package com.twitter.recosinjector.config

import com.twitter.finagle.mtls.authentication.ServiceIdentifier

import com.twitter.finagle.stats.StatsReceiver

import com.twitter.finagle.thrift.ClientId

import com.twitter.logging.Logger

import com.twitter.recosinjector.decider.RecosInjectorDecider

case class ProdConfig(

override val serviceIdentifier: ServiceIdentifier

)(implicit val statsReceiver: StatsReceiver) extends {

// Due to trait initialization logic in Scala, any abstract members declared in Config or

// DeployConfig should be declared in this block. Otherwise the abstract member might initialize

// to null if invoked before before object creation finishing.

val recosInjectorThriftClientId = ClientId("recos-injector.prod")

val outputKafkaTopicPrefix = "recos\_injector"

val log = Logger("ProdConfig")

val recosInjectorCoreSvcsCacheDest = "/srv#/prod/local/cache/recos\_metadata"

val recosInjectorDecider = RecosInjectorDecider(

isProd = true,

dataCenter = serviceIdentifier.zone

)

} with DeployConfig