This direction documents the procedures of performing an extraction for an ES cluster, and the usage for the scripts involved. The sample code given in this document is all based on the DEV perf\_1 cluster:

Domain name:

elasticsearch.dc\_perf\_1.us-west-2.migration.dev.oneplatform.build

Security group:

infra\_elasticsearch\_eagan\_dc\_perf\_cluster\_1

**1. Check the health status of target cluster**

We must make sure the target cluster is completely healthy before starting the extraction. Either of the following approaches can be used to check the health status:

1) curl -XGET elasticsearch.dc\_perf\_1.us-west-2.migration.dev.oneplatform.build:9200/\_cluster/health?pretty

2) http://35.164.243.109:9200/\_plugin/head/

**2. Select nodes for storing extracted data**

It is suggested to select 10 data nodes from the cluster to execute the extraction and receive data in parallel in order to save time. It is also suggested to use the nodes having most spare disk space to store the extracted data. The following command can make it easy to do the selection:

curl -XGET elasticsearch.dc\_perf\_1.us-west-2.migration.dev.oneplatform.build:9200/\_cat/nodes?h=h,i,r,m,d | awk '$3 ~ /d/ {print $0}' | sort -k 5 | tail

Once nodes are selected, make sure the python module 'elasticsearch' is installed on them.

**3. Commit updated code to GitHub**

Another preparation that needs to be done before running the extraction is to commit your updated code to GitHub if any update is made. Go to the following repository and upload your files to the master branch:

https://github.com/liusibo/Elasticsearch\_data\_extraction

**4. Jenkins project**

Now we are ready to go to the Jenkins project and fill in the parameter boxes. Open the following page:

http://ec2-35-164-170-131.us-west-2.compute.amazonaws.com:8080/job/elasticsearch-dump-project/

Click on ‘Build with Parameters’, then modify the parameters to meet your requirements.

**5. Node list**

The ‘node\_list’ parameter expects a space separated list of internal IPs for the nodes selected in step 2. This list will be forwarded to a script named ‘dumpnodes.sh’, which does 3 things: list the node list, clear the node list, and add IPs to the list. Its usage is:

dumpnodes.sh –l # list the IP of selected nodes

dumpnodes.sh –d # clear the IPs

dumpnodes.sh <IP1> <IP2> ... # add IPs to the list"

This script actually manipulates a hidden file ‘.nodes’ that located in the same directory with the script. You can change the list by manually editing this file too.

**6. Index list**

A space separated list of indices is provided to the ‘index\_list’ parameter. This list will be forwarded to a script named ‘dumpindices.sh’, which similarly does 3 things: list the index list, clear the index list, and add indices to the list. Its usage is:

dumpnodes.sh –l # list the indices

dumpnodes.sh –d # clear the indices

dumpnodes.sh <IP1> <IP2> ... # add indices to the list"

This script manipulates another hidden file ‘.indices’ that located in the same directory. You can change the index list by manually editing this file too.

**7. URL**

This parameter is the domain name and Elasticsearch port of the target cluster, e.g.

elasticsearch.dc\_perf\_1.us-west-2.migration.dev.oneplatform.build:9200

**8. Data path**

This ‘data\_path’ parameter is the name of the directory that holds the extracted data. You do not need to provide the abstract path because the leading directory components are configured in ‘admindump.sh’ as:

espath="/opt/reuters/data/elasticsearch"

The directory name provided here defaults to ‘data’. It doesn’t make any difference to change it to another name so using the default value is just fine. The extraction program will create a subdirectory under this ‘data’, and name it with the ID of the latest incremental batch that loaded to the cluster. After that the directory structure will be like:

/opt/reuters/data/elasticsearch/data/1481793819/

**9. Options to the entry script**

The entry script is ‘admindump.sh’. It accepts two positional parameters that mentioned above: URL and data path. Following them there are several options:

-a Declare to do full-dump, all contents will be extracted for each index

-f Declare to do ID-dump for source indices, and provide the names of the specific fields that to be extracted. This option must not be used together with –a. Note that the parameters to this option must be separated by comma rather than blank.

-s Declare to do ID-dump for superunif, and provide the names of the specific fields of superunif that to be extracted. This option must not be used together with –a. Note that the parameters to this option must be separated by comma rather than blank.

-b Declare the batch ID. If not given, the ID of the newest incremental batch that loaded to the cluster will be used, which can be found at the end of the following S3 location:

aws s3 ls s3://tr-search-data/1.5/dev/incrementals/

-q Provide a specific query if needed. Note that the query format must meet the following standards:

1) The query must be written in a complete and correct format of Elasticsearch query

2) Quotes must not be put around the entire query body

3) The query body should not have any space in it

4) Quotes inside the query body should not be escaped

-n Provide a specific range of binnum, other than ‘0,65536’. The range must be a pair of non-negative integers and comma separated.

**10. Examples of parameters for Jenkins project**

Here shows how to provide parameters in some typical scenarios:

1) Perhaps the most common case: full-dump, nothing else is restricted. Leave all the parameters with their default value.

2) ID-dump for source indices and superunif, the needed fields are ‘fuid(docid)’ and ‘editions’, nothing else is restricted. Modify the options to:

-f fuid,editions –s docid,editions

3) ID-dump for source indices only, the needed fields are ‘fuid’ and ‘category’, and only those records that have value for ‘category’ are required. Remove ‘superunif’ from the ‘index\_list’ parameter, and modify the options to:

-f fuid,category –q {"query":{"bool":{"must":{"exists":{"field":"category"}}}}}

4) Full-dump for all indices, a batch ID is specified, and a specific range of binnum is provided. Modify the options to:

-a –b 1481793819 –n 58983,65536

**11. Shell commands to be executed for Jenkins project**

In the configuration page of the Jenkins project, make sure the ‘Execute shell’ box is filled with the following commands:

sh dumpnodes.sh -d

sh dumpnodes.sh ${node\_list}

sh dumpindices.sh -d

sh dumpindices.sh ${index\_list}

sudo chmod 400 KP\_infrastructure.pem

sh admindump.sh ${url} ${data\_path} ${parameters}

**12. Kick-off and report**

When you are satisfied with all the parameters and configurations, click on ‘Build’ button to kick off the extraction. The report will be displayed in the ‘console output’ as following:

20161220165549 START DUMP bci

20161220165926 END DUMP bci

20161220165926 START DUMP inspec

20161220170739 END DUMP inspec

20161220170739 START DUMP wos

20161220171558 END DUMP wos

collection real hits dumped hits speed (item/hour) time(hours) success

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

bci 20558483 20558483 341062390.783410 0.060278 Y

inspec 14961498 14961498 109252318.052738 0.136944 Y

wos 50176930 50176930 361997891.783567 0.138611 Y

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

total time : 0.335833

20161220171559 DUMP completed successfully!

This is a sample output of a successful dump. In the case of failure, the report will indicate what indices on which node are failed, and display the following line before exiting the program:

Dump failed, ignore rename, upload and backup processes.

**13. Rename and upload to S3**

If the extraction succeeds, the extracted files will be automatically renamed in order to make them have consecutive sequence suffixes across nodes. Then they will be uploaded to one of the following locations on S3 storage:

Full-dump: s3://tr-ips-ses-data/wos-extractions/wos-full-extractions/

ID-dump: s3://tr-ips-ses-data/wos-extractions/wos-id-extractions/

**14. Backup and cleanup**

The final step is to make backup for the extracted files and clean up the backups from previous extractions. The backup files on each node will eventually be found at:

/opt/reuters/data/elasticsearch/data\_uploaded/

**15. Jenkins server and workspace**

Internal IP: 10.152.11.173

workspace: /work2/tools/jenkins/.jenkins/jobs/elasticsearch-dump-project/workspace/