Upgrading Elasticsearch from version 2.x to 5.x (Linux)

Summary

Elasticsearch is a required component of ALM Octane. This document is provided as a service and provides instructions on upgrading Elasticsearch on Red Hat Enterprise Linux / Centos platforms.

Topic

This article provides the steps for upgrading the Elasticsearch database from version 2.x to 5.x. Elasticsearch is a required database component for ALM Octane. Elasticsearch version 5.x is required for ALM Octane 12.55.25 and later.

Overview

The upgrade is divided into three stages:

- **Before upgrading**
  Install the migration plugin and run the cluster checkup to see what you will need to adjust during the upgrade.

- **Upgrade**
  Perform the upgrade.

- **After upgrading**
  Restart the cluster, and resume indexing and searching.

Before upgrading

1. **Install the migration plugin**

   Install the migration plugin with a version corresponding to the version of Elasticsearch currently installed.

   You can install from the web or from a local file.

   Install from web

   ```
   ./bin/plugin install https://github.com/elastic/elasticsearch-migration/releases/download/v2.0.4/elasticsearch-migration-2.0.4.zip
   ```

   Install from a local file:

   ```
   /usr/share/elasticsearch/bin/plugin install file:///root/elasticsearch-migration-2.0.4.zip
   ```

   You do not have to restart or reload the cluster restart or reload is required.

2. **Check findings**

   ```
   http://{hostname}:9200/_plugin/elasticsearch-migration
   ```

Click option 1, Cluster Checkup. Write down the recommendations listed for the settings in **elasticsearch.yml** and plugins.

The tables below list changes that usually have to be made.

Plugins not supported

head	Should be run as standalone server
kopf	cerebro
cloud-aws	discovery-ec2, repository-s3
shield	Part of x-pack
license	Part of x-pack

For detailed instructions, see the Elastic documentation here.

Node settings

discovery.ec2.ping_timeout	Not supported	Remove from yml
cloud.aws.s3.proxy.host	Changed	cloud.aws.protocol: https cloud.aws.proxy.host: <proxy host name>
cloud.aws.s3.proxy.port	Changed	cloud.aws.proxy.port: <port>
bootstrap.mlockall: true	Changed	bootstrap.memory_lock: true
shield.enabled	Renamed	xpack.security.enabled
index.number_of_replicas	Not supported	Remove from config file. Should be defined on index level
index.number_of_shards	Not supported	Remove from config file. Should be defined on index level
node.zone	renamed	node.attr.zone
foreground	Not supported	Remove from config file

3. Back up configuration files

```
cp -R /etc/elasticsearch /tmp
```

4. Check OS definitions

Check that the limits are set correctly.

- **Limits.conf**

  ```
  vi /etc/security/limits.conf
  elasticsearch soft memlock unlimited
  elasticsearch hard memlock unlimited
  elasticsearch soft nproc 4096
  elasticsearch hard nproc 4096
  ```

- **/bin/java**

  Check that **/bin/java** is defined correctly. If not, check softlinks and recreate.

  ```
  ls -ltr /bin/java
  ls -ltr /etc/alternatives/java
  rm -f /etc/alternatives/java
  ln -s /usr/java/openjdk-8u65.0.b17.el6_7/bin/java /etc/alternatives/java
  /bin/java -version
  ```

5. Back up ELS indexes

Take a snapshot backup of the Elasticsearch server indexes: `mqm_*`

6. Download ELS installation

6.1 rpm file

```
wget https://artifacts.elastic.co/downloads/elasticsearch/elasticsearch-5.6.5.rpm
```

6.2 zip file

```
wget https://artifacts.elastic.co/downloads/elasticsearch/elasticsearch-5.6.5.zip
```

7. Prepare for plugin re-installation

Plugins are built for a specific version of Elasticsearch, and therefore must be re-installed each time Elasticsearch is updated.

- Write down all plugins that are installed on all nodes.

  ```
  bin/elasticsearch-plugin list
  ```

- Remove the plugins.

  ```
  bin/elasticsearch-plugin remove [pluginname]
  ```

- If you are using the **shield** plugin, remove the **shield** directory.

  ```
  mv /etc/elasticsearch/shield /tmp
  ```

- For external plugins, like prometheus, download the rpm files of plugins for the correct version of Elasticsearch.

8. Stop ALM Octane servers

Windows installations:

Stop the ALM Octane service.

Linux installations:

```
/opt/octane/wrapper/HPALM stop
```

9. Disable shard allocation before shutting the node

```
curl -XPUT '{hostname}:9200/_cluster/settings?pretty' -H 'Content-Type:
application/json' -d'
{
  "persistent": {
    "cluster.routing.allocation.enable": "none"
  }
}
'
```

10. Perform a synced flush and stop indexing

```
curl -XPOST '{hostname}:9200/_flush/synced?pretty'
```

A synced flush request is a “best effort” operation. It will fail if there are any pending indexing operations, *but it is safe to reissue the request multiple times if necessary.*

11. Shutdown all cluster nodes

```
service elasticsearch stop
```

Upgrade

1. Upgrade all nodes

1.1 Use the rpm downloaded earlier under “Before upgrading” to install the new package.

```
yum localinstall  /tmp/elasticsearch-5.6.5.rpm
```

1.2 Use the zip file downloaded earlier under “Before upgrading” to install the new package.

```
gunzip /tmp/elasticsearch-5.6.5.zip
```

2. Edit /etc/elasticsearch/jvm.options

```
vi /etc/elasticsearch/jvm.options
```

Change from	Change to
-Xms<**XXX**>g	Change the **XXX** to half of memory available on the machine minus 1.
-Xmx<**XXX**>g	Change the **XXX** to half of memory available on the machine minus 1.

-**XX**:CMSInitiatingOccupancyFraction	Change the **XX** to **92**.

3. Edit elasticsearch.yml

Fix settings according to the findings of migration tool you ran earlier (see above).

elasticsearch.yml is usually located in the **/etc/elasticsearch** folder.

```
vi /etc/elasticsearch/Elasticsearch.yml
```

Section	Change type	From	To
#Memory parameter	Change	bootstrap.mlockall: true	bootstrap.memory_lock: true
#X-Pack parameter	Add	NA	xpack.security.enabled: true
#Not supported definitions	Remove	NA	index.number_of_shards: xx **Note**: Mark down this parameters as it will be used later in the index template.
#Not supported definitions	Remove	NA	index.number_of_replicas: xx **Note**: Mark down this parameter as it will be used later in the index template.
#For aws plugin	Change	node.zone: <zone>	node.attr.zone: <zone>
#For aws plugin	Remove	NA	plugin.madatory: cloud-aws
#For aws plugin	Remove	NA	discovery.ec2.ping_timeout: <xx>s
#For linux kernel 6.5	Add	NA	Add: bootstrap.system_call_filter: false
#For linux kernel 6.5	Upgrade	NA	Consider upgrading to Linux 7.

4. Data Path (default path)

If data path was not configured in Elastic 2.4 it uses default path to save the data in that case you will need to configure the new elastic to point to that default data path so the data will be upgraded.

The default path is usually can be found at: /var/lib/Elasticsearch so:

Section	Change type	From	To
#Path.Data	Add	NA	path.data: /var/lib/Elasticsearch

5. **Edit /etc/sysconfig/elasticsearch**

```
vi /etc/sysconfig/elasticsearch
```

Change type	From	To
Remove	NA	ES_HEAP_SIZE=<xxx>g

6. **Edit /usr/lib/systemd/system/elasticsearch.service**

Add this parameter to work with memory locking.

Section	Change type	From	To
# allow memory lock	Add	NA	LimitMEMLOCK=infinity

7. **Re-install plugins**

Use the list of plugins prepared earlier under "Before upgrading" to install new versions.

- For plugins written by Elasticsearch, use the following command:

```
kopfl [plugin-name]
```

  For example, if you use Amazon discovery plugins:

```
./elasticsearch-plugin install discovery-ec2
```

```
./elasticsearch-plugin install repository-s3
```

- For external plugins, use previously-loaded plugin installation files.

```
./elasticsearch-plugin install file://[path to file/file_name]
```

  For example, for Prometheus:

```
./elasticsearch-plugin install file:///tmp/elasticsearch-prometheus-exporter-5.6.5.0.zip
```

- **X-Pack** installation and definition

```
./elasticsearch-plugin install x-pack
```

```
cd /usr/share/elasticsearch
```

  - Create system_key

```
sudo bin/x-pack/syskeygen -> file created on /etc/elasticsearch/x-pack
```

    For cluster configuration, copy the system key file to **all nodes**.

  - Change the owner of the system_key file

```
chown elasticsearch:elasticsearch /etc/elasticsearch/x-pack/system_key
```

- Add admin user for Elasticsearch (this should be done on **all nodes**)

  ```
  sudo bin/x-pack/users useradd -v <username> -r superuser -p <password>
  ```

- Restart the service on **all cluster nodes**

  ```
  systemctl daemon-reload
  ```

  ```
  service elasticsearch restart
  ```

- Update the x-pack license

  ```
  curl -XPUT -u <user> 'http://<host>:<port>/_xpack/license?' -H "Content-Type: application/json" -d @license.json
  ```

  If you get the message "If there are limitations in the license, run with acknowledge=true", run with the syntax below:

  ```
  curl -XPUT -u <user> 'http://<host>:<port>/_xpack/license?acknowledge=true' -H "Content-Type: application/json" -d @/tmp/license.json
  ```

After upgrading

1. **Start the cluster**

   If you have dedicated master nodes, start them first. Dedicated master nodes are nodes with **node.master** set to **true** (the default) and **node.data** set to **false**.

   The number of nodes started should be at least the number defined in **elasticsearch.yml**:

   ```
   discovery.zen.minimum_master_nodes: xxx
   ```

   If not, the recovery will not start until the number of nodes reaches number defined in **discovery.zen.minimum_master_nodes**.

   To start Elasticsearch, run the following on **all cluster nodes**:

   ```
   systemctl daemon-reload
   ```

   ```
   service elasticsearch start
   ```

   Check that nodes respond:

   ```
   curl -XGET '{hostname}:9200/_cat/health?pretty'
   ```

   ```
   curl -XGET '{hostname}:9200/_cat/recovery?pretty'
   ```

2. **Wait for yellow**

   Run the commands below and wait until all primary shards have been recovered, but not all replica shards are allocated. This is to be expected because allocation is still disabled.

   ```
   curl -XGET '{hostname}:9200/_cat/health?pretty'
   ```

   ```
   curl -XGET '{hostname}:9200/_cat/recovery?pretty'
   ```

3. **Re-enable allocation**

   When all nodes joined the cluster and the cluster is yellow, re-enable the allocation.

```
curl -XPUT '{hostname}:9200/_cluster/settings?pretty' -H 'Content-Type:
application/json' -d'
{
  "persistent": {
    "cluster.routing.allocation.enable": "all"
  }
}
'
```

4. **Back up ELS indexes**

   Take a snapshot backup of the Elasticsearch server indexes after upgrading.

```
mqm_*
```

5. **Create index template**

   Replace **num_shards** and **num_replicas** in the template below with the values you removed from the **elasticsearch.yml** file earlier.

```
PUT _template/mqm_index_template
{
 "order": 0,
 "template": "mqm*",
 "settings": {
  "index": {
   "number_of_shards": "num_shards",
   "number_of_replicas": "num_replicas"
  }
 },
 "mappings": {},
 "aliases": {}
}
```

6. **Resume indexing and searching**

   At this point it is safe to resume indexing and searching. Start application servers.

7. **Wait for green**

   All primary and replica shards successfully allocated.

```
curl -XGET '{hostname}:9200/_cat/health?pretty'
curl -XGET '{hostname}:9200/_cat/recovery?pretty'
```