[**Topcoder ElasticSearch Feeder Service - Way To Populate Challenge-Listing Index v1.0**](https://apps.topcoder.com/forums/?module=Category&categoryID=43128)

# 

# Setup

For the feeder service, run 'mvn clean package' to compile the code and then cd local folder and run './run.sh' to start the services. Before starting the services, you need to first setup the envrionment(such as direct-app, elasticsearch and aws, etc) as mentioned in the Verification\_Topcoder - Elasticsearch Feeder Service For Challenges version 1.0.docx

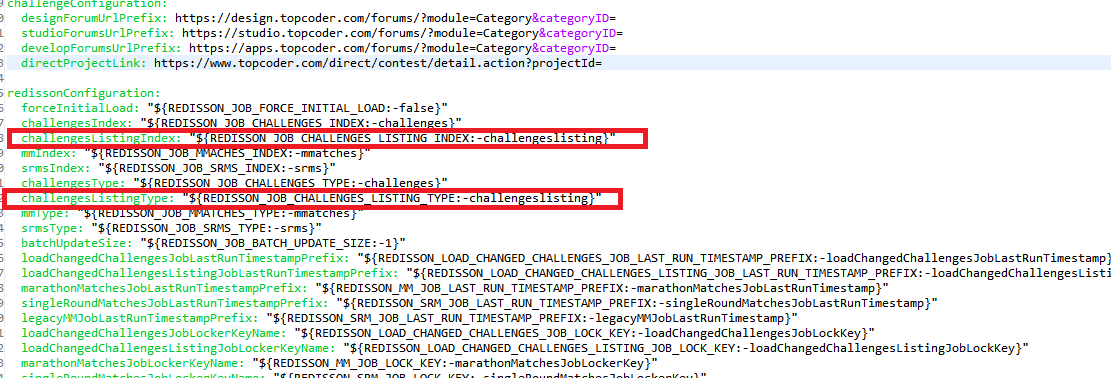
For local deployment, mainly, you need to clone the <https://github.com/appirio-tech/tc-common-tutorials/tree/master/docker/direct-app>, and cd the direct-app directory and run 'docker-compose up tc-direct' to start the tc direct app, and run 'docker-compose up elasticsearch' to start the elasticsearch service. Run 'docker-compose up run-online-review' to start the online review docker.

And also please check Verification\_Topcoder - Elasticsearch Feeder Service For Challenges version 1.0.docx about how to create the challenges(such as assembly challenges) for test purpose.

If the data in the elastics search is not consistent when you push the feeders, you might need to fire the delete http request to delete the mapping first.

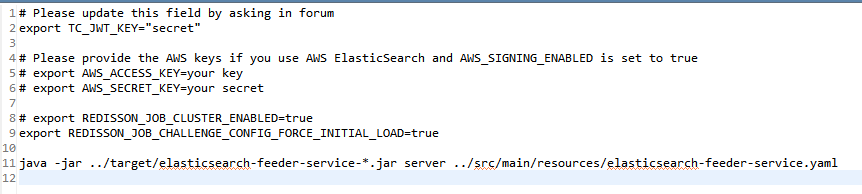
For instance, for the challenges type, fire http request to http://cockpit.cloud.topcoder.com:9200/challenges via delete method.

Check the index and type for the challenges listing in the elasticsearch-feeder-service.yaml configuration file.



In src/main/resources/elasticsearch-feeder-service.yaml, all the config values can be overridden by the environment. But you do not need to change the default ones.

In local/run.sh, if the REDISSON\_JOB\_CLUSTER\_ENABLED is commented out, the service will be run in the standalone model. By default the service will be run against the standalone model.



If REDISSON\_JOB\_CLUSTER\_ENABLED is set to true, the redis clusters for the redisson should be setup.

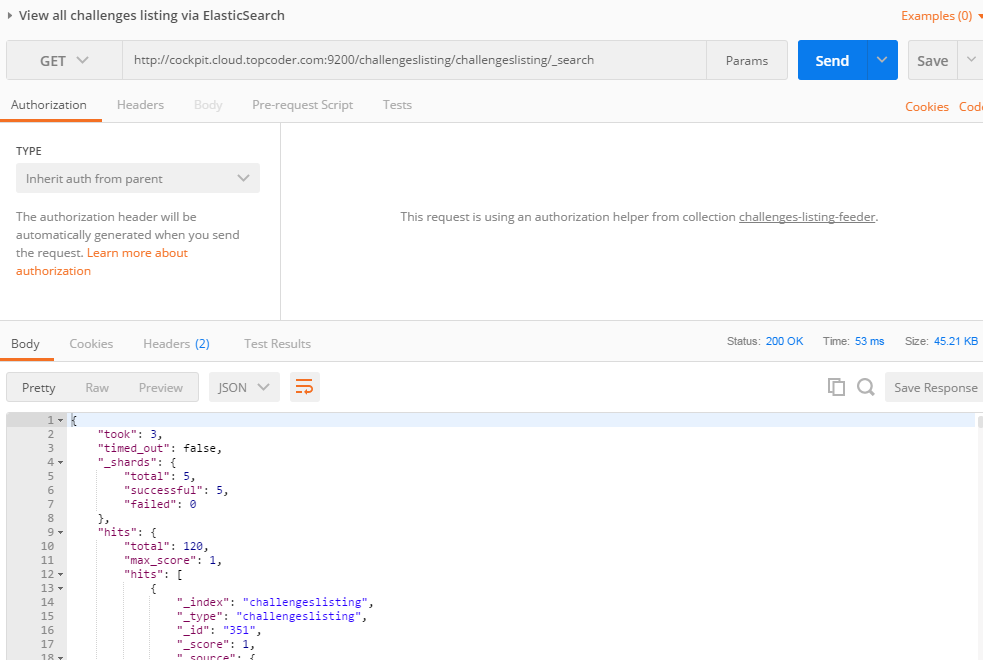
Please refer to Verification\_Topcoder - Create CronJob For Populating Changed Challenges To Elasticsearch version 1.0.docx about how to setup the redis service in the cluster mode.

# Verify

Run local/run.sh, and wait several minutes for the job finishing the load.

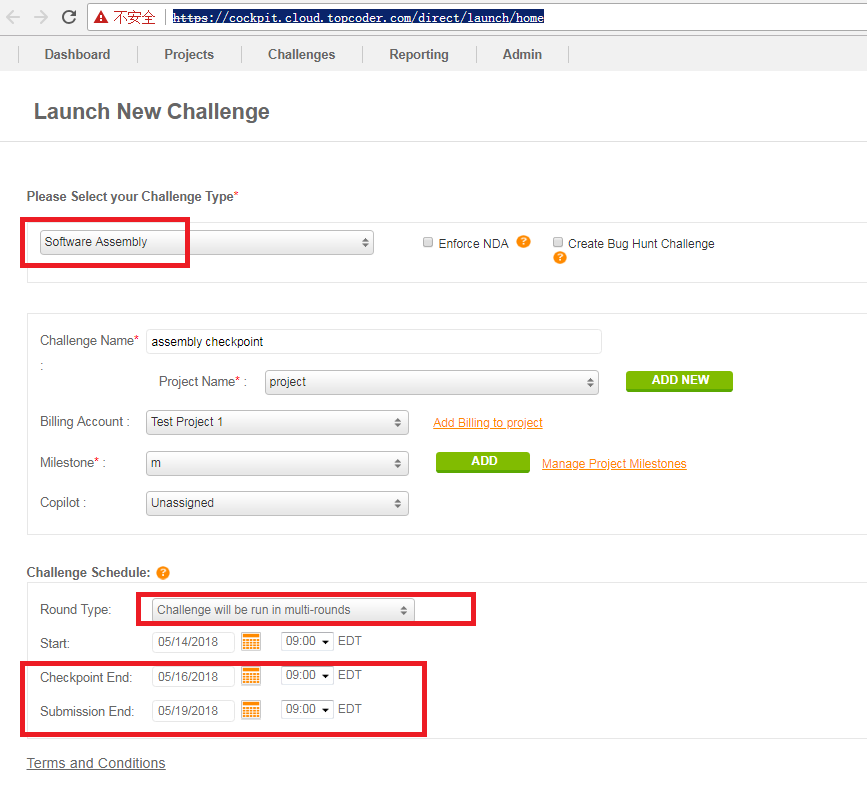
Import the challenges-listing-feeder.postman\_collection.json into the postman.

Check the View all challenges listing via ElasticSearch:



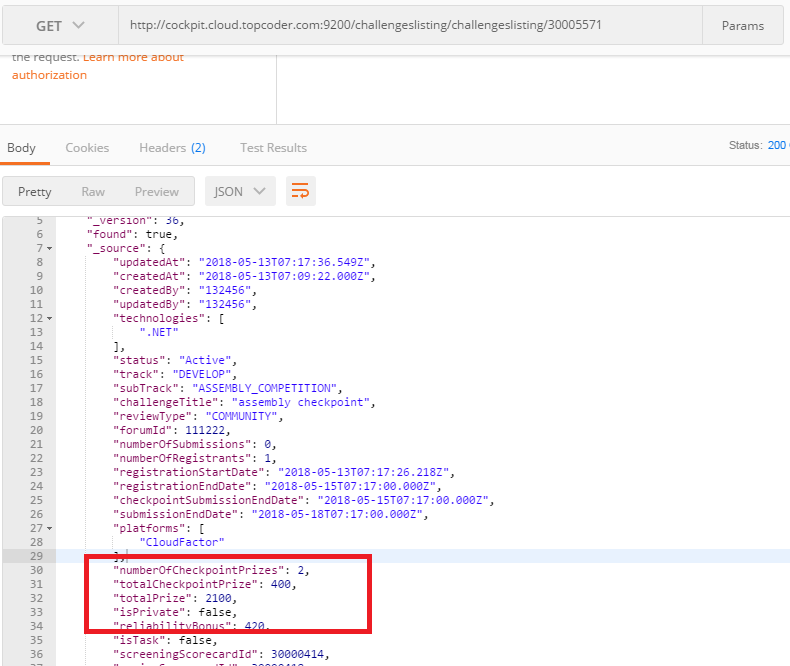
All the data should be loaded.

Create an assembly challenge in the direct app with checkpoints setup. Go to https://cockpit.cloud.topcoder.com/direct/launch/home and set the checkpoint as the following:



The continue to fill any necesary data and start the assembly challenge.

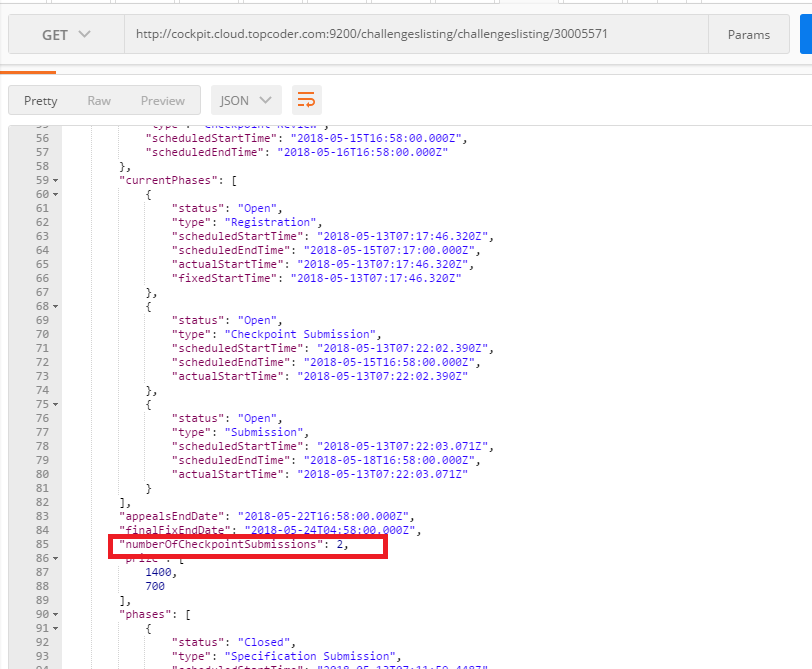
Waits for several minutes for the data to load and check View challenge via ElasticSearch(assembly with checkpoints)



Then go the online review system to start the registration phase, add some user as registrants and then upload the checkpoint submission for the user. You should logout and login as user to upload the checkpoint submissions.

Check Verification\_Topcoder - Elasticsearch Feeder Service For Challenges version 1.0.docx.

Waits for several minutes for the data to load and check View challenge via ElasticSearch(assembly with checkpoints).



The number of the checkpoint submissions should be > 0.

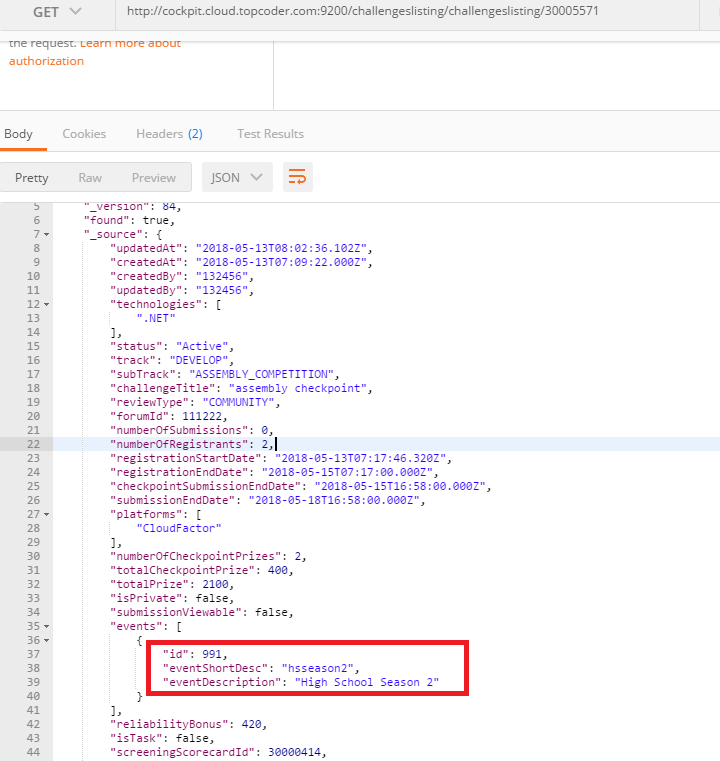
Then select the tcs\_catalog database for informix, and insert some event data for the challenge 30005571(you can change to other event ids from 991):

insert into contest (contest\_id, contest\_name, event\_id) values(1000, "assembly checkpoint", 991);

insert into contest\_project\_xref values(1000, 30005571, current);

You might need to change some data(such as the requirements) in the direct app for 30005571 for the job to reload it.

Waits for several minutes for the data to load and check View challenge via ElasticSearch(assembly with checkpoints)



Assume that you have created and launched an assembly challenge with a winner at the final fix pahse in the tc direct whose challenge id is 30005524.

You might need to change some data(such as the requirements) in the direct app for 30005524 for the job to reload it.

Check View challenge via ElasticSearch(wireframe with winner):

