Для смены сертификатов идущих с демо-инсталляцией необходимо:

1. Воспользоваться ролью "deploy\_opendistro" (<https://devrshb.tech:18443/core-team/ansible/deploy_opendistro>) по деплою всего стека включая генерацию сертификатов.

* сохранить архивом плейбук и распаковать на целевом хосте (или git clone)[](https://devrshb.tech:18443/core-team/planing/devops/uploads/d1cda637f0d57c499b2079dbfd8b7588/image.png)
* перейти в каталог с распакованным архивом, проверить inventory-файл hosts, проверить все переменные
* из корневого каталого плейбука выполнить (пример для среды dtln):

# ansible-playbook deploy\_opendistro.yml -i inventories/dtln/hosts

В рез-те поднимется кластер из ES нод, Kibana и все это на само подписных сертификатах.

1. Если необходимо только сгенерировать сертификаты, то из выше скопированного репозитория воспользоваться таской get\_certs.yml перед этим поправив пути для сгенерированных сертификатов. Выполнить:

# ansible-playbook deploy\_opendistro.yml -i inventories/dtln/hosts –t "deploy\_certs"

1. Скопировать сертификаты на необходимые хосты, привести к виду docker-compose.yml добавив во все разделы с сервисом ES нод:

volumes:

- ./путь\_до\_сертификата/admin.pem:/usr/share/elasticsearch/config/admin.pem

- ./путь\_до\_сертификата/admin.key:/usr/share/elasticsearch/config/admin.key

- ./путь\_до\_сертификата/root-ca.pem:/usr/share/elasticsearch/config/root-ca.pem

- ./путь\_до\_сертификата/odfe-node1-test.key:/usr/share/elasticsearch/config/odfe-node1-test.key

- ./путь\_до\_сертификата/odfe-node1-test.pem:/usr/share/elasticsearch/config/odfe-node1-test.pem

- ./путь\_до\_сертификата/odfe-node1-test.key:/usr/share/elasticsearch/config/odfe-node1-test\_http.key

- ./путь\_до\_сертификата/odfe-node1-test.pem:/usr/share/elasticsearch/config/odfe-node1-test\_http.pem

- ./путь\_до\_конфига/elasticsearch.yml:/usr/share/elasticsearch/config/elasticsearch.yml

- ./config/securityconfig/:/usr/share/elasticsearch/plugins/opendistro\_security/securityconfig/

Соответственно для 2ой ноды наименование сертификатов будет другим.

1. Привеcти к виду docker-compose.yml добавив в сервис с kibana:

environment:

SERVER\_SSL\_ENABLED: "true"

SERVER\_SSL\_KEY: /usr/share/kibana/config/odfe-kibana-test\_http.key

SERVER\_SSL\_CERTIFICATE: /usr/share/kibana/config/odfe-kibana-test\_http.pem

SERVER\_SSL\_CERTIFICATEAUTHORITIES: /usr/share/kibana/config/root-ca.pem

volumes:

- ./config/custom-kibana.yml:/usr/share/kibana/config/kibana.yml

- ./путь\_до\_сертификата/root-ca.pem:/usr/share/kibana/config/root-ca.pem

- ./путь\_до\_сертификата/odfe-kibana-test\_http.pem:/usr/share/kibana/config/odfe-kibana-test\_http.pem

- ./путь\_до\_сертификата/odfe-kibana-test\_http.key:/usr/share/kibana/config/odfe-kibana-test\_http.key

Файл custom-kibana.yml:

server.name: kibana

server.host: "0"

elasticsearch.hosts: https://odfe-node1-test:9200

elasticsearch.ssl.verificationMode: "certificate"

elasticsearch.ssl.certificateAuthorities: ["/usr/share/kibana/config/root-ca.pem"]

elasticsearch.username: kibanaserver

elasticsearch.password: <пароль\_пользователя\_kibanaserver>

elasticsearch.requestHeadersWhitelist: ["securitytenant","Authorization"]

opendistro\_security.multitenancy.enabled: true

opendistro\_security.multitenancy.tenants.preferred: ["Private", "Global"]

opendistro\_security.readonly\_mode.roles: ["kibana\_read\_only"]

# Use this setting if you are running kibana without https

opendistro\_security.cookie.secure: false

newsfeed.enabled: false

telemetry.optIn: false

telemetry.enabled: false

security.showInsecureClusterWarning: false

Что бы параллельно сменить пароль от доступа в Kibana необходимо:

* # docker exec <имя\_контейнера\_любой\_ноды> /bin/sh /usr/share/elasticsearch/plugins/opendistro\_security/tools/hash.sh -p <пароль\_пользователя\_kibanaserver>
* # docker cp <имя\_контейнера\_любой\_ноды>:/usr/share/elasticsearch/plugins/opendistro\_security/securityconfig /opt/opendistro/config/
* скопировать hash полученный при генерации <пароль\_пользователя\_kibanaserver> в конфиг файл /opt/opendistro/config/securityconfig/internal\_users.yml в секцию hash с паролем для пользователя kibanaserver

1. Отредактировать конфиг файл elasticsearch.yml на каждой ноде приведя строки к виду:

opendistro\_security.ssl.transport.enabled: true

opendistro\_security.ssl.transport.pemcert\_filepath: /usr/share/elasticsearch/config/odfe-node1-test.pem

opendistro\_security.ssl.transport.pemkey\_filepath: /usr/share/elasticsearch/config/odfe-node1-test.key

opendistro\_security.ssl.transport.pemtrustedcas\_filepath: /usr/share/elasticsearch/config/root-ca.pem

opendistro\_security.ssl.transport.enforce\_hostname\_verification: false

opendistro\_security.ssl.http.enabled: true

opendistro\_security.ssl.http.pemcert\_filepath: /usr/share/elasticsearch/config/odfe-node1-test\_http.pem

opendistro\_security.ssl.http.pemkey\_filepath: /usr/share/elasticsearch/config/odfe-node1\_test\_http.key

opendistro\_security.ssl.http.pemtrustedcas\_filepath: /usr/share/elasticsearch/config/root-ca.pem

opendistro\_security.allow\_default\_init\_securityindex: true

opendistro\_security.authcz.admin\_dn:

- 'CN=admin.go.rshbank.ru,OU=Ops,O=go.rshbank.ru\, Inc.,DC=go.rshbank.ru'

opendistro\_security.nodes\_dn:

- 'DC=go.rshbank.ru,O=go.rshbank.ru\, Inc.,OU=Ops,CN=odfe-node1-test.go.rshbank.ru'

- 'DC=go.rshbank.ru,O=go.rshbank.ru\, Inc.,OU=Ops,CN=odfe-node2-test.go.rshbank.ru'

Соответственно для остальных нод наименование сертификатов для нод будет другим. В разделе "opendistro\_security.nodes\_dn" указываются все имеющиеся ноды (что бы они знали др о др). Информация для поля opendistro\_security.nodes\_dn и opendistro\_security.authcz.admin\_dn берется из вывода команды:

# openssl x509 -in <путь\_до\_сертификата>/<имя\_сертификата>.pem -text -noout

1. Из каталога с docker-compose.yml выполнить

# docker-compose up -d

1. После настройки сертификатов и запуска Elasticsearch запустить securityadmin.sh, чтобы инициализировать плагин безопасности:

# docker exec <контейнер\_с\_es\_нодой> /bin/sh /usr/share/elasticsearch/plugins/opendistro\_security/tools/securityadmin.sh -cd /usr/share/elasticsearch/plugins/opendistro\_security/securityconfig/ -icl -nhnv -cacert /usr/share/elasticsearch/config/root-ca.pem -cert /usr/share/elasticsearch/config/admin.pem -key /usr/share/elasticsearch/config/admin.key

Дополнительно потребуется инициализация после изменения каких-либо конфигов из каталога /usr/share/elasticsearch/plugins/opendistro\_security/securityconfig