CS 6353
Unix and Network Security
Assignment 2
Due Wednesday March 23

1. (100 pts) Write a C/C++ program to experiment with one-way property of cryptographic hash functions. Use the *openssl* implementation of hash functions for this purpose. Include the following in your program to use openssl

```
#include <openssl/ssl.h>
```

Your program should read the number of bytes n and the byte values $b_1, \ldots, b_n$ from the user and find a 20 byte message that produces a matching cryptographic hash. Sample output is given below. This output shows the message and digest using hexadecimal format. Each symbol represents the value of 4 bits. 0 corresponds to 0000 and f corresponds to 1111.

```
Enter Number of Bytes to match
1
Enter the byte values to match for 1 bytes
3
TargetDigestPrefix in hexadecimal for 1 bytes
03

Match in 308.000000 runs
Message in hexadecimal format
   867c5d940b53a21ad9aebfec32f09ebb32448f76
MD5 Digest in hexadecimal format
   034279537f0135350f584dc13bc98ab3
```

In about output user wants to find a message whose first byte matches 3 (03 in hexadecimal). After 308 random messages are generated a match is found. The 20 byte message and 16 byte MD5 Hash are displayed as part of the output.

```
Enter Number of Bytes to match
2
Enter the byte values to match for 2 bytes
3 4
TargetDigestPrefix in hexadecimal for 2 bytes
0304

Match in 1035.000000 runs
Message in hexadecimal format
   58a36df59b53dd2f35b4ca9dda4b7cb6fb802264
MD5 Digest in hexadecimal format
   030424cc397f2351b65e7de2468eeec8
```

To find a match for two bytes requires 1035 runs. This output finds a match for two bytes 3 and 4 (0304 in hexadecimal).

Use 20 byte messages and MD5 hash function in your program. You need to the use EVP API. Take a look at the following functions in EVP API. You need these functions in your program

```
EVP_DigestInit(...);
EVP_DigestUpdate(...);
EVP_DigestFinal(...);
```

Use the following to compile your program

```
gcc -o hash hash.c -lssl -lcrypto
```

You can use the openssl documentation: *EVP*[1], *digestinit()*[2].

Submit your program electronically using the blackboard system

The program you submit should be your own work. Cheating will be reported to office of academic integrity. Both the copier and copiee will be held responsible.

[1] http://www.openssl.org/docs/crypto/evp.html
[2] http://www.openssl.org/docs/crypto/EVP_DigestInit.html