```python
from pydrive.auth import GoogleAuth
from pydrive.drive import GoogleDrive as PyDriveGoogleDrive
#from tkinter import Tk
#from tkinter.filedialog import askopenfilename
from tkinter.filedialog import asksaveasfile
#from threadsafe_tkinter import *
from easygui import fileopenbox
from easygui import filesavebox
import ntpath
from Crypto.Cipher import AES
from app.models import Group
import os

class GoogleDrive():
    gauth = None
    drive = None

    def __init__(self):
        self.gauth = GoogleAuth()
        self.gauth.LoadCredentialsFile("mycreds.txt")
        if self.gauth.credentials is None:
            # Authenticate if they're not there
            self.gauth.LocalWebserverAuth()
        elif self.gauth.access_token_expired:
            # Refresh them if expired
            self.gauth.Refresh()
        else:
            # Initialize the saved creds
            self.gauth.Authorize()
        # Save the current credentials to a file
        self.gauth.SaveCredentialsFile("mycreds.txt")

        self.drive = PyDriveGoogleDrive(self.gauth)

    def getFiles(self):
        return self.drive.ListFile({'q': "'1a_ZOqi75h6nTvsUEPDi8NGUrb9Tk-dkh' in parents and trashed=false"}).GetList()

    def getFilesInFolder(self, folder):
        return self.drive.ListFile({'q': "'" + folder + "' in parents and trashed=false"}).GetList()

    def uploadFile(self, folder, groupId):
        #Tk().withdraw() # we don't want a full GUI, so keep the root window from appearing
        filepath = fileopenbox() # show an "Open" dialog box and return the path to the selected file
        file = self.drive.CreateFile({'title': ntpath.basename(filepath), 'parents': [{'kind': 'drive#fileLink', 'id': folder}]})
        f = open(filepath, 'rb')
        self.encrypt(groupId, f.read())
        file.SetContentFile('encrypted.bin')
        file.Upload()
        #if os.path.exists('encrypted.bin'):
        #    os.remove('encrypted.bin')

    def downloadFile(self, id, title, groupId):
        file = self.drive.CreateFile({'id': id})
        file.GetContentFile('title')
        file.GetContentFile('encrypted.bin')
        #Tk().withdraw()
        destination = filesavebox()
        if destination is None:
            return
```

```
60          self.decrypt(groupId, destination)
61          #file.GetContentFile(destination)
62
63      def createGroup(self, title):
64          folder = self.drive.CreateFile({'title': title, 'mimeType' : 'application/vnd.google-
    apps.folder', 'parents': [{'kind': 'drive#fileLink', 'id': '1a_ZOqi75h6nTvsUEPDi8NGUrb9Tk-
    dkh'}]})
65          folder.Upload()
66          return folder['id']
67
68      def encrypt(self, groupId, data):
69          key = Group.objects.get(id=groupId).key
70          cipher = AES.new(key, AES.MODE_EAX)
71          ciphertext, tag = cipher.encrypt_and_digest(data)
72          file_out = open("encrypted.bin", "wb")
73          [ file_out.write(x) for x in (cipher.nonce, tag, ciphertext) ]
74
75      def decrypt(self, groupId, destination):
76          key = Group.objects.get(id=groupId).key
77          file_in = open("encrypted.bin", "rb")
78          nonce, tag, ciphertext = [ file_in.read(x) for x in (16, 16, -1) ]
79          cipher = AES.new(key, AES.MODE_EAX, nonce)
80          data = cipher.decrypt_and_verify(ciphertext, tag)
81          typeofdata = type(data)
82          file_out = open(destination, "wb")
83          file_out.write(data)
84
85      def fucksake(self):
86          with open('E:\\TEST1.PNG', 'rb') as f1:
87              with open('E:\\ugggggggh.png', 'wb') as f2:
88                  f2.write(f1.read())
```