wm054@cs2:~$ cat DHTalker.cpp

/\* tcptalker.cpp

\*

\* TCP client that gets the time from a server

\*

\* This program is a TCP client using port 1400 that receives, in readable

\* form, the time on the server machine. All that we do is to connect to

\* the server and send the word 'time', then read what is sent back.

\*

\*

\*/

#include <iostream>

#include <stdlib.h>

#include <errno.h>

#include <string.h>

#include <sys/types.h>

#include <netinet/in.h>

#include <sys/socket.h>

#include <sys/wait.h>

#include <arpa/inet.h> // inet\_addr()

#include <math.h>

#include <sstream>

#define SIZE 1024

#define TIME\_PORT 20054

using namespace std;

char buf[SIZE];

int numDigits(int number)

{

int digits = 0;

while (number>=0) {

number /= 10;

digits++;

}

return digits;

}

/\*int CStringConverter(char& [] ToSend, int AnInt){

ostringstream OSS = new ostringstream;

string AString = "";

OSS << AnInt;

int SizeInt = 0;

AString = OSS.str();

SizeInt = AString.length();

ToSend = new char[SizeInt + 1];

strcpy(ToSend, AString.c\_str());

return SizeInt;

}\*/

int main(int argc, char \*argv[])

{

int p = 7;

int g = 6; //0<g<p

int x2 = rand() % (p-1); //0<x1<p-1

int y2 = pow(g, x2); //g^x2

y2 = y2 % p; // %p

cout << "p = " << p << endl;

cout << "g = " << g << endl;

cout << "x2 = " << x2 << endl;

cout << "y2 = " << y2 << endl;

char \* ToSend;

int size = 0;

int sockfd;

int nread, structlen;

struct sockaddr\_in serv\_addr;

if (argc != 2)

{

cout << "usage: " << argv[0] << " IPaddr\n" << argv[0] << endl;

exit(1);

}

if ((sockfd = socket(AF\_INET, SOCK\_STREAM, 0)) < 0)

{ cout << "error with socket()";

exit(2);

}

cout << "sockfd = " << sockfd << endl ;

serv\_addr.sin\_family = AF\_INET;

serv\_addr.sin\_addr.s\_addr = inet\_addr(argv[1]);

serv\_addr.sin\_port = htons(TIME\_PORT);

bzero(&(serv\_addr.sin\_zero),8);

cout << "sockfd " << sockfd << endl;

if (connect(sockfd, (struct sockaddr \*)&serv\_addr, sizeof(serv\_addr)) < 0)

{ cout << "error with connect()" << endl;

exit(3);

}

cout << "after connect" << endl;

//size = CStringConverter(ToSend, y2);

ostringstream OSS;

string AString = "";

OSS << y2;

AString = OSS.str();

size = AString.length();

ToSend = new char[size + 1];

strcpy(ToSend, AString.c\_str());

/\* send the word 'time' to the server \*/

write(sockfd, ToSend, size);

nread = read(sockfd,&buf,SIZE);

buf[nread] = 0;

cout << "tcptalker::Listen-y1: " << buf << endl;

int y1 = atoi(buf);

int z2 = pow(y1, x2);

z2 = z2 % p;

ostringstream OSS2;

string AString2 = "";

OSS2 << z2;

AString2 = OSS2.str();

size = AString2.length();

char\* key = new char[size + 1];

strcpy(key, AString2.c\_str());

string ToEnc = "Chris Graff Said So";

size = ToEnc.length();

ToSend = new char[size+1];

for(int x = 0; x < size; x++)

{

ToSend[x] = ToEnc[x] ^ z2;

}

cout << "Encrypted Message: " << ToSend << endl;

//if (connect(sockfd, (struct sockaddr \*)&serv\_addr,

// sizeof(serv\_addr)) < 0)

// { cout << "error with connect()" << endl;

// exit(3);

// }

//cout << "after connect" << endl;

write(sockfd, ToSend, size);

nread = read(sockfd,&buf,SIZE);

cout << "DHTalker::buf " << buf << endl;

close(sockfd);

}