Installing Datamart in New Environments

Purpose

This document details the necessary actions by both the Operations (Ops) and Platform teams in the installation of datamart in a new environment.

Procedure

1. **Ops team sets up inbound routes, vpc rules, external load balancers, target groups, and web proxies.**

2. **Ops team creates build machine EC2 instance as detailed here:**

   https://confluencesw.t-mobile.com/display/PETD/Launching+EC2+Instances

3. **Platform team sets up build machine for all developers**:

   ```
   install JDK 1.8
   install maven 3.*
   install ruby 2.6
   ```

4. **Each Platform developer sets up their own environment on the build machine**:

   ```
   cd ~
   gem install net-ssh
   gem install net-scp
   git clone the com.denaliai.fw repository
   cd com.denaliai.fw
   mvn clean install -Dskip.unit.tests
   git clone the datamart repository
   cd datamart
   git checkout <current datamart branch>
   git pull
   run a build_all.sh script
   ```

5. **Optionally, each Platform developer sets up a build for eventlog ReactUI**:

   ```
   cd ~
   curl -o- https://raw.githubusercontent.com/nvm-
   sh/nvm/v0.39.7/install.sh | bash
   nvm install node
   nvm install-latest-npm
   cd ~/datamart/datamart-federatedsearch/device-ui
   npm install react
   ```

6. **Ops team creates VerneMQ PostgreSQL cluster**.

7. **Ops team creates VerneMQ device and internal broker(s)**.

8. **Platform team prepares VerneMQ PostgreSQL DB**:
    8.1. Create a VerneMQ user for use by container-api for deleting/adding credentials and VerneMQ for querying.
    8.2. Create PostgreSQL DB and table as described by https://docs.vernemq.com/configuring-vernemq/db-auth.
    8.3. Give permissions to the VerneMQ user for DB and tables.
    8.4. Create VerneMQ credentials for devicecontroller, speedtest, MQTT-distributor, and USP-distributor as documented by https://docs.vernemq.com/configuring-vernemq/db-auth with publish and subscribe ACLs used in other environments.

9. **Ops team updates VerneMQ config for PostgreSQL**:
    9.1. Update vernemq.conf for PostgreSQL connection information as described by https://docs.vernemq.com/configuring-vernemq/db-auth.

10. **Ops team creates Kafka cluster as detailed here:**

    https://confluencesw.t-mobile.com/display/PETD/Deploying+Kafka+Managed+Service

11. **Platform team installs Kafka tool on the build machine**:

```
wget https://dlcdn.apache.org/kafka/3.6.1/kafka_2.13-3.6.1.tgz
tar -xzf kafka_2.13-3.6.1.tgz
rm kafka_2.13-3.6.1.tgz
```

12. **Platform team creates Kafka topics**:

```
bin/kafka-topics.sh --bootstrap-server b-
1.devllkmsk.0gutzw.c5.kafka.us-west-2.amazonaws.com:9092 --create
--topic "MQTTDistributor" --partitions 4 --replication-factor 1 -
-config "message.timestamp.type=CreateTime" --config
"retention.ms=28800000" --config "segment.bytes=26214400"

bin/kafka-topics.sh --bootstrap-server b-
1.devllkmsk.0gutzw.c5.kafka.us-west-2.amazonaws.com:9092 --create
--topic "ACSDistributor"  --partitions 4 --replication-factor 1 -
-config "message.timestamp.type=CreateTime" --config
"retention.ms=28800000" --config "segment.bytes=26214400"

bin/kafka-topics.sh --bootstrap-server b-
1.devllkmsk.0gutzw.c5.kafka.us-west-2.amazonaws.com:9092 --create
--topic "DeviceEvents"    --partitions 4 --replication-factor 1 -
-config "message.timestamp.type=CreateTime" --config
"retention.ms=28800000" --config "segment.bytes=26214400"
```

13. **Ops team sets up remaining external services and instances**:
    - 13.1. MongoDB for silo 1 is created and running.
    - 13.2. Redis cluster for Tower lookup created: https://confluencesw.t-mobile.com/display/PETD/Redis+cluster+for+tower+data
    - 13.3. 3 EC2 instances for silo 1 is created and running.
    - 13.4. 1 EC2 instance for federatedsearch, configservice, camp-api, container-api, and distributors is created and running.
    - 13.5. Redshift cluster is created and configured.
    - 13.6. Redshift schema is created.
    - 13.7. S3 bucket is created and permissions are configured: https://confluencesw.t-mobile.com/display/PETD/Creating+S3+Raw+Message+Buckets
    - 13.8. Athena tables are created and configured.
    - 13.9. Redshift permissions to S3 bucket are configured.

14. **Platform team prepares environment config**:
    - 14.1. Copy `<ARTIFACTS>/5.20240104/stage/*` to `<ARTIFACTS>/5.20240104/tmo-dev`
    - 14.2. Update `<ARTIFACTS>/5.20240104/tmo-dev/env.json` with:
        - All VerneMQ credentials created earlier for devicecontroller, speedtest, MQTT-distributor and USP distributor
        - Incognito rest api username and password
        - PostgreSQL credentials for container-api
        - Any updates needed for environmental differences
    - 14.3. Update `<ARTIFACTS>/5.20240104/tmo-dev/global-siloconfig.json` with:
        - MongoDB URL
        - URLs for services
        - Any updates needed for environmental differences
    - 14.4. Add environment files in deploy/scripts (copied and modified from stage versions):
        - `deploy/scripts/env_tmo_dev.rb`
        - `deploy/scripts/release_tmo_dev.r`
        - `deploy/scripts/commands_tmo_dev.rb`
    - 14.5. Copy `deploy/scripts/service-file/stage/*` to `deploy/scripts/service-file/tmo-dev`
    - 14.6. Copy `deploy/scripts/service-artifacts/stage/config/device-types.json` to `deploy/scripts/service-artifacts/tmo-dev/config`
    - 14.7. Copy `deploy/scripts/service-artifacts/stage/config/message-types.json` to `deploy/scripts/service-artifacts/tmo-dev/config`
    - 14.8. Build Datamart via developer maintained shell script:
        - Run: `ruby deploy/scripts/copy_jars.rb`

15. **Platform team prepares EC2 instances for silo and federatedsearch from build machine**:
    - Run: `ruby deploy/scripts/machine_prep.rb <ip-address>`

16. **Platform team deploys to shared machine from build machine**:
    - 16.1. Deploy global-siloconfig.properties:

- Run: `ruby deploy/scripts/release_tmo_dev.rb --federatedsearch --update-global-siloconfig`

16.2. Deploy configservice:

- Copy `deploy/scripts/script-artifacts/stage/datamart-configservice.properties` to `deploy/scripts/script-artifacts/tmo-dev`
- Modify properties file as desired for the environment:
  - Run: `ruby deploy/scripts/release_tmo_dev.rb --federatedsearch --update configservice`

16.3. Deploy container-api:

- Copy `deploy/scripts/service-artifacts/stage/config/container-api-*` to `deploy/scripts/service-artifacts/tmo-dev/config`
- Copy `deploy/scripts/script-artifacts/stage/datamart-container-api.properties` to `deploy/scripts/script-artifacts/tmo-dev`
- Modify properties file as desired for the environment:
  - Run: `ruby deploy/scripts/release_tmo_dev.rb --container-api --update container_api_1`

16.4. Deploy MQTT-Distributor:

- Copy `deploy/scripts/service-artifacts/stage/config/mqtt-distributor-*` to `deploy/scripts/service-artifacts/tmo-dev/config`
- Copy `deploy/scripts/script-artifacts/stage/mqtt-distributor.properties` to `deploy/scripts/script-artifacts/tmo-dev`
- Modify properties file as desired for the environment:
  - Run: `ruby deploy/scripts/release_tmo_dev.rb --distributors --update mqtt_distributor_1`

16.5. Deploy USP-Distributor:

- Copy `deploy/scripts/service-artifacts/stage/config/usp-distributor-*` to `deploy/scripts/service-artifacts/tmo-dev/config`
- Copy `deploy/scripts/script-artifacts/stage/usp-distributor.properties` to `deploy/scripts/script-artifacts/tmo-dev`
- Modify properties file as desired for the environment:
  - Run: `ruby deploy/scripts/release_tmo_dev.rb --distributors --update usp_distributor_1`

17. **Platform team deploys to silo instances from build machine**:

17.1. Deploy global-siloconfig.properties:

- Run `ruby deploy/scripts/release_tmo_dev.rb --silo --update-global-siloconfig`

17.2. Deploy collection service:

- Copy `deploy/scripts/script-artifacts/stage/datamart-collection.properties` to `deploy/scripts/script-artifacts/tmo-dev`

- Modify properties file as desired for the environment:
    - Run `ruby deploy/scripts/release_tmo_dev.rb --silo --update collection`

17.3. Deploy aggregator service:

- Copy `deploy/scripts/script-artifacts/stage/datamart-aggregator.properties` to `deploy/scripts/script-artifacts/tmo-dev`
- Modify properties file as desired for the environment:
    - Run `ruby deploy/scripts/release_tmo_dev.rb --silo --update aggregator_1`

17.4. Deploy host-aggregator service:

- Copy `deploy/scripts/service-artifacts/stage/config/host-aggregator-*` to `deploy/scripts/service-artifacts/tmo-dev/config`
- Copy `deploy/scripts/script-artifacts/stage/datamart-host-aggregator.properties` to `deploy/scripts/script-artifacts/tmo-dev`
- Modify properties file as desired for the environment:
    - Run `ruby deploy/scripts/release_tmo_dev.rb --silo --update host_aggregator_1`

17.5. Deploy cacheservice service:

- Copy `deploy/scripts/script-artifacts/stage/datamart-cacheservice.properties` to `deploy/scripts/script-artifacts/tmo-dev`
- Modify properties file as desired for the environment:
    - Run `ruby deploy/scripts/release_tmo_dev.rb --silo --update cacheservice_1`

17.6. Deploy mqttcache service:

- Copy `deploy/scripts/service-artifacts/stage/config/mqttcache-*` to `deploy/scripts/service-artifacts/tmo-dev/config`
- Copy `deploy/scripts/script-artifacts/stage/datamart-mqttcache.properties` to `deploy/scripts/script-artifacts/tmo-dev`
- Modify properties file as desired for the environment:
    - Run `ruby deploy/scripts/release_tmo_dev.rb --silo --update mqttcache_1`

17.7. Deploy devicecontroller service:

- Copy `deploy/scripts/script-artifacts/stage/datamart-devicecontroller.properties` to `deploy/scripts/script-artifacts/tmo-dev`
- Modify properties file as desired for the environment:
    - Run `ruby deploy/scripts/release_tmo_dev.rb --silo --update devicecontroller`

17.8. Deploy event-actions service:

- Copy `deploy/scripts/script-artifacts/stage/datamart-event-actions.properties` to `deploy/scripts/script-artifacts/tmo-dev`

- Modify properties file as desired for the environment:
    - Run `ruby deploy/scripts/release_tmo_dev.rb --silo --update event_actions`

17.9. Deploy dwh-extractor service:

- Copy `deploy/scripts/script-artifacts/stage/datamart-dwh-extractor.properties` to `deploy/scripts/script-artifacts/tmo-dev`
- Modify properties file as desired for the environment:
    - Run `ruby deploy/scripts/release_tmo_dev.rb --silo --update dwh_extractor`

17.10. Deploy rawmessagestore service:

- Copy `deploy/scripts/service-artifacts/stage/config/rawmessagestore-*` to `deploy/scripts/service-artifacts/tmo-dev/config`
- Copy `deploy/scripts/script-artifacts/stage/datamart-rawmessagestore.properties` to `deploy/scripts/script-artifacts/tmo-dev`
- Modify properties file as desired for the environment:
    - Run `ruby deploy/scripts/release_tmo_dev.rb --silo --update rawmessagestore`

17.11. Deploy usp-rawvaluesstore service:

- Copy `deploy/scripts/service-artifacts/stage/config/usp-rawvaluesstore-*` to `deploy/scripts/service-artifacts/tmo-dev/config`
- Copy `deploy/scripts/script-artifacts/stage/datamart-usp-rawvaluesstore.properties` to `deploy/scripts/script-artifacts/tmo-dev`
- Modify properties file as desired for the environment:
    - Run `ruby deploy/scripts/release_tmo_dev.rb --silo --update usp_rawvaluesstore`

18. **Platform team deploys additional components:**

18.1. Deploy federatedsearch service:

- Copy `deploy/scripts/script-artifacts/stage/datamart-federatedsearch.properties` to `deploy/scripts/script-artifacts/tmo-dev`
- Modify properties file as desired for the environment:
    - Run `ruby deploy/scripts/release_tmo_dev.rb --federatedsearch --update federatedsearch_1 --no-start`

18.2. Deploy camp-api service:

- Copy `deploy/scripts/script-artifacts/stage/datamart-camp-api.properties` to `deploy/scripts/script-artifacts/tmo-dev`
- Modify properties file as desired for the environment**:**
    - Run `ruby deploy/scripts/release_tmo_dev.rb --federatedsearch --update camp_api_1 --no-start`

18.3. Manually on federatedsearch machine:

```
mkdir /opt/datamart/federatedsearch/config-strata
mkdir -p /opt/datamart/federatedsearch/www/eventlog
systemctl start datamart-federatedsearch
mkdir /opt/datamart/camp-api/config-strata
systemctl start datamart-camp-api
```

19. **Platform team installs eventlog UI from build machine**:

```
    cd ~/datamart/datamart-federatedsearch/device-ui
    npm run build
    scp -r build/ <ntid>@<destination fedsearch
machine>:/home/<ntid>/
    ssh <ntid>@<destination fedsearch machine>
    sudo su -
    cd /opt/datamart/federatedsearch/www
    rm -rf eventlog/*
    mv /home/<ntid>/build/* eventlog/
    systemctl restart datamart-federatedsearch
```

Version	Date	Editor	Change description	JIRA #
1.0	2/28/2024	cshymko@denaliai.com	Document created	DSD-8369