DATE: 09/26/2013
COURSE: CSCI-P538
YEAR: Fall 2013
NAME: Prachi Shah(pracshah)
PROJECT: Lab 2: Socket Programming: netcat_part

1) The project explains us Application level programming using Sockets.
2) We will perform TCP socket communication between the client and server.
3) This will be implemented using the netcat command which is used for client-server communication over network.
4) netcat command:

- The netcat command enables one to read and write data over different network connections using TCP(Transmission Control Protocol) and UDP(User Datagram Protocol).
- We will use the TCP communication in this project.
- Syntax:
  > bash> nc mail.servername.com portnumber
  > where, nc is the netcat command
  > This command will enable you to connect to the hostname 'mail.servername.com' using port number specified.
- This command has the following usages:
  > netcat_part [OPTIONS] dest_ip file
  > -h Print this help screen
  > -v Verbose output
  > -w Enable website get mode at client
  > -p Set the port to connect on (dflt: 6767)
  > -n Number of bytes to send, default is whole file
  > -o Offset into file to start sending
  > -l Listen on port instead of connecting and write output to file and dest_ip refers to which ip to bind to (dflt: localhost)
- -h will display the above help data for a user.
  > -v will print the verbose output
  > -w will enable GET mode at client side to get data from the server
  > -p port number specifies the port used to while connecting a client/server
  > -n Its specifies the number of bytes to send
  > -o Offset specifies the number of bytes to be fetched from a file from a certain location.
  > -l It is a server side option to enable a server to listen to incoming connections over a port and to write the output data to a file
- The netcat command works as both a client and server.
  > a] netcat as client: It enables connection to a remote server to send input data from stdin over the network. The client opens a socket to establish a TCP connection with the server.
  > b] netcat as server: It can also function as a server. The server will listen to incoming connections and store incoming data to stdout or into a file.

- Security in netcat: This command is used to provide a level of security to the packets being transferred over the network. You will use the openssl/hmac.h library to enable security. This will create a message authentication code (MAC) which will act as an integrity check for the data being send over the network.
- Also, Various file operations are used in this project to read and write data to a file.:
    ssize_t write(int filedes, void * buf, size_t nbyte)
    ssize_t read(int filedes, void * buf, size_t nbyte)
    fseek(), fopen(), fread(), fwrite() file operations are also used.

Credits:
1)   CSCI- P538 lab document.
        This lab is c Adam Aviv, which we have modified for P-538.
2)   Netcat security
        http://archive.networknewz.com/networknewz-10-20031020NetCatSecurity.html
3)   Netcat
        http://en.wikipedia.org/wiki/Netcat
4)   Netcat – a couple of useful examples
        http://www.g-loaded.eu/2006/11/06/netcat-a-couple-of-useful-examples/