package com.twitter.tweetypie.serverutil

import com.twitter.tweetypie.thriftscala.CardReference

import com.twitter.util.Try

import java.net.URI

import scala.util.control.NonFatal

/\*\*

\* Utility to extract the stored card id out of a CardReference

\*/

object StoredCard {

private val cardScheme = "card"

private val cardPrefix = s"$cardScheme://"

/\*\*

\* Looks at the CardReference to determines if the cardUri points to a "stored"

\* card id. Stored Card URIs are are expected to be in the format "card://<long>"

\* (case sensitive). In future these URIs can potentially be:

\* "card://<long>[/path[?queryString]]. Note that this utility cares just about the

\* "Stored Card" types. So it just skips the other card types.

\*/

def unapply(cr: CardReference): Option[Long] = {

try {

for {

uriStr <- Option(cr.cardUri) if uriStr.startsWith(cardPrefix)

uri <- Try(new URI(uriStr)).toOption

if uri.getScheme == cardScheme && uri.getHost != null

} yield uri.getHost.toLong // throws NumberFormatException non numeric host (cardIds)

} catch {

// The validations are done upstream by the TweetBuilder, so exceptions

// due to bad URIs will be swallowed.

case NonFatal(e) => None

}

}

}