Interface Specification

Paidwall Charging Server

1. Purpose

The purpose of this document is to share the details of charging server.

Charging server have following interfaces/ cronjobs which are listed below:

- ChargingQueryScheduler_CQ
- ChargingScheduler_DR
- ChargeProcessorSchedulerTimer_CP
- AccessTokenScheduler_AT
- OptoutSMSScheduler_OS
- UnsubAPISubscribersScheduler_US
- VoiceBundleCheckScheduler_VB
- ChargingPurgingNotificationScheduler_CHR_PUR_NT
- ChargingPurgingScheduler_CHR_PUR
- UsagePurgingNotificationScheduler_USG_PUR_NT
- UsagePurgingScheduler_USG_PUR

2. Scope

The document explains the specifications of charging server that will be used for charge all subscribers in daily cycles.

3. Functionality of Charging Server

KZ NOC Team would monitor all the Paidwall charging cycles. The third party API and DB credentials are configurable application.properties file is used. The functionality is based on given below cronjobs.

3.1 ChargingQueryScheduler_CQ

The charging query scheduler cron trigger at after every two minutes, every day starting on the 1st, every month. First of all checking the redis connectivity is it pingable or is connected if not then "**CHARGIN_FAILURE_SMS**" sent to NOC team "**please check the system there is an issue with charging deduction**". Following steps are performed if redis connectivity is up.

- executing batch from redis
- Update subscriber table from subscriber_mem
- Update charge_process table from charge_mem
- Insert into daily_charge_attempts from daily_charge_attempts_mem

- TRUNCATE TABLE charge_mem
- TRUNCATE TABLE subscriber_mem
- TRUNCATE TABLE daily_charge_attempts_mem

3.2 ChargingScheduler_DR

The charging scheduler cron trigger to populate new base for today charging. Following steps are performed in this cron.

- Checks if isDailyResetRunning is running
- Checks if isChargingProcessRunning is running
- Checks if daily reset already done exiting
- If above mentioned steps are false then dumping data yesterday's charge_process to table charge_{current_month}
- TRUNCATE TABLE charge_process
- Populate charge_process for today from subscriber table

3.3 ChargeProcessorSchedulerTimer_CP

Charging deduction are started here when charge processor scheduler timer cron triggered. Following steps are performed in this cron.

- Charging cycle start message sent via email and sms to selected participants and charging thread scheduler called in threads for deduction.
- Checks FailedRequestsCount from charge process if it returns zero then update all records as set processed=0 where (stat=0 or stat=-100) to start pointer from top.
- The burstChunkSize calculations are performed.
- Populate msisdn chunk from charge_process. [First attempts all zero stat then stat -100]
- GetToken from DB if expired then token will be updated first.
- A few checks are implemented before call middleware for deduction if no conditions met then deduction api call will be originate otherwise log with failure.
- In api call method first checks redis connectivity if it fails an alert sms will be sent to NOC.
- API response will be log as either success or failure.
- The above mentioned steps are performed recursively until all cellno are processed and finally charging cycle ended alert sent via email and sms.
- Deduction cron has been set for next 10 minutes after 10 minutes same cycle will be triggered.

3.4 AccessTokenScheduler_AT

Access token scheduler cron triggered after every minute. It insures and checks that token is updated properly in DB if token is expired then updates it by calling Update Token API.

3.5 OptoutSMSScheduler_OS

Opt out SMS scheduler when it is triggered then a notification SMS sent to the subscribers whose next_charge_dt is next day.

3.6 UnsubAPISubscribersScheduler_US

Unsub API subscribers scheduler triggered and remove all subscribers from service where next_charge_dt is less than current_date and sub_mode is equal to “API” via unsub API.

3.7 VoiceBundleCheckScheduler_VB

Voice bundle check scheduler triggered and remove all subscribers from service where sub_mode is equal to “VB” and current_date-1 = next_charge_dt via unsub API.

3.8 ChargingPurgingNotificationScheduler_CHR_PUR_NT

Charging purging notification scheduler is triggered and notification sms sent to the subscribers where subscriber exist and not charged from last 58 days.

3.9 ChargingPurgingScheduler_CHR_PUR

Charging purging scheduler is triggered and remove subscriber from service where user not charged from last 60 days.

3.10 UsagePurgingNotificationScheduler_USG_PUR_NT

Usage purging notification scheduler triggered and notification sms sent to the subscribers where user not used IVR from last 9999 days.

3.11 UsagePurgingScheduler_USG_PUR

Remove all users from service where last_call_dt is null or not used in 9999 days.