Network Hashing Server

## Overview

Write a network server and client for cryptographically hashing files. The hashing server will be listening on the network. The client will connect to the server, indicate the desired hashing algorithm, upload the file, and receive the hash of the file. The client will then display the hash

## Server Requirements

* The server must be written in python.
* The server will be executed using python 3.6.
* The server must accept the following arguments:
  + Port: The TCP port to listen on. This argument is optional.

## Client Requirements

* The client must be written in C.
* The client must compile using gcc on Ubuntu 16.04.
* The client will be executed on Ubuntu 16.04.
* The client must accept the following arguments:
  + Port: The TCP port to connect on. This argument is optional.
  + Address: The IP address where the server is listening. This argument is mandatory.
  + Hash name: The name of the hashing algorithm to use. The following values should be supported: sha1, sha256, sha512, and md5. This argument is mandatory.
  + Filename: The name of the file or files to hash. A minimum of one filename is required with no maximum.
  + Example usage: hashclient [-p port] <address> <hashname> <filename>…
* The client will print the name and hash (in hex format) of each file specified to stdout.

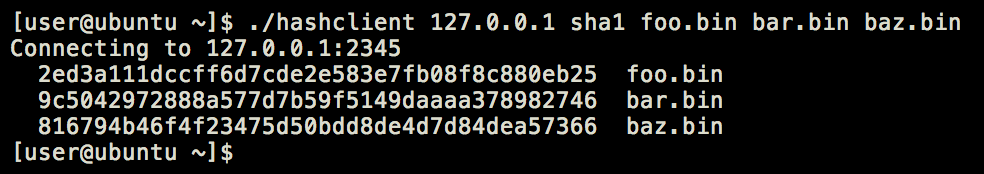


Figure : Example client output

## System Requirements

* The standard port on which the client and server communicate is TCP 2345.
* During file transfers, the client must not exceed 4096 byte chunks.
* The protocol between the client and server should be as efficient as possible with regards to size.
* Third-party python packages should be installable via requirements.txt.
* Third-party Ubuntu packages should be indicated via README.
* Include a Makefile to build the project.
* Automated tests are encouraged but not required.
* Documentation is encouraged but not required.