

## Steps for Collection live indexing

### Collection Services: Tenant and date range wise re-indexing Utility

- Need to have criteria based re-indexing because in any production environments we would not be doing or needing entire data reindexing.
- It would be for either for one tenant or for certain criteria like all payments within a date range.
- Added a plainsearch API endpoints for multiple tenantIds and date range wise reindexing.
- fromDate() and toDate() are to be converted to respective unix timestamps.
- Query parameters are tenantId(can be single or multiple), fromDate and toDate which are to be converted to respective unix timestamps.
- Basically Unix timestamp is a method to track time as running total of seconds. The count starts from Jan 1st 1970, which is considered as Unix Epoch. Unix timestamp is the number of seconds elapsed between a given date and unix epoch.

The part of pushing records to elastic search for DSS and Collection indexes is done through Kafka-connect. The commands to create connector for indexing are as follows:-

#### 1. **DSS Indexing** (to index dss-collection\_v2)

```
curl -X POST http://kafka-connect.kafka-cluster:8083/connectors/ -H 'Content-Type: application/json' -H 'Cookie: SESSIONID=f1349448-761e-4ebc-a8bb-f6799e756185' -H 'Postman-Token: adabf0e8-0599-4ac9-a591-920586ff4d50' -H 'cache-control: no-cache' -d '{"name": "cms-case-es-sink9132","config": {"connector.class": "io.confluent.connect.elasticsearch.ElasticsearchSinkConnector","connection.url": "http://elasticsearch-data-v1.es-cluster:9200","type.name": "general","topics": "egov-dss-ingest-enriched","key.ignore": "false","schema.ignore": true,"value.converter.schemas.enable": false,"key.converter": "org.apache.kafka.connect.storage.StringConverter","value.converter": "org.apache.kafka.connect.json.JsonConverter","transforms": "TopicNameRouter","transforms.TopicNameRouter.type": "org.apache.kafka.connect.transforms.RegexRouter","transforms.TopicNameRouter.regex": ".*","transforms.TopicNameRouter.replacement": "dss-collection_v2","batch.size": 10,"max.buffered.records": 500,"flush.timeout.ms": 600000,"retry.backoff.ms": 5000,"read.timeout.ms": 10000,"linger.ms": 100,"max.in.flight.requests": 2,"errors.log.enable": true,"errors.deadletterqueue.topic.name": "dss-collection_v2-es-failed","tasks.max": 1}}'
```

#### 2. **Collection Indexing** (to index paymentsindex-v1)

```
curl -X POST \
  http://kafka-connect.kafka-cluster:8083/connectors/ \
  H 'Content-Type: application/json' \
  H 'Cookie: SESSIONID=f1349448-761e-4ebc-a8bb-f6799e756185' \
  H 'Postman-Token: adabf0e8-0599-4ac9-a591-920586ff4d50' \
  H 'cache-control: no-cache' \
  d '{
    "name": "cms-case-es-sink9121",
    "config": {
      "connector.class": "io.confluent.connect.elasticsearch.ElasticsearchSinkConnector",
      "connection.url": "http://elasticsearch-data-v1.es-cluster:9200",
      "type.name": "payments",
      "topics": "paymentsindex-v1-enriched",
      "key.ignore": "false",
      "schema.ignore": true,
      "value.converter.schemas.enable": false,
      "key.converter": "org.apache.kafka.connect.storage.StringConverter",
      "value.converter": "org.apache.kafka.connect.json.JsonConverter",
      "transforms": "TopicNameRouter",
      "transforms.TopicNameRouter.type":
"org.apache.kafka.connect.transforms.RegexRouter",
      "transforms.TopicNameRouter.regex": ".*",
      "transforms.TopicNameRouter.replacement": "paymentsindex-v1",
      "batch.size": 10,
      "max.buffered.records": 500,
      "flush.timeout.ms": 600000,
      "retry.backoff.ms": 5000,
      "read.timeout.ms": 10000,
      "linger.ms": 100,
      "max.in.flight.requests": 2,
      "errors.log.enable": true,
      "errors.deadletterqueue.topic.name": "paymentsindex-v1-es-failed",
      "tasks.max": 1
    }
  }'
```