

# Fabric Service Setup

Global Rollout - 2018

Exported on 10/12/2018

## Table of Contents

- Clone the repository
  - **git clone** [git@bitbucket.org<sup>1</sup>:seaecom/fabric-service.git](https://bitbucket.org/1:seaecom/fabric-service.git)
- Cockroach DB Setup:
  - brew install cockroach
  - Start the cockroach db (please note: always start cockroach db from the same directory, as the data will be stored in ./cockroach-data by default)

```
cd <path to where you want the cockroachdb subdirectory to be created>
cockroach start --insecure

// execute the below when another service is already running in the port 8080
cockroach start --insecure --host=localhost --http-port=8081
```

- Use Postico to connect with cockroach
  - **New Favorite** → **Host: localhost, Port: 26257, User: root** → **Connect**
- Execute all the queries present under - ***fabric-service/fabric-service-cockroachdb.sql***

```
cockroach sql --insecure --user=root --port=26257 < ./fabric-service/fabric-service-cockroachdb.sql
```

- **Add the following line to /etc/hosts file**

```
127.0.0.1 cockroachdb
```

- **Clone the global boot config in your repo:**
  - **git clone** [git@bitbucket.org<sup>2</sup>:seaecom/boot-config.git](https://bitbucket.org/2:seaecom/boot-config.git)
- **Run the fabric-service using the above boot config**
  - Before running the fabric service using bootConfig, install dependencies using following command

```
cd <path to root of local fabric service repo>
npm install
```

```
node server.js --bootConfig='../boot-config/config.json'
```

- **Postman Collection**
  - Import the below url in postman and do the 2 API calls mentioned below

 <https://www.getpostman.com/collections/e278da87a46db05a2b7e>

- **Populate the tenant level config using the API**
  - **POST /v1/\_fabric/tenants/{tenant\_id}/config**
    - **tenant\_id**: 3 character ISO country code (See the complete list here: <https://confluence.samsungmtv.com/display/GR2018/Globalization+V4>)

<sup>1</sup> <mailto:git@bitbucket.org>

<sup>2</sup> <mailto:git@bitbucket.org>

**Sample Curl of tenant 'IND'**

```

curl -X POST \
  http://localhost:5099/v1/_fabric/tenants/IND/config \
  -H 'Cache-Control: no-cache' \
  -H 'Content-Type: application/json' \
  -H 'Postman-Token: aff72579-b61c-8ae9-efdc-0172721d4dc1' \
  -H 'x-ecom-app-id: fabric-service' \
  -H 'x-ecom-app-secret: app_secret' \
  -H 'x-ecom-ops-internal-credentials: {"email": "vijayganesh@gmail.com"}' \
  -d '{
    "basic_service_configs": [
      {
        "name": "shopping-carts-ind",
        "server": {
          "instances": [
            {
              "label": "public",
              "port": "4988"
            },
            {
              "label": "private",
              "port": "5088"
            }
          ],
          "logging_file_path": "/tmp"
        }
      },
      {
        "keys": {
          "fulfillment": {
            "appId": "ops-tools",
            "appSecret": "ops-tools"
          },
          "orders": {
            "appId": "ops-tools",
            "appSecret": "ops-tools"
          },
          "pricing": {
            "appId": "shopping-carts-v3",
            "appSecret": "FiDBnIkqd0rcbdlacaddgngFafeFhNdk"
          }
        },
        "name": "ops-web-server-ind",
        "server": {
          "instances": [
            {
              "label": "public",
              "port": "4966"
            },
            {
              "label": "private",
              "port": "5066"
            }
          ]
        }
      }
    ]
  }'

```

```

    }
  ],
  "logging_file_path": "/tmp"
}
},
{
  "ecom-ops-gateway": {
    "rules": [
      {
        "method": "get",
        "permissions": [
          "fabric.write"
        ],
        "urlPattern": "app/fabric/tenants/new"
      },
      {
        "method": "get",
        "permissions": [
          "fabric.write"
        ],
        "urlPattern": "app/fabric/tenants/[0-9a-zA-Z]+/edit"
      },
      {
        "method": "get",
        "permissions": [
          "fabric.write"
        ],
        "urlPattern": "app/fabric/tenants/[0-9a-zA-Z]+/services/new"
      },
      {
        "method": "get",
        "permissions": [
          "fabric.write"
        ],
        "urlPattern": "app/fabric/tenants/[0-9a-zA-Z]+/services/[0-9a-zA-Z]+/
edit"
      },
      {
        "method": "put",
        "permissions": [
          "fabric.write"
        ],
        "urlPattern": "api/v1/_fabric/tenants/[0-9a-zA-Z]+/config"
      },
      {
        "method": "post",
        "permissions": [
          "fabric.write"
        ],
        "urlPattern": "api/v1/_fabric/tenants/[0-9a-zA-Z]+/config"
      },
      {
        "method": "put",
        "permissions": [

```

```


        "fabric.write"
    ],
    "urlPattern": "api/v1/_fabric/tenants/[0-9a-zA-Z]+/services/[0-9a-zA-Z]+/service-config"
  },
  {
    "method": "post",
    "permissions": [
      "fabric.write"
    ],
    "urlPattern": "api/v1/_fabric/tenants/[0-9a-zA-Z]+/services/[0-9a-zA-Z]+/service-config"
  }
]
},
"name": "ops-gateway-ind",
"server": {
  "instances": [
    {
      "label": "public",
      "port": "4977"
    },
    {
      "label": "private",
      "port": "5077"
    }
  ],
  "logging_file_path": "/tmp"
},
{
  "name": "ops-user-service-ind",
  "server": {
    "instances": [
      {
        "label": "public",
        "port": "4955"
      },
      {
        "label": "private",
        "port": "5055"
      }
    ],
    "logging_file_path": "/tmp"
  },
  "sso": {
    "apiBaseUrl": "http://samsunglocal.samsung.com:8111/sso/apiservices"
  }
},
"config_db": {
  "database": "indconfigs",
  "host": "cockroachdb",
  "port": "26257",

```

```

    "ssl": {
      "cert": "/certs/client.ecomdba.crt",
      "key": "/certs/client.ecomdba.key",
      "rejectUnauthorized": false
    },
    "user": "ecomdba"
  },
  "schema_version": "v4.0.0",
  "tenant_level_config": {
    "allowed_locales": [
      "en-in",
      "en-us"
    ],
    "default_locale": "en-in",
    "service_nginx_base_url": "ind-qa1-api.seasam.org"
  }
}'

```

- Above properties explained: (should follow **TenantConfigSchema.js** present in fabric-service)
  - **config\_db**: Consists of the tenant db name, host and port. Db name should be in the format - "{tenant\_id}configs". If the tenant is COLUMBIA, then db is "COLCONFIGS".
  - **basic\_service\_configs**: Array - Each value of the array contains the basic service config of a service. This data can be populated with the help of config.json file present in the service repo.
- **Note**: Configs can be updated using the PUT request with same settings.
-  Only during the first call, POST method is required, for adding the basic\_service\_configs of other services, edit the basic\_service\_configs array and finally make a PUT request to update
- **Populate Service level config using the API. This is currently stored in rethinkdb with values from db-example-config.json.**
  - **POST /v1/\_fabric/tenants/{tenant\_id}/services/{service\_name}/service-config**
    - **tenant\_id**: 3 character ISO country code (See the complete list here: <https://confluence.samsungmtv.com/display/GR2018/Globalization+V4>)
    - **service\_name**: Service repository name (name in package.json)

**Sample curl for inserting shopping-carts-ind config**

```
curl -X POST \
  http://localhost:5099/v1/_fabric/tenants/IND/services/shopping-carts-ind/service-
  config \
  -H 'Cache-Control: no-cache' \
  -H 'Content-Type: application/json' \
  -H 'Postman-Token: 3e820a5b-f331-beee-5fdc-761f320397b7' \
  -H 'x-ecom-app-id: fabric-service' \
  -H 'x-ecom-app-secret: app_secret' \
  -H 'x-ecom-ops-internal-credentials: {"email": "vijayganesh@gmail.com"}' \
  -d '{
    "app_config": {
      "cache": {
        "cachingDisabled": false,
        "db": 5,
        "ttl": 86400
      },
      "cockroachdb": {
        "connection": {
          "database": "ind_shoppingcarts",
          "host": "cockroachdb",
          "port": "26257",
          "ssl": {
            "cert": "/certs/client.ecomdba.crt",
            "key": "/certs/client.ecomdba.key",
            "rejectUnauthorized": false
          },
          "user": "ecomdba"
        },
        "maxPoolSize": 10,
        "minPoolSize": 0
      },
      "elasticsearch": {
        "host": "localhost",
        "port": "9200"
      },
      "po_id": {
        "B2B": {
          "allowedCharacters": "23456789ABCDEFGHIJKLMNOPQRSTUVWXYZ",
          "generatorlength": 8,
          "prefix": "1",
          "suffix": "-IN"
        },
        "B2C": {
          "allowedCharacters": "23456789ABCDEFGHIJKLMNOPQRSTUVWXYZ",
          "generatorlength": 8,
          "prefix": "2",
          "suffix": "-IN"
        },
        "B2E": {
          "allowedCharacters": "23456789ABCDEFGHIJKLMNOPQRSTUVWXYZ",
          "generatorlength": 8,
```





```

        "prefix": "3",
        "suffix": "-IN"
    },
    },
    "redis": {
        "host": "localhost",
        "maxReconnectWaitTime": 1000,
        "port": "6379",
        "totalRetryTimeInSeconds": 1000
    },
    "requestTimeout": 120000
},
"schema_version": "v4.0.0",
"service_name": "shopping-carts-ind"
}'

```

**The above contains only minimal settings, copy the whole db-example-config.json and post it.**

- **Note: Configs can be updated using the PUT request with same settings.**
- **Boot config setup**
  -  Clone the IND boot config repo in your local: `git clone git@bitbucket.org3:seacom/boot-config-ind.git`
  - **Run your service using the above bootconfig and it should fetch the config from fabric-service without any error.**
    -  `node server.js --bootConfig='../boot-config-ind/config.json'`

#### NOTE:

**If you receive the errors during your service startup similar:**

**`There was a problem (Error: write after end) in server-http and it has been destroyed.`**

**Then it may mean that your log file Path is not present. Please correct that in the tenant\_configs DB → Configs Table → Server Hash → logging\_file\_path KEY . Point it to a correct folder.**

**It may also mean that permissions are set correctly and you may be able to fix the issue by doing something like:**

```

cd /tmp
chmod -R 777 fabric/
chmod -R 777 IND/

```

---

<sup>3</sup> <mailto:git@bitbucket.org>