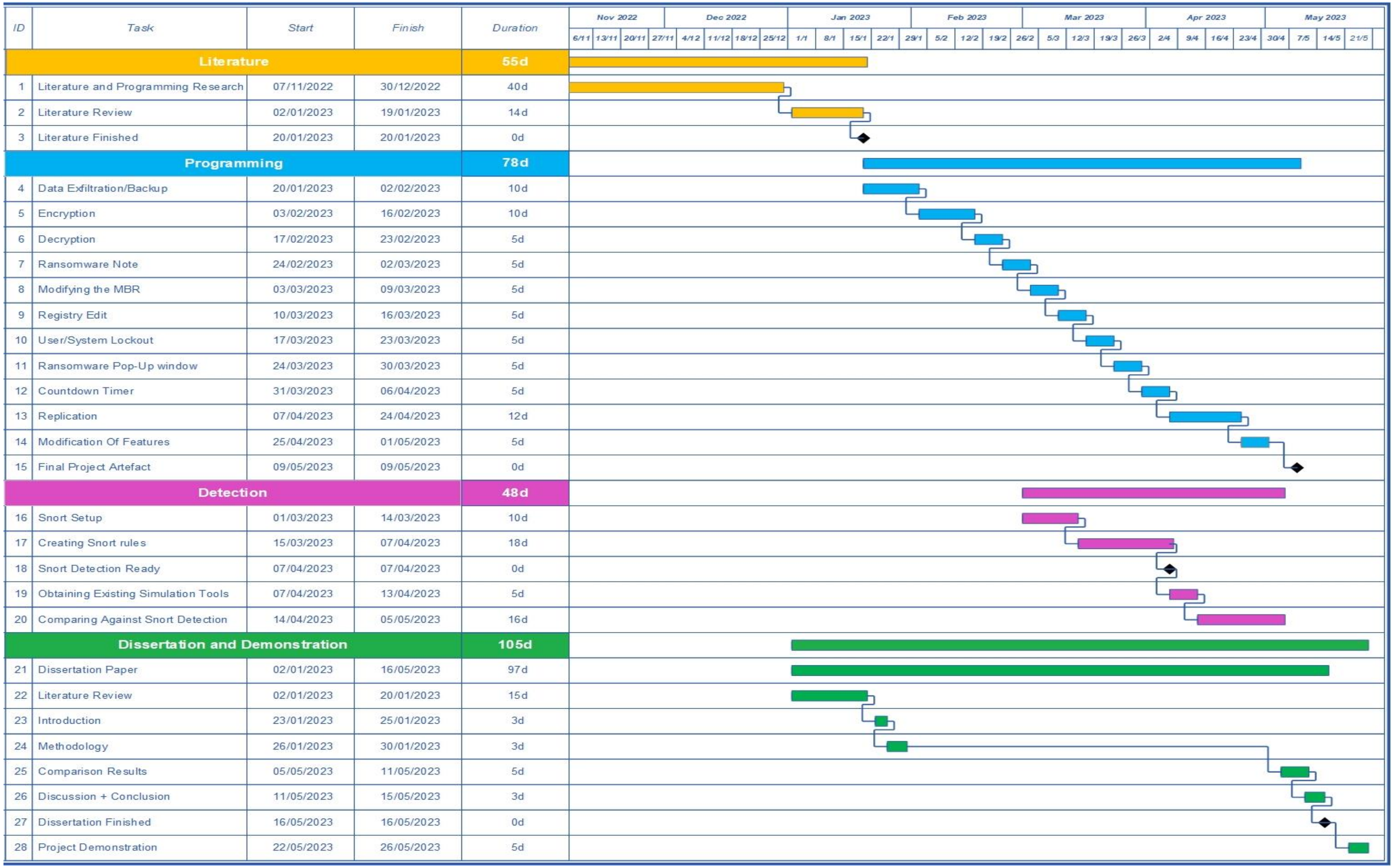
Gantt Chart



ID	Task	Start	Finish	Duration
	Literature			55d
1	Literature and Programming Research	07/11/2022	30/12/2022	40d
2	Literature Review	02/01/2023	19/01/2023	14d
3	Literature Finished	20/01/2023	20/01/2023	0d
	Programming			78d
4	Data Exfiltration/Backup	20/01/2023	02/02/2023	10d
5	Encryption	03/02/2023	16/02/2023	10d
6	Decryption	17/02/2023	23/02/2023	5d
7	Ransomware Note	24/02/2023	02/03/2023	5d
8	Modifying the MBR	03/03/2023	09/03/2023	5d
9	Registry Edit	10/03/2023	16/03/2023	5d
10	User/System Lockout	17/03/2023	23/03/2023	5d
11	Ransomware Pop-Up window	24/03/2023	30/03/2023	5d
12	Countdown Timer	31/03/2023	06/04/2023	5d
13	Replication	07/04/2023	24/04/2023	12d
14	Modification Of Features	25/04/2023	01/05/2023	5d
15	Final Project Artefact	09/05/2023	09/05/2023	0d
	Detection			48d
16	Snort Setup	01/03/2023	14/03/2023	10d
17	Creating Snort rules	15/03/2023	07/04/2023	18d
18	Snort Detection Ready	07/04/2023	07/04/2023	0d
19	Obtaining Existing Simulation Tools	07/04/2023	13/04/2023	5d
20	Comparing Against Snort Detection	14/04/2023	05/05/2023	16d
	Dissertation and Demonstration			105d
21	Dissertation Paper	02/01/2023	16/05/2023	97d
22	Literature Review	02/01/2023	20/01/2023	15d
23	Introduction	23/01/2023	25/01/2023	3d
24	Methodology	26/01/2023	30/01/2023	3d
25	Comparison Results	05/05/2023	11/05/2023	5d
26	Discussion + Conclusion	11/05/2023	15/05/2023	3d
27	Dissertation Finished	16/05/2023	16/05/2023	0d
28	Project Demonstration	22/05/2023	26/05/2023	5d

Risk Analysis

R1 Personal data is encrypted and becomes unrecoverable.

The highest risk and probability during the project is that personal data could be encrypted by the simulation tool and the user ends up losing their data. During development of the tool only the researcher's own personal data and own computer system until there is confidence the product won't have this risk. Then the tool may be tested on different machines such as in the Hacklab/Netlab.

R2 Some ransomware features unable to be safely implemented as envisioned

Due to the nature of this risk and harm to the system and user data, it's best to leave out the feature from the overall ransomware simulation tool. The reason why this feature could not be implemented will be explained in dissertation for future researchers to hopefully aid in their attempt.

R3 Not able to use Hacklab/Netlab network for replication feature.

In the event that the Hacklab/Netlab network is not usable during the replication stage of the tool the project plan will be changed. Instead, a smaller scale proof of concept will be created using a virtual machine network in VMWare or VirtualBox. This will allow testing in a controlled environment that will not affect systems the researcher does not own.

R4 Project data loss

At any point in the project development the project may lose data and progress due to system issues, human error and many more issues that may arise. Backups of the project will be made regularly as a restore point to not lose any significant progress in the project keeping the project on track.

R5 Unable to obtain Ransomware Simulation tools provided by companies.

It's possible that when requesting to use the existing ransomware tools created and provided by companies, they will deny it for my intended use for this project. If this occurs with enough companies only open-source ransomware simulation tools could be used and compared instead.

R6 Unable to implement a feature due to lack of skill

It is a possibility and risk that the researcher simply will not be skilled and knowledgeable enough to implement a feature of the ransomware simulation tool despite best efforts to do so. The researcher will undertake extensive research and previous literature on how certain features could be implemented to avoid this.

R7 Personal circumstances

Illness, family emergencies or other circumstances may occur during the project. This will have an impact on the project productivity and timescales. By keeping in line with the project plan and Gantt chart the workload will be manageable during these situations especially reducing stress and pressure to myself.

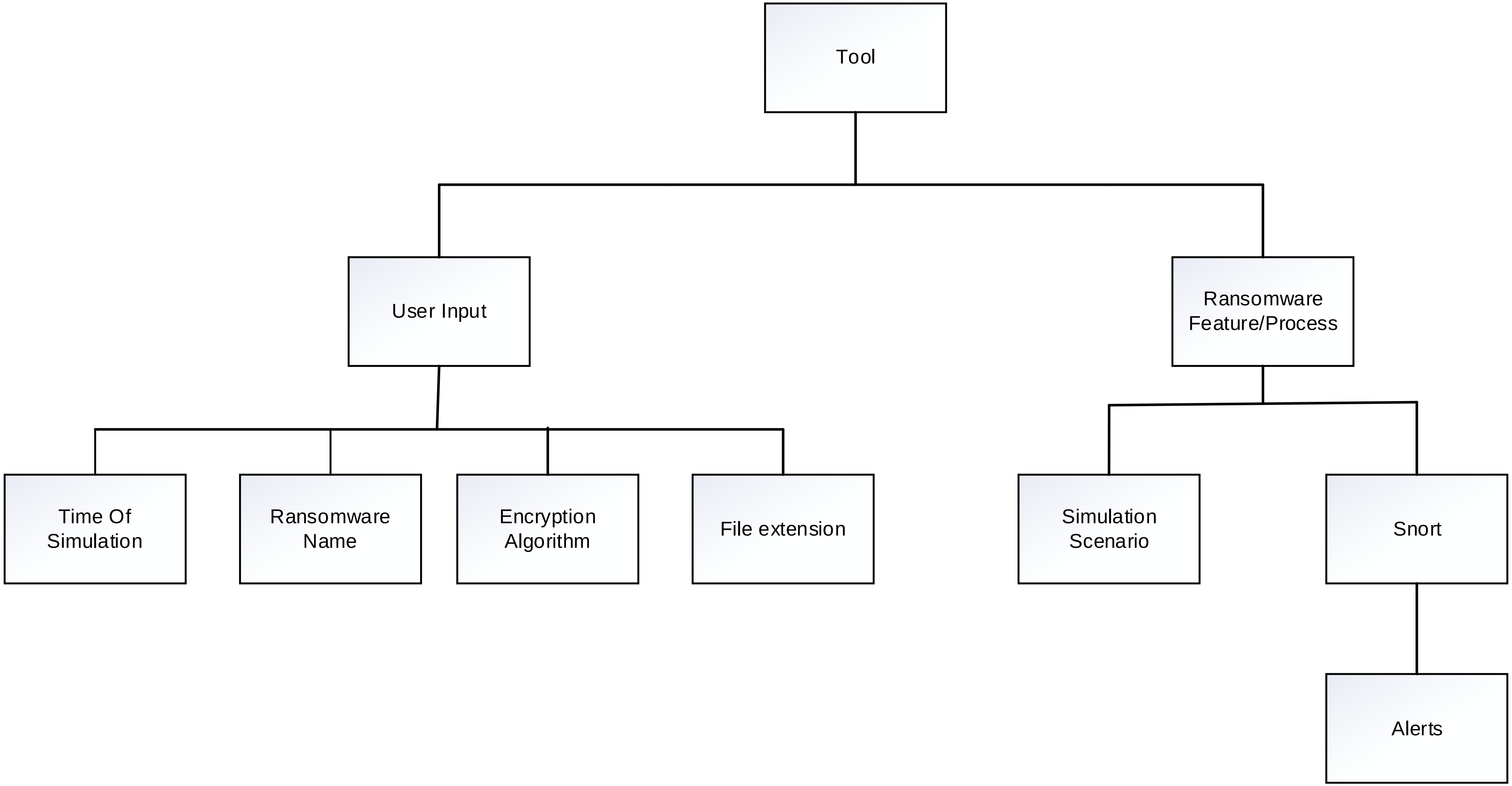
Risk Matrix

Probability of Risk	High			R1
	Moderate		R2 R6	
	Low	R4	R7	R3 R5
		Low	Moderate	High
			Impact of Risk	

Research Question

Are ransomware simulation tools an effective measure to accurately assess network security against a ransomware attack?

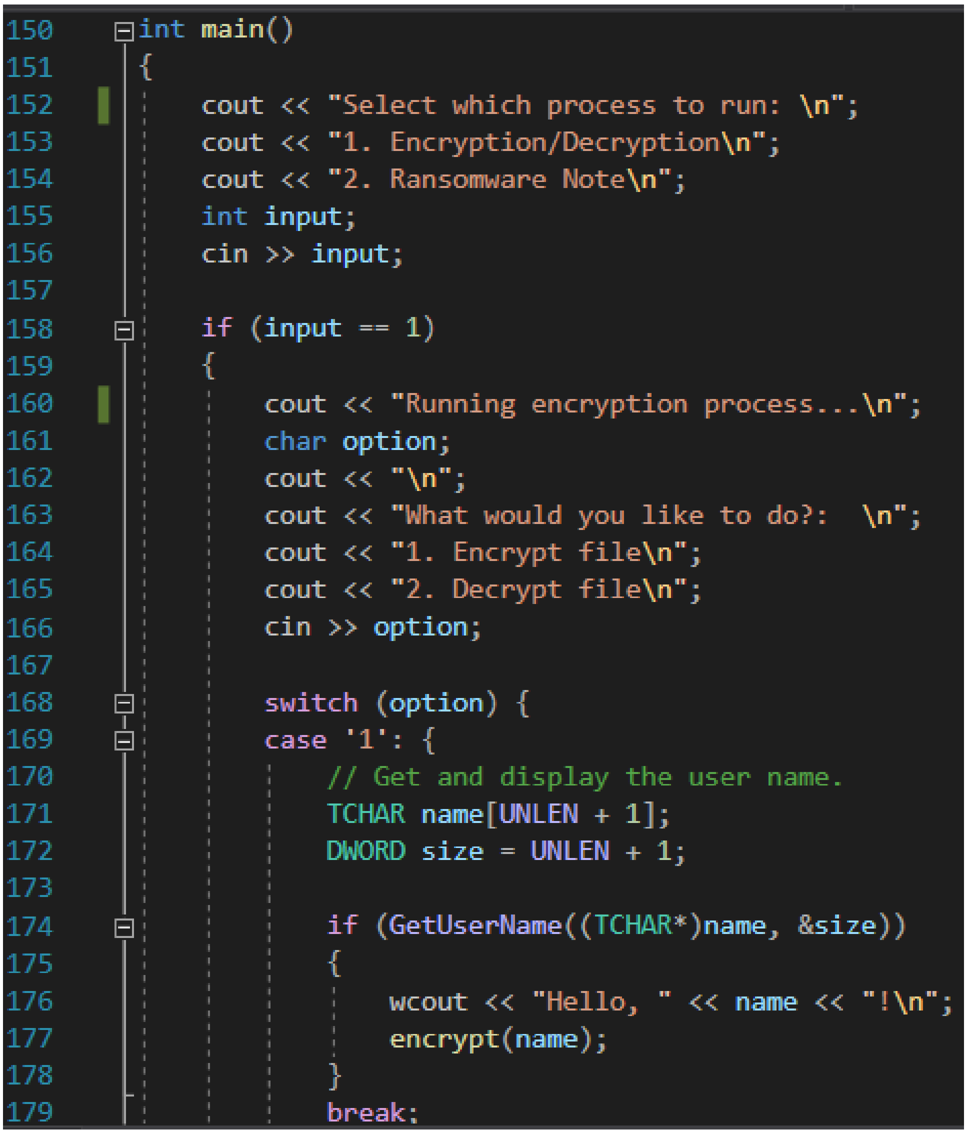
Structure Diagram



Example C++20 Programming Code

Main Function

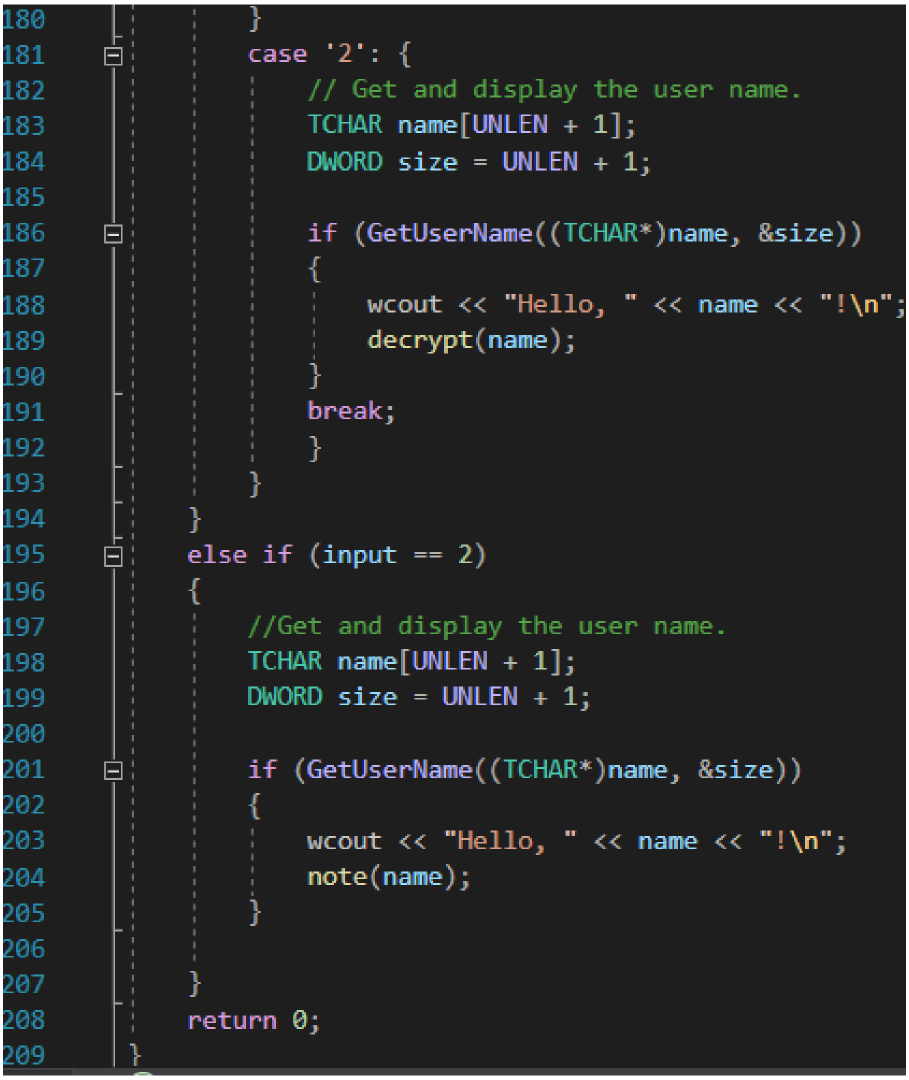
The user selects which process they would like to run first. Either Encryption/Decryption or creating a ransomware note. If in the encryption process, another option is given to either encrypt the file or decrypt the file. The file being used is a text file on the User’s Desktop called “importantData.txt”. To run the program simply create a text file called “importantData.txt” on the windows Desktop and the program will function at its current state.



```
150  int main()
151  {
152      cout << "Select which process to run: \n";
153      cout << "1. Encryption/Decryption\n";
154      cout << "2. Ransomware Note\n";
155      int input;
156      cin >> input;
157
158      if (input == 1)
159      {
160          cout << "Running encryption process...\n";
161          char option;
162          cout << "\n";
163          cout << "What would you like to do?:  \n";
164          cout << "1. Encrypt file\n";
165          cout << "2. Decrypt file\n";
166          cin >> option;
167
168          switch (option) {
169          case '1': {
170              // Get and display the user name.
171              TCHAR name[UNLEN + 1];
172              DWORD size = UNLEN + 1;
173
174              if (GetUserName((TCHAR*)name, &size))
175              {
176                  wcout << "Hello, " << name << "!\n";
177                  encrypt(name);
178              }
179              break;
```

Figure 1: Main function lines 150-179

Next, the Windows account username is passed to each function: encrypt, decrypt and note (Stevewhims, 2021). The Username is used to find the correct file to encrypt/decrypt and to place the ransomware note in the correct Desktop directory on the user system.

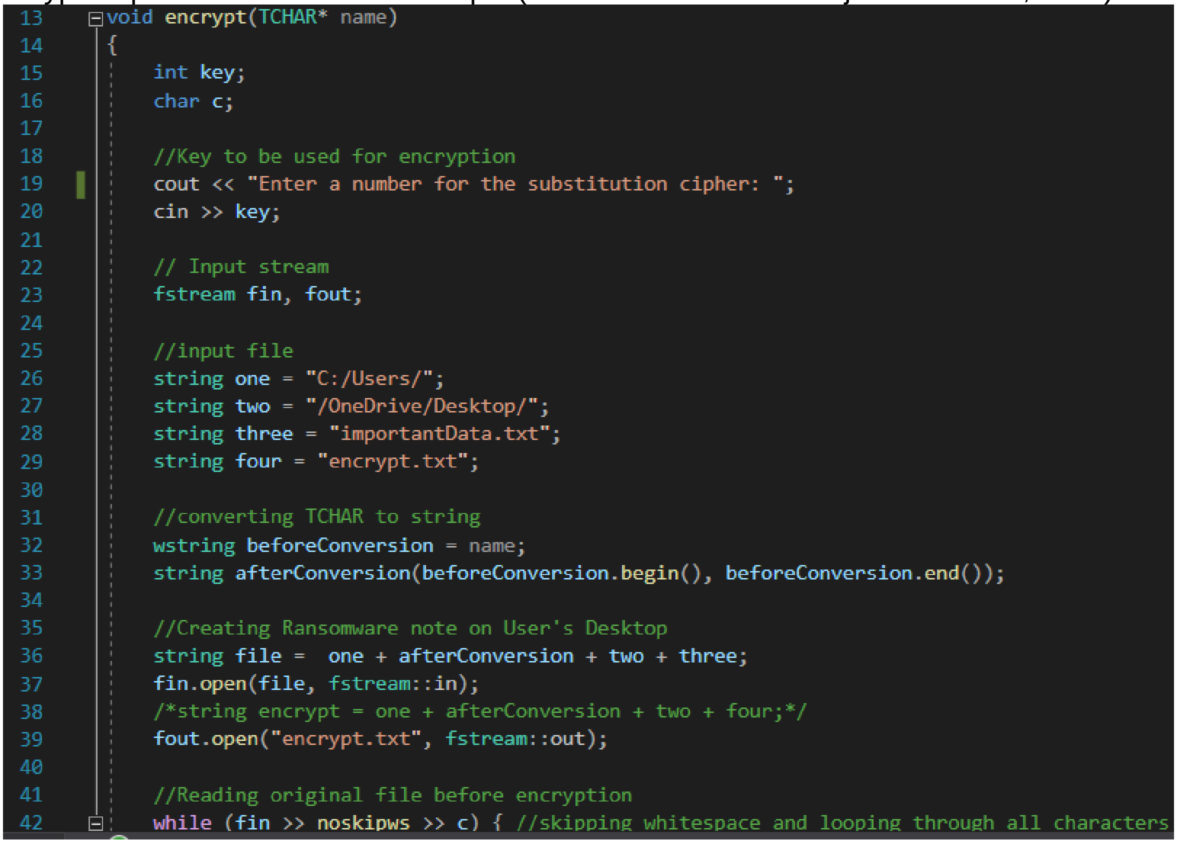


```
        }
        case '2': {
            // Get and display the user name.
            TCHAR name[UNLEN + 1];
            DWORD size = UNLEN + 1;

            if (GetUserName((TCHAR*)name, &size))
            {
                wcout << "Hello, " << name << "!\n";
                decrypt(name);
            }
            break;
            }
        }
    }
    else if (input == 2)
    {
        //Get and display the user name.
        TCHAR name[UNLEN + 1];
        DWORD size = UNLEN + 1;

        if (GetUserName((TCHAR*)name, &size))
        {
            wcout << "Hello, " << name << "!\n";
            note(name);
        }

    }
    return 0;
}
```

Figure 2: Main function lines 180-209

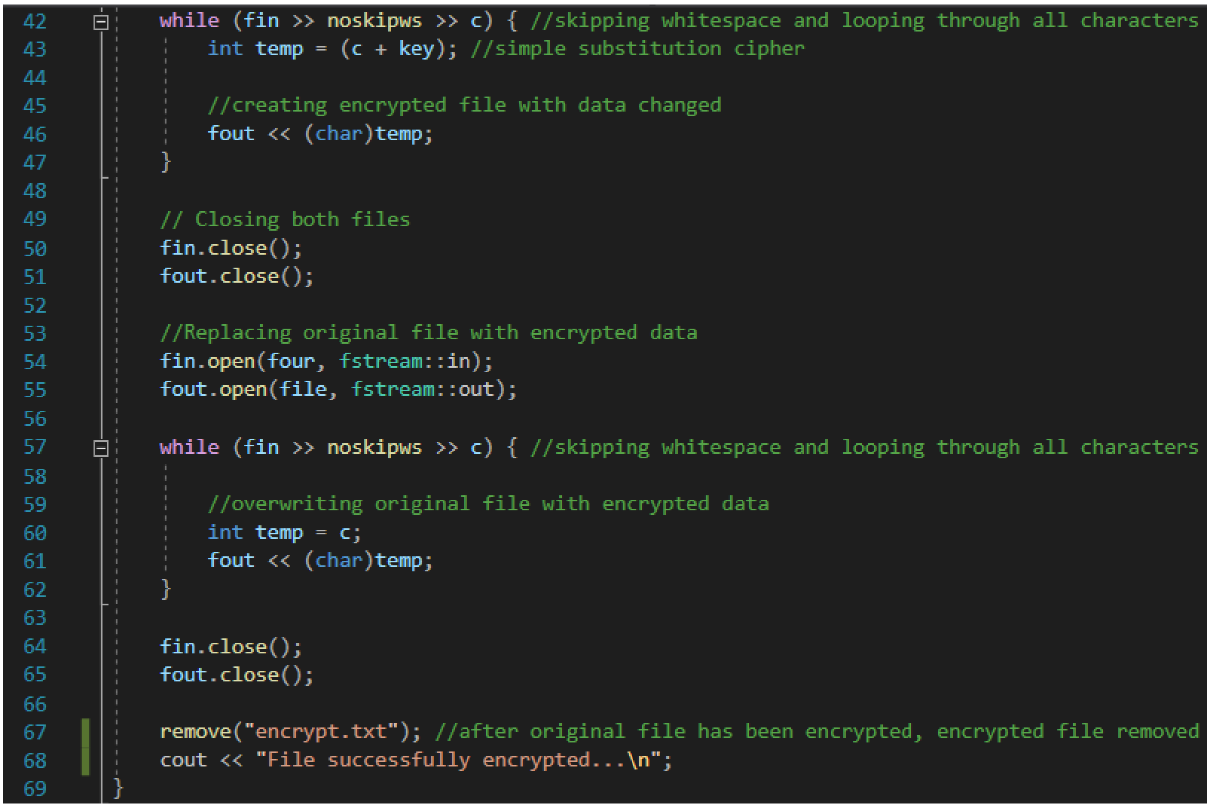
Encryption

A simple substitution cipher was taken from GeeksforGeeks and adapted to create the encryption process seen in this example (GeeksForGeeks and baljeet11801868, 2021).



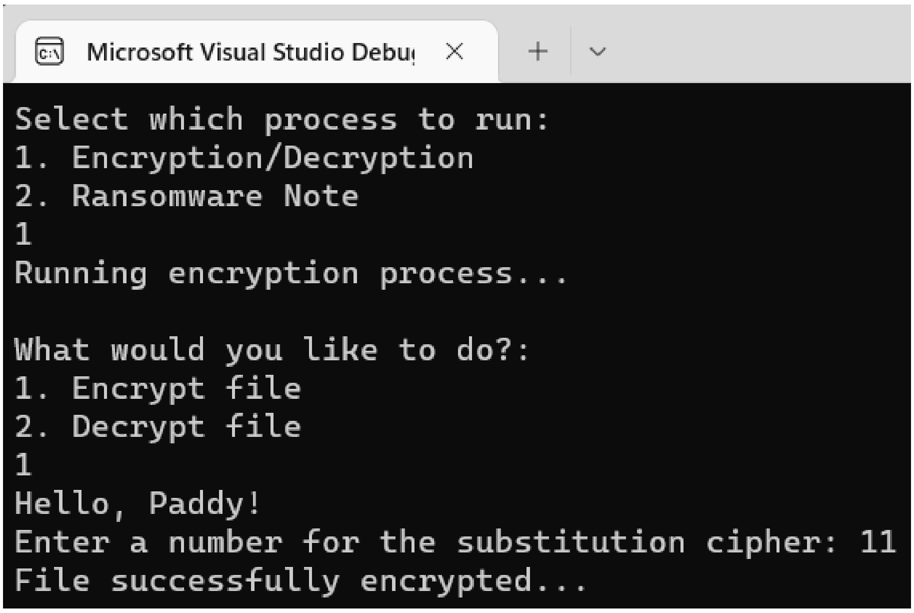
```
13  void encrypt(TCHAR* name)
14  {
15      int key;
16      char c;
17
18      //Key to be used for encryption
19      cout << "Enter a number for the substitution cipher: ";
20      cin >> key;
21
22      // Input stream
23      fstream fin, fout;
24
25      //input file
26      string one = "C:/Users/";
27      string two = "/OneDrive/Desktop/";
28      string three = "importantData.txt";
29      string four = "encrypt.txt";
30
31      //converting TCHAR to string
32      wstring beforeConversion = name;
33      string afterConversion(beforeConversion.begin(), beforeConversion.end());
34
35      //Creating Ransomware note on User's Desktop
36      string file =  one + afterConversion + two + three;
37      fin.open(file, fstream::in);
38      /*string encrypt = one + afterConversion + two + four;*/
39      fout.open("encrypt.txt", fstream::out);
40
41      //Reading original file before encryption
42      while (fin >> noskipws >> c) { //skipping whitespace and looping through all characters
```

Figure 3: Encrypt function lines 13-42



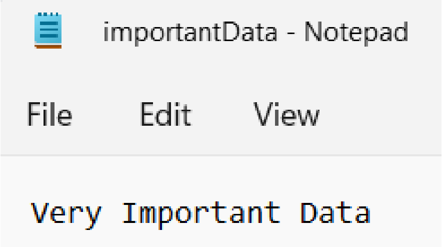
```
42      while (fin >> noskipws >> c) { //skipping whitespace and looping through all characters
43          int temp = (c + key); //simple substitution cipher
44
45          //creating encrypted file with data changed
46          fout << (char)temp;
47      }
48
49      // Closing both files
50      fin.close();
51      fout.close();
52
53      //Replacing original file with encrypted data
54      fin.open(four, fstream::in);
55      fout.open(file, fstream::out);
56
57      while (fin >> noskipws >> c) { //skipping whitespace and looping through all characters
58
59          //overwriting original file with encrypted data
60          int temp = c;
61          fout << (char)temp;
62      }
63
64      fin.close();
65      fout.close();
66
67      remove("encrypt.txt"); //after original file has been encrypted, encrypted file removed
68      cout << "File successfully encrypted...\n";
69  }
```

Figure 4: Encrypt function lines 42-69



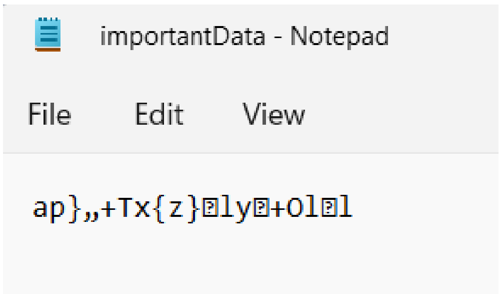
```
Select which process to run:
1. Encryption/Decryption
2. Ransomware Note
1
Running encryption process...

What would you like to do?:
1. Encrypt file
2. Decrypt file
1
Hello, Paddy!
Enter a number for the substitution cipher: 11
File successfully encrypted...
```

Figure 5: Running encryption process



```
Very Important Data
```

Figure 6: data before encryption

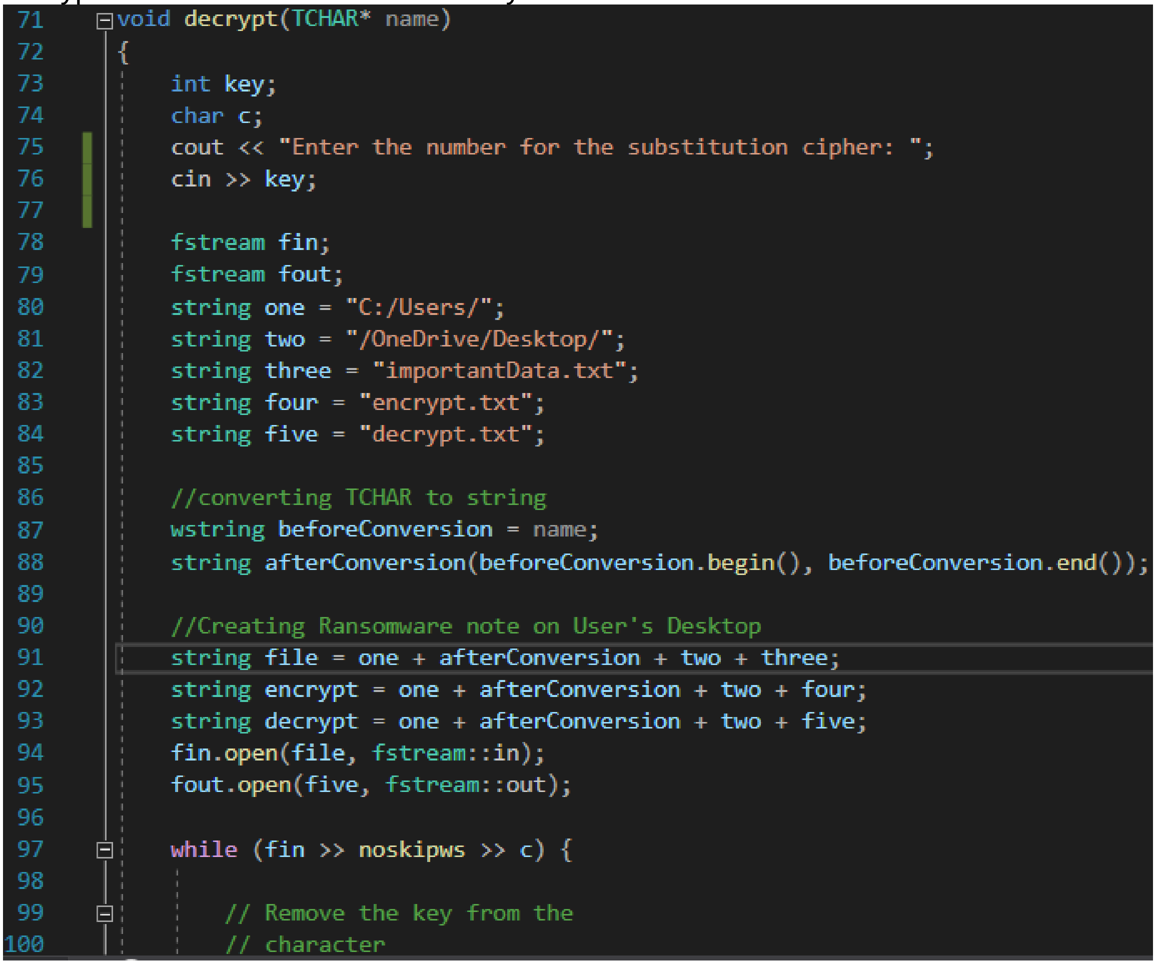


```
ap}„+Tx{z}▯ly▯+Ol▯l
```

Figure 7: Data after encryption

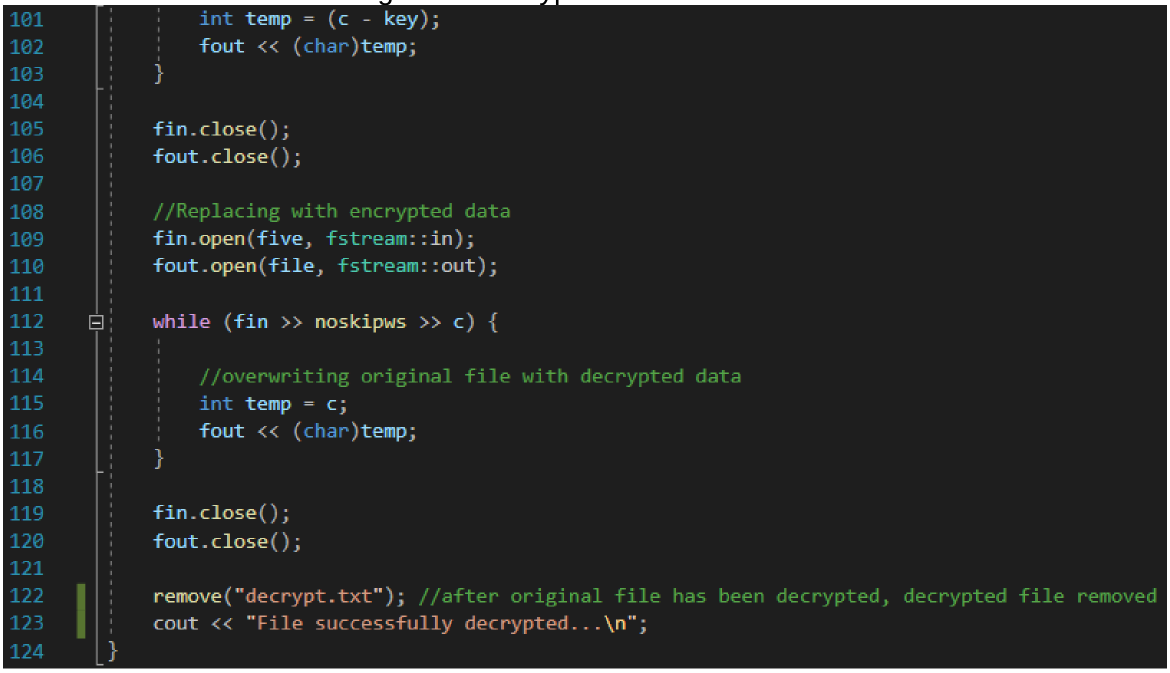
Decryption

To decrypt the process is simply reversed. The encrypted data is decrypted, placed into a text file and then the main file overwritten again with the decrypted data. After this the decrypted text file is deleted from the system.



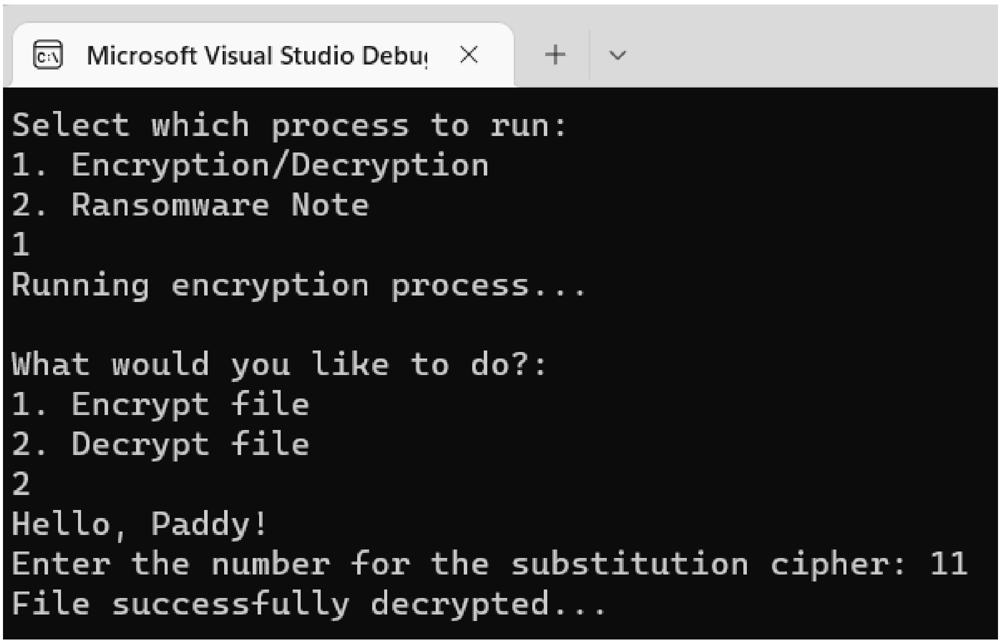
```
71  void decrypt(TCHAR* name)
72  {
73      int key;
74      char c;
75      cout << "Enter the number for the substitution cipher: ";
76      cin >> key;
77  
78      fstream fin;
79      fstream fout;
80      string one = "C:/Users/";
81      string two = "/OneDrive/Desktop/";
82      string three = "importantData.txt";
83      string four = "encrypt.txt";
84      string five = "decrypt.txt";
85  
86      //converting TCHAR to string
87      wstring beforeConversion = name;
88      string afterConversion(beforeConversion.begin(), beforeConversion.end());
89  
90      //Creating Ransomware note on User's Desktop
91      string file = one + afterConversion + two + three;
92      string encrypt = one + afterConversion + two + four;
93      string decrypt = one + afterConversion + two + five;
94      fin.open(file, fstream::in);
95      fout.open(five, fstream::out);
96  
97      while (fin >> noskipws >> c) {
98  
99          // Remove the key from the
100         // character
```

Figure 8: Decrypt function lines 71-100



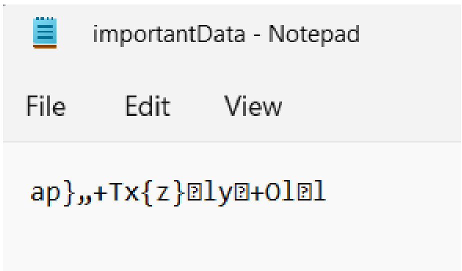
```
101         int temp = (c - key);
102         fout << (char)temp;
103     }
104 
105     fin.close();
106     fout.close();
107 
108     //Replacing with encrypted data
109     fin.open(five, fstream::in);
110     fout.open(file, fstream::out);
111 
112     while (fin >> noskipws >> c) {
113 
114         //overwriting original file with decrypted data
115         int temp = c;
116         fout << (char)temp;
117     }
118 
119     fin.close();
120     fout.close();
121 
122     remove("decrypt.txt"); //after original file has been decrypted, decrypted file removed
123     cout << "File successfully decrypted...\n";
124 }
```

Figure 9: Decrypt function lines 101-124



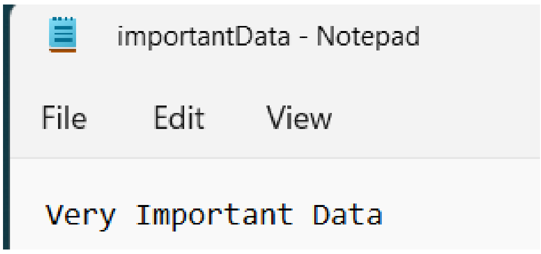
```
Select which process to run:
1. Encryption/Decryption
2. Ransomware Note
1
Running encryption process...

What would you like to do?:
1. Encrypt file
2. Decrypt file
2
Hello, Paddy!
Enter the number for the substitution cipher: 11
File successfully decrypted...
```

Figure 10: Running decryption process



```
ap}„+Tx{z}⍰ly⍰+Ol⍰l
```

Figure 11: Data before decryption

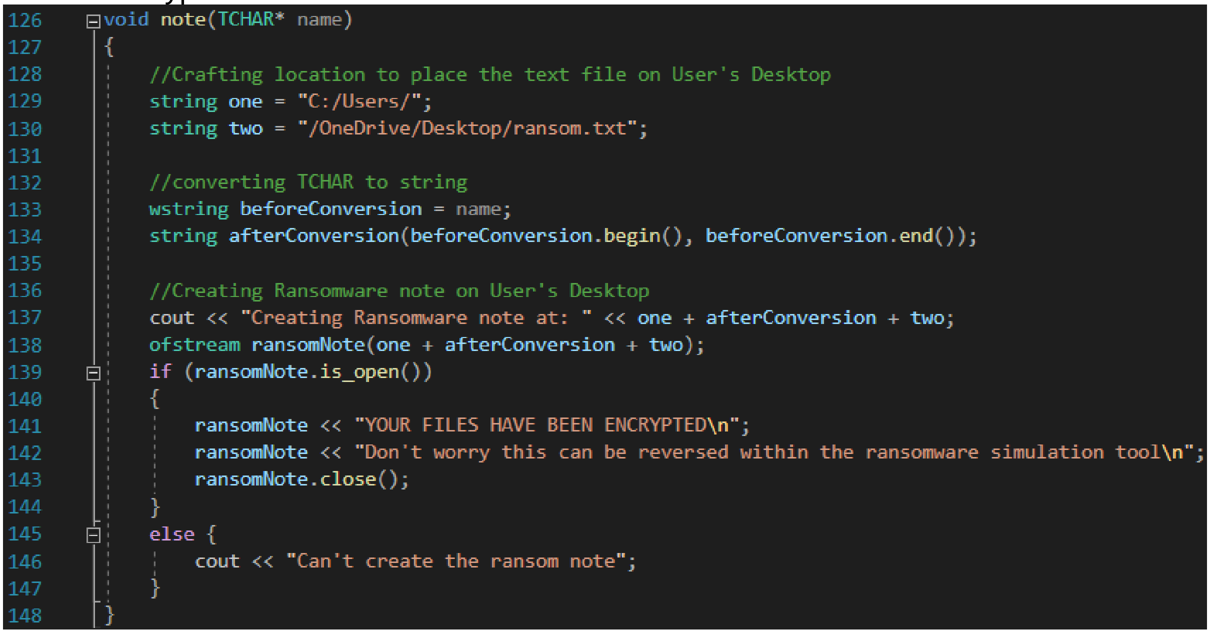


```
Very Important Data
```

Figure 12: Data after decryption

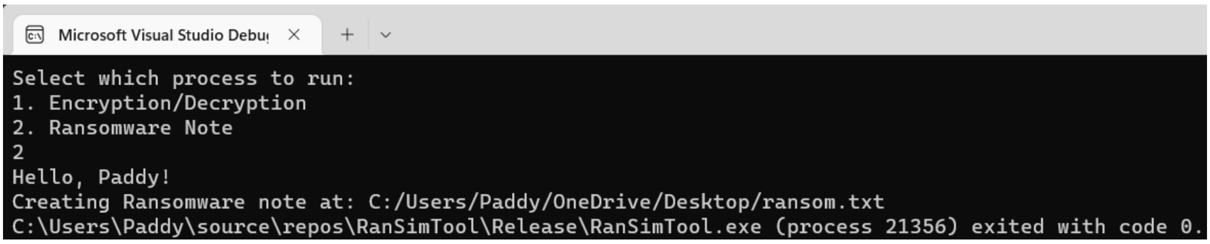
Ransomware Note

A simple ransomware note is placed on the User's Desktop with the contents being "YOUR FILES HAVE BEEN ENCRYPTED" matching the current language used in live ransomware samples. A message lets the user know the data can be decrypted using the tool again by selecting the Decrypt function and entering the number of the substitution cipher previously used to encrypt the data.



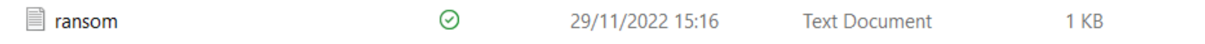
```cpp
void note(TCHAR* name)
{
    //Crafting location to place the text file on User's Desktop
    string one = "C:/Users/";
    string two = "/OneDrive/Desktop/ransom.txt";

    //converting TCHAR to string
    wstring beforeConversion = name;
    string afterConversion(beforeConversion.begin(), beforeConversion.end());

    //Creating Ransomware note on User's Desktop
    cout << "Creating Ransomware note at: " << one + afterConversion + two;
    ofstream ransomNote(one + afterConversion + two);
    if (ransomNote.is_open())
    {
        ransomNote << "YOUR FILES HAVE BEEN ENCRYPTED\n";
        ransomNote << "Don't worry this can be reversed within the ransomware simulation tool\n";
        ransomNote.close();
    }
    else {
        cout << "Can't create the ransom note";
    }
}
```

Figure 13: Ransomware note function



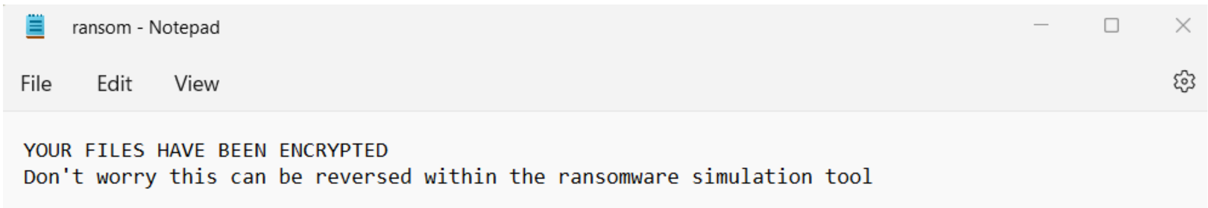
```
Select which process to run:
1. Encryption/Decryption
2. Ransomware Note
2
Hello, Paddy!
Creating Ransomware note at: C:/Users/Paddy/OneDrive/Desktop/ransom.txt
C:\Users\Paddy\source\repos\RanSimTool\Release\RanSimTool.exe (process 21356) exited with code 0.
```

Figure 14: Running ransomware note process



ransom   29/11/2022 15:16   Text Document   1 KB

Figure 15: ransom note in Desktop directory



```
YOUR FILES HAVE BEEN ENCRYPTED
Don't worry this can be reversed within the ransomware simulation tool
```

Figure 16: Ransom note contents

References

GeeksForGeeks and baljeet11801868 (2021) *Encrypt and decrypt text file using C++*, *GeeksforGeeks*. Available at: https://www.geeksforgeeks.org/encrypt-and-decrypt-text-file-using-cpp/ [Accessed: November 29, 2022].

Stevewhims (2021) *Getting system information - win32 apps*, *Win32 apps* | *Microsoft Learn*. Available at: https://learn.microsoft.com/en-us/windows/win32/sysinfo/getting-system-information [Accessed: November 29, 2022].