Up and Running Software - C++/S3 Mini-Project

Overview:
Implement a multi-threaded HTTP listener that performs the following operations via a REST interface

1. WAV Info - Accepts an S3 path to a WAV file and returns the number of channels and sample rate of the WAV file with a JSON result.
2. MP3 to WAV - Accepts an S3 path to an MP3 file that should be converted to a WAV file and uploaded to the provided destination path with a JSON result including the file size of the resulting WAV file.

Requirements:

a. General Listener requirements
   i. Should respond to HTTP requests on port 8089 or whatever port is passed via the -p command line parameter
   ii. Should be multi-threaded/multi-process with non-blocking socket
   iii. Should build via make or cmake on Ubuntu 16.04 with gcc 5.4
   iv. S3 credentials and bucket info can be hard-coded however should support subfolders of the bucket in the key
b. Wav Info requirements
   i. Should process requests for path: /wav-info
   ii. The parameter will be: wavkey (s3 key)
   iii. The JSON result should look like:
   {
   “channel_count”: [integer],
   “sample_rate”: [integer],
   “execution_time”: [float seconds]
   }
   iv. Example request: /wav-info?wavkey=test1.wav
   v. You can extract this info via the header struct of the WAV file or using the soxi command line app in a shell
c. MP3 to WAV requirements
   i. Should process requests for path /mp3-to-wav
   ii. The parameters will be: mp3key (s3 key) and wavkey (s3 key)
   iii. The JSON result should look like:
   {
   “file_size”: [long bytes of new wav file],
   “execution_time”: [float seconds]
   }
   iv. Example request: /mp3-to-wav?mp3key=test2.mp3&wavkey=test2.wav
   v. This conversion can be performed using sox in a shell