Telephony Capture Service

Quick Start Guide

Version 1.0.0



**Document Modification History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Editor** | **Date** | **Modification** |
| 1.0.0 | R Monk | 2017-01-31 | Original Release |

Table of Contents

1 Introduction 4

1.1 Purpose 4

1.2 Scope 4

1.3 Document Location 4

1.4 Document Status 4

1.5 Acronyms Definitions 4

1.6 Reference and Related Documents 4

1.7 Open Issues 4

2 Basic TCS Usage 5

3 Database Management 6

4 Version Management 7

# Introduction

## Purpose

ToDo

## Scope

ToDo

## Document Location

This document is found in the ‘docs’ folder of the TCS GitHub repository:

<https://github.com/ccbcadmin/telephony-capture-service.git>

## Document Status

This document must be kept current and released concurrently with each software release.

## Acronyms Definitions

The reader is referred to the [TCS Software Requirements Document](TCS Software Requirements Document.docx)

## Reference and Related Documents

* [TCS Software Requirements Document](TCS Software Requirements Document.docx)
* [TCS Developer Manual](TCS Developer Manual.docx)

## Open Issues

ToDo

# Basic TCS Usage

1. Start the TCS:
   * $ tcsproj
   * $ tcs
2. Confirm 2 ways that the 6 containers are running (barman, pg1, rabbitmq, database-interface, pbx-interface, and tms-interface):
   * Using Kitematic
   * $ docker ps
3. Right now nothing is actually happening, but you can get a input of SMDR traffic with the following command:

$ pbx-simulator (one message per second).

* Confirm 7 Running containers
* There are 2 useful observations you can make at this point:
  + First, the the tms-simulator is not yet running, which means that the queue PROD\_TMS\_QUEUE should be slowing growing in size.  You can check this on a browser with the following url: localhost:15672.  Select guest/guest and then select the menu Queues.  Look for PROD\_TMS\_QUEUE - it should be non-empty.
  + Execute the following commands:
    - $ psql1   
      This will open an SQL terminal session to pg1.
    - prod=# select count(\*) from smdr;   
      Execute this several times.  The total record count should be increasing.
    - \q (exits out of psql)
* Now start the tms-simulator with the following command:

$ tms-simulator

Check that there are now 8 running containers.  Use the RabbitMQ interface to check that indeed the number of PROD\_TM\_QUEUE messages is now zero.

# Database Management

This section simply asks you to try out the various database management utilities (list-backups, pg-switch, pg-rollback-recovery, pg-standby-recovery).

* $ list-backups  
  (ToDo: there are some Barman config issues that need to be addressed).
* $ pg-rollback-recovery <pg1/2> backup\_id [pitr\_timestamp]  
  Note that  pitr\_timestamp is optional.  You need to run $ list-backups to get the backup\_id that you want.
* $ pg-offline-recovery backup\_id [pitr\_timestamp]  
  The parameters are the same as for pg-rollback-recovery, but this time the recovery is the pg2 container. Note that after executing this command, a new container, pg2, will exist and be running. You can connect to pg2 using port 5433.

# Version Management

I will leave the TCS state as follows:

* Three versions will be available, let me call them versions x, x+1, x+2 (or more properly v0.x, v0.x+1, v0.x+2).  I will give you specific values for x later, but the docker images for only version x will be available.  You will have been running version x thus far.
* Now let us assume that a developer has been working on a feature request. But in the first instance, s/he introduces a bug and releases a version x+1 to you.  Let's make it an unsubtle bug - the PROD-pbx-interface container can no longer connect to RabbitMQ.  Run the following:

$ tcsproj v0.x+1

This command does a few things:

* It goes to github and gets the latest software and does a branch to version v0.x+1
* It then goes and gets from the Docker Hub the 2 tcs images (tcs-image and tcs-jenkins).
* It will then restart all containers to use the new x+1 version.
* BUT, angry messages will be shown in Kitematic for the container PROD-pbx-interface indicating the bug.
* So, let us say that your developer fixes the problem and it is in version x+2 and it is now available.  Execute the following:

$ tcsproj v0.x+2

* After this completes executing, you should see that the PROD-pbx-interface problem has been fixed.