Team 25

Team members: Jon Crow (cjonat1@vt.edu), Chris Henshaw (chenshaw@vt.edu, Bryan Pangburn (bpang@vt.edu)

Assignment: ECE 4564 Assignment 1

Libraries needed: sys, os, subprocess, socket, pickle, cryptography (Fernet), hashlib, wolframalpha, vlc, Watson_developer_cloud, pyzbar, time, pulseaudio, cvlc, click, cv2 (See instructions below)

To run the program, change to the appropriate directory and type:

Client: python3 client.py -sip <IP address> -sp <PORT> -z <Socket size>
Server: python3 server.py -sip <IP address> -sp <PORT> -z <Socket size> on the server

We changed the parameters of server.py so the user knows the IP address on both boards. Initiate the server before the client.

Design:

Once the connection has been made, the server will then wait on the client to get a question from the camera in the form of a QR code. The camera will scan for an image. Press any key on the keyboard to initiate the program. If the program finds an acceptable QR code, it will proceed. If not, repeat the process until so. The client will then decode the QR and then send the question to the server. Before sending it, the question will be pickled, encrypted, and an md5 checksum will be generated. The server will decrypt the question and speak the question with the use of IMB Watson and then send it to Wolfram Alpha. Wolfram Alpha will send the answer back to the server. The server will then pickle and encrypt the answer and generate an md5 checksum before the answer is sent back to the client. Then client will decrypt the answer and then speak the answer with the use of IBM Watson again.

Outcomes:

The project works as intended. If vlc does not play the audio file, it is due to vlc not responding. Reboot the boards and try again. If Wolfram Alpha returns an error, it is due to Wolfram not being able to answer the question. Sometimes it will return "(no data available)" or it will return an error. Control-C to exit the program.

To get cv2 to run, follow the instruction at: https://www.deciphertechnic.com/install-opencv-python-on-raspberry-pi/

Resources Used:

Sockets – code provided by Dr. Plymale.
Wolfram Alpha - https://pypi.org/project/wolframalpha/
Watson - https://watson-developer-cloud-python-sdk.readthedocs.io/en/latest/
Fernet - https://cryptography.io/en/latest/fernet/
Pickle - http://code.activestate.com/recipes/577667-pickle-tofrom-socket/
CV2 - https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_gui/py_image_display/py_image_display.html
QR - https://pypi.org/project/pyzbar/