package com.twitter.visibility.interfaces.push\_service

import com.twitter.finagle.stats.StatsReceiver

import com.twitter.tweetypie.thriftscala.Tweet

import com.twitter.visibility.builder.VisibilityResult

import com.twitter.visibility.rules.Rule

import com.twitter.visibility.rules.RuleResult

import com.twitter.visibility.rules.State

object PushServiceVisibilityLibraryUtil {

def ruleEnabled(ruleResult: RuleResult): Boolean = {

ruleResult.state match {

case State.Disabled => false

case State.ShortCircuited => false

case \_ => true

}

}

def getMissingFeatures(ruleResult: RuleResult): Set[String] = {

ruleResult.state match {

case State.MissingFeature(features) => features.map(f => f.name)

case \_ => Set.empty

}

}

def getMissingFeatureCounts(results: Seq[VisibilityResult]): Map[String, Int] = {

results

.flatMap(\_.ruleResultMap.values.toList)

.flatMap(getMissingFeatures(\_).toList).groupBy(identity).mapValues(\_.length)

}

def logAllStats(

response: PushServiceVisibilityResponse

)(

implicit statsReceiver: StatsReceiver

) = {

val rulesStatsReceiver = statsReceiver.scope("rules")

logStats(response.tweetVisibilityResult, rulesStatsReceiver.scope("tweet"))

logStats(response.authorVisibilityResult, rulesStatsReceiver.scope("author"))

}

def logStats(result: VisibilityResult, statsReceiver: StatsReceiver) = {

result.ruleResultMap.toList

.filter { case (\_, ruleResult) => ruleEnabled(ruleResult) }

.flatMap { case (rule, ruleResult) => getCounters(rule, ruleResult) }

.foreach(statsReceiver.counter(\_).incr())

}

def getCounters(rule: Rule, ruleResult: RuleResult): List[String] = {

val missingFeatures = getMissingFeatures(ruleResult)

List(s"${rule.name}/${ruleResult.action.name}") ++

missingFeatures.map(feat => s"${rule.name}/${feat}") ++

missingFeatures

}

def getAuthorId(tweet: Tweet): Option[Long] = tweet.coreData.map(\_.userId)

def isRetweet(tweet: Tweet): Boolean = tweet.coreData.flatMap(\_.share).isDefined

def isQuotedTweet(tweet: Tweet): Boolean = tweet.quotedTweet.isDefined

}