to the next step.

[ ... ]

You did not enter any IP addresses.

Is this correct [Y/n]y

## Other certificate options

The generated certificate will have the following additional
configuration
values. These values have been selected based on a combination of the
information you have provided above and secure defaults. You should not
need to
change these values unless you have specific requirements.

Key Name: elk-test-elasticsearch
Subject DN: CN=elk-test-elasticsearch
Key Size: 2048

Do you wish to change any of these options? [y/N]n
Generate additional certificates? [Y/n]n

## What password do you want for your private key(s)?

Your private key(s) will be stored in a PKCS#12 keystore file named
"http.p12".
This type of keystore is always password protected, but it is possible
to use a
blank password.

If you wish to use a blank password, simply press <enter> at the prompt
below.
Provide a password for the "http.p12" file:  [<ENTER> for none]

## Where should we save the generated files?

A number of files will be generated including your private key(s),
public certificate(s), and sample configuration options for Elastic
Stack products.
```

```
These files will be included in a single zip archive.

What filename should be used for the output zip file? [/usr/share/elasticsearch/elasticsearch-ssl-http.zip]

Zip file written to /usr/share/elasticsearch/elasticsearch-ssl-http.zip

[root@elk-test-elasticsearch elasticsearch]# unzip -d /certs/ elasticsearch-ssl-http.zip
Archive:  elasticsearch-ssl-http.zip
   creating: /certs/elasticsearch/
  inflating: /certs/elasticsearch/README.txt
  inflating: /certs/elasticsearch/http.p12
  inflating: /certs/elasticsearch/sample-elasticsearch.yml
   creating: /certs/kibana/
  inflating: /certs/kibana/README.txt
  inflating: /certs/kibana/elasticsearch-ca.pem
  inflating: /certs/kibana/sample-kibana.yml
```

Configure elasticsearch for TLS & X-Pack Security

1. Shutdown elasticsearch

   ```
   $ docker-compose down
   Stopping elk-test-elasticsearch ... done
   Removing elk-test-elasticsearch ... done
   Removing network elk-test_default
   ```

2. Add certificate mount to docker-compose

   ```
   volumes:
     - ./elasticsearch/config/elasticsearch.p12:/usr/share/elasticsearch/config/elasticsearch.p12:ro
   ```

3. Copy P12 file to correct place

   ```
   $ cp certs/elasticsearch/http.p12 elasticsearch/config/elasticsearch.p12
   $ sudo chown 1000:1000 elasticsearch/config/elasticsearch.p12
   $ sudo chmod 600 elasticsearch/config/elasticsearch.p12
   ```

1. Update elasticsearch.yml

   ```
   $ grep xpack elasticsearch/config/elasticsearch.yml
   xpack.security.enabled: true
   xpack.security.http.ssl.enabled: true
   xpack.security.http.ssl.verification_mode: "certificate"
   ```

```
xpack.security.http.ssl.keystore.path: "elasticsearch.p12"
```

2. Start elasticsearch

```
docker-compose up -d elasticsearch
Creating network "elk-test_default" with the default driver
Creating elk-test-elasticsearch ... done
```

3. Create Users

```
$ docker exec -it elk-test-elasticsearch bash
[root@elk-test-elasticsearch elasticsearch]# bin/elasticsearch-setup-passwords auto
Initiating the setup of passwords for reserved users elastic,apm_system,kibana,kibana_system,logstash_system,beats_system,remote_monitoring_user.
The passwords will be randomly generated and printed to the console.
Please confirm that you would like to continue [y/N]y


Changed password for user apm_system
PASSWORD apm_system = ***

Changed password for user kibana_system
PASSWORD kibana_system = ***

Changed password for user kibana
PASSWORD kibana = ***

Changed password for user logstash_system
PASSWORD logstash_system = ***

Changed password for user beats_system
PASSWORD beats_system = ***

Changed password for user remote_monitoring_user
PASSWORD remote_monitoring_user = ***

Changed password for user elastic
PASSWORD elastic = ***
```

Configure Kibana for TLS and User Authentication

1. Copy CA file to correct path

```
$ cp certs/kibana/elasticsearch-ca.pem kibana/config/
```

2. Add CA file mount do docker-compose.yml

```
services:
  kibana:
    volumes:
      - ./kibana/config/elasticsearch-ca.pem:/usr/share/kibana/config/elasticsearch-ca.pem:ro
```

3. Add TLS & authentication information to kibana.yml

   ```
   elasticsearch:
     hosts: [ "https://elk-test-elasticsearch:9200" ]
     ssl.certificateAuthorities: [ "config/elasticsearch-ca.pem" ]
     username: "kibana_system"
     password: "***"
   ```

4. Restart elk-stack

   ```
   $ docker-compose up -d && docker-compose logs -f
   ```

kb, elasticsearch