



## Protection of the Texts Using Base64 and MD5

Mohammad A. Ahmad<sup>1,a</sup>, Imad Fakhri Al Shaikhli<sup>1,b</sup>, Hanady Mohammad Ahmad<sup>2,c</sup>

<sup>1</sup> Department of Computer Science, International Islamic University Malaysia, Malaysia

<sup>2</sup> Department of Computer, Basic Education College, Public Authority of Applied Education and Training, Kuwait

<sup>a</sup>malahmads@yahoo.com, <sup>b</sup>imadyaseen39@yahoo.com  
<sup>c</sup>hanadym.1359@windowslive.com

### Article Info

Received: 1<sup>st</sup> February 2012  
Accepted: 22<sup>nd</sup> February 2012  
Published online: 1<sup>st</sup> March 2012

ISSN: 2231-8852

© 2012 Design for Scientific Renaissance All rights reserved

### ABSTRACT

The encryption process combines mathematics and computer science. Cryptography consists of a set of algorithms and techniques to convert the data into another form so that the contents are unreadable and unexplainable to anyone who does not have the authority to read or write on these data. The main objective of the use of encryption algorithms is to protect data and information in order to achieve privacy. This paper discusses an encryption method using base64, which is a set of encoding schemes that convert the same binary data to the form of a series of ASCII code. Also, The MD5 hash function is used to hash the encrypted file performed by Base64. As an example for the two protection mechanisms, Arabic letters are used to represent the texts. So using the two protection methods together will increase the security level for protecting the data.

**Keywords:** Encryption, Decryption, Base64, MD5

### 1. Introduction

The encryption process combines mathematics and computer science. Cryptography consists of a set of algorithms and techniques to convert the data into another form so that the contents are unreadable and unexplainable to anyone who does not have the authority to read or write on these data. The main objective of the use of encryption algorithms is to protect data and information in order to achieve privacy. The protection mechanism choices are applied based on the data sensitivity. For example, the data bank “ex, clients accounts” needs to be protected by latest security and protection mechanisms. In fact, with the available tools for intrusion in the internet today, computer intruders can hack to secure systems easily. Consequently, combining more than

one protection mechanisms is so crucial to achieve the highest level security against intruders. (Alshaikhli, 2011)

There are several functions for the protection processes to protect the information and files from intrusion. It is possible to employ encryption in various fields. In this paper, an encryption method is presented to protect the texts. One way to protect the texts from changes is by encryption. This paper will explain the method of encryption using base64. The first step of the encryption method using base64 is to convert text to unreadable text and create the ASCII for each character and convert it to a binary number. Then we convert the binary number to a decimal number and find the character that corresponds to the decimal number, and in so doing, the text will be rendered incomprehensible by the encryption process. Also, Message Digest 5 “MD5” is used as protection mechanism associated with Base64 encryption method. MD5 is an algorithm that is used to verify data integrity through the creation of a 128-bit message digest from data input (which may be a message of any length); the product is claimed to be as unique to that specific data as a fingerprint is to the specific individual. (Rivest, 1992)

MD5 and Base64 are used together to increase the security level of the data that needs protection. The details are explained in this paper.

## **2. Proposed System**

Computer security is a major challenge for all computer users, and use of encryption protects data and information from modification. Many businessmen, professionals, and home users employ encryption to protect their data and to maintain strict confidentiality. The system proposed in this paper is to encrypt the texts through the use of the Visual Basic program, as well as the use of encryption method of Base64 and hash function MD5.

The particular choices for the 64 characters required for the base varies between implementations. The general rule is to choose a set of 64 characters that is both part of a subset common to most encodings, and also printable. This combination leaves the data unlikely to be modified in transit through information systems, such as email, that were traditionally not 8-bit clean. For example, MIME’s base64 implementation uses A–Z, a–z, and 0–9 for the first 62 values. Other variations, usually derived from Base64, share this property but differ in the symbols chosen for the last two values; an example is UTF-7.

### **2.1 Hash MD5**

#### **2.1.1 Definition of Hash MD5**

MD5 is an algorithm that is used to verify data integrity through the creation of a 128-bit message digest from data input (which may be a message of any length); the product is claimed to be as unique to that specific data as a fingerprint is to the specific individual. MD5, which was developed by Professor Ronald L. Rivest of MIT, is intended for use with digital signature applications, which require that large files must be compressed by a secure method before being encrypted with a secret key, under a public key cryptosystem. MD5 is currently a standard,

Internet Engineering Task Force (IETF) Request for Comments (RFC) 1321. According to the standard, it is “computationally infeasible” that any two messages that have been input to the MD5 algorithm could have as the output the same message digest, or that a false message could be created through appropriation of the message digest. MD5 is the third message digest algorithm created by Rivest. All three (the others are MD2 and MD4) have similar structures, but MD2 was optimized for 8-bit machines, in comparison with the two later formulas, which are optimized for 32-bit machines. The MD5 algorithm is an extension of MD4, which the critical review found to be fast, but possibly not absolutely secure. By comparison, MD5 is not quite as fast as the MD4 algorithm, but offers much more assurance of data security. (Rivest, 1992)

### 2.1.2 The Algorithm of Hash MD5.

MD5 processes a variable-length message into a fixed-length output of 128 bits. The input message is broken up into chunks of 512-bit blocks (sixteen 32-bit little endian integers); the message is padded so that its length is divisible by 512. The padding works as follows: first a single bit, 1, is appended to the end of the message. This is followed by as many zeroes as are required to bring the length of the message up to 64 bits fewer than a multiple of 512. The remaining bits are filled up with a 64-bit little endian integer representing the length of the original message in bits.

The main MD5 algorithm operates on a 128-bit state, divided into four 32-bit words, denoted A, B, C, and D. These are initialized to certain fixed constants. The main algorithm then operates on each 512-bit message block in turn, each block modifying the state. The processing of a message block consists of four similar stages, termed rounds; each round is composed of 16 similar operations based on a non-linear function F, modular addition, and left rotation. There are four possible functions F; a different one is used in each round:

$$F(X, Y, Z) = (X \wedge Y) \vee (\neg X \wedge Z)$$

$$G(X, Y, Z) = (X \wedge Z) \vee (Y \wedge \neg Z)$$

$$H(X, Y, Z) = X \oplus Y \oplus Z$$

$$I(X, Y, Z) = Y \oplus (X \vee \neg Z)$$

$\oplus$ ,  $\wedge$ ,  $\vee$ ,  $\neg$  denote the XOR, AND, OR, and NOT operations respectively. (Rivest, 1992)

### 2.2 The Technique of Base64

The base64 method is used to protect the text and files from changes and that is discussed in this paper (Baccala, 1997). The base64 method involves finding all the ASCII characters, converting them to binary numbers, and then dividing the binary number for the text to 6 bits and converting them to their corresponding values in base64.

### 2.2.1 Base64 Mechanism

To encrypt this line using Base64:

- الحمد لله رب العالمين

1. First find the ASCII code for each character, Table 1.

Table 1: ASCII code

Letter	ASCII
ا	199
ل	225
ح	205

2. Second, convert the ASCII number of the characters to a binary number.

Table 2: ASCII to Binary Conversion

Letter	ASCII	Binary
ا	199	11000111
ل	225	11100001
ح	205	11001101

3. Third, divide the Binary number to parts and identify a number of bits so that the total is less than or equal to 64 bits in this example, the Binary number divided to 6-bit.

Table 3: Binary Division into parts

Letter	ASCII	Binary	Divided binary
ا	199	11000111	11000111
ل	225	11100001	11100001
ح	205	11001101	11001101

4. Fourth, convert parts of the binary number, which has been divided into a decimal number.

Table 4: Conversion binary parts to decimal

Letter	Divided binary	Index
ا	110001	49
ل	111110	62
ح	000111	7
	001101	13

5. Next, find the character (Char) that corresponds to the number (Value) in the Index Table below.

Table 5: Index Table

Value	Char	Value	Char	Value	Char	Value	Char
0	A	16	Q	32	g	48	W
1	B	17	R	33	h	49	X
2	C	18	S	34	i	50	Y
3	D	19	T	35	j	51	Z
4	E	20	U	36	k	52	0
5	F	21	V	37	l	53	1
6	G	22	W	38	m	54	2
7	H	23	X	39	n	55	3
8	I	24	Y	40	o	56	4
9	J	25	Z	41	p	57	5
10	K	26	a	42	q	58	6
11	L	27	b	43	r	59	7
12	M	28	c	44	s	60	8
13	N	29	d	45	T	61	9
14	O	30	e	46	U	62	+
15	P	31	f	47	V	63	/

Letter	Divided binary	Index	Base64-encoded
l	110001	49	x
U	111110	62	+
z	000111	7	H
	001101	13	N

6. Then perform the encryption for the letters.

Table 6: Encryption

Letter	Base64-encoded
ا	x
ل	+
ح	H
	N

$x+HN$                       ← الح

### 2.2.2 The Steps of Encrypting the Text

Table 7: Text encryption

Letter	Ascii	Binary	Divided binary	Index	Base64-encoded
ا	199	11000111	110001	49	x
ل	225	11100001	111110	62	+
ح	205	11001101	000111	7	H
م	227	11100011	001101	13	N
د	207	11001111	111000	56	4
ل	225	11100001	111100	60	8
ل	225	11100001	111111	63	/
هـ	229	11100101	100001	33	h
" "	32	00100000	111000	56	4
ر	209	11010001	011110	30	e
ب	200	11001000	010100	20	U
" "	32	00100000	100000	32	g
ا	199	11000111	110100	52	0
ل	225	11100001	011100	28	c
ع	218	11011010	100000	32	g
ا	199	11000111	100000	32	g
ل	225	11100001	110001	49	x
م	227	11100011	111110	62	+
ي	237	11101101	000111	7	H
ن	228	11100100	011010	26	a
			110001	49	x
			111110	62	+
			000111	7	H
			100011	35	j
			111011	59	7
			011110	30	e
			010000	16	Q

### 3. Conclusion and Future Work

This paper presented a method of base64 encryption to protect the text from being changed. The most important points raised by the paper include:

1. Use of encryption to protect of the texts from modification.
2. Use of a base64 encryption method, which relies on finding the ASCII for each character, converting them to binary numbers, then dividing them into a number of bits and converting them to their corresponding values in base64.
3. Use of the Visual Basic program for the application program.
4. Use of MD5 hash function for more security so that each file has its own number. When any change occurs in the files, it will change the number and the user will know that is not the same file.

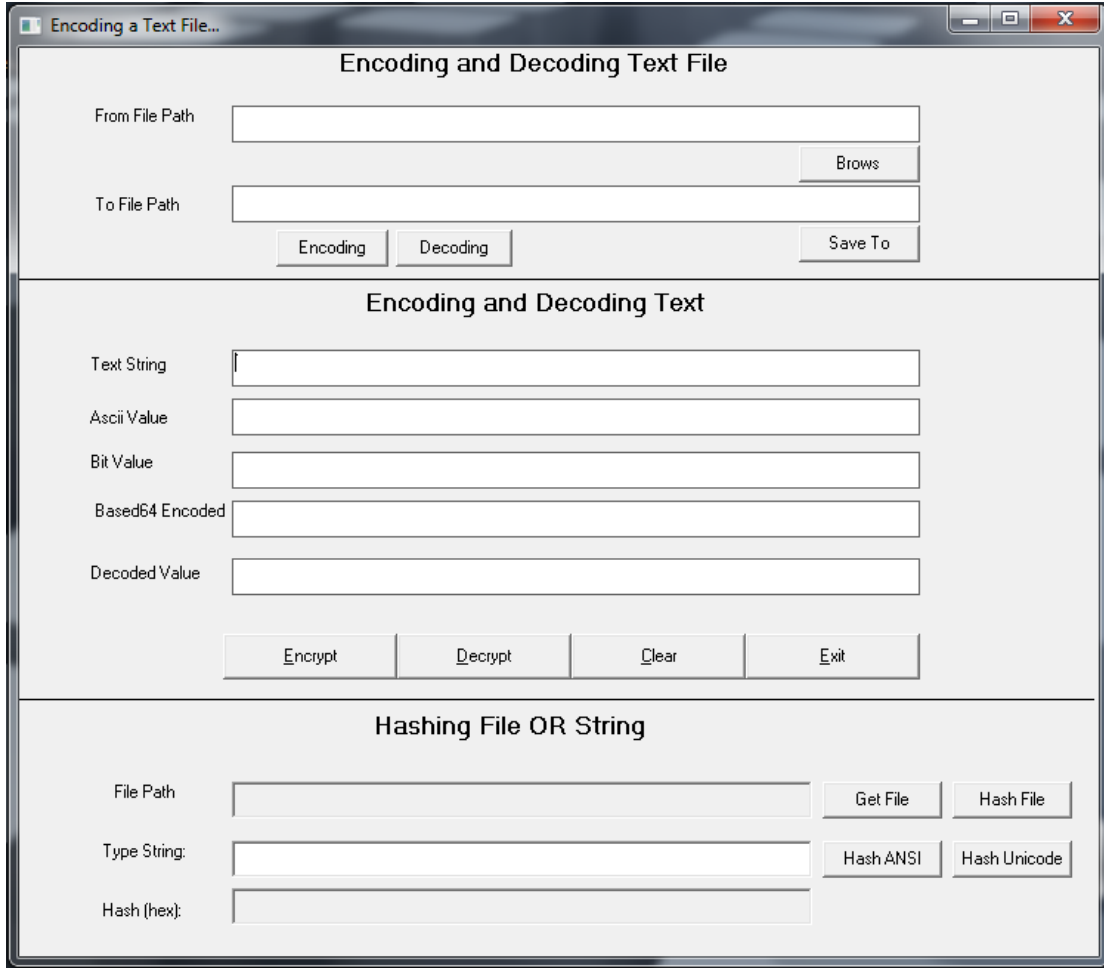
In the future, this work will be applied to protect the electronic Holy Quran from being tampered, changed or modified. More precisely, this paper is the first stage of other series of papers that will lead to a complete project of protecting the different formats of the Holy Quran.

### References

- Binark, I., Eren, H., & İhsanoğlu, E. (1986). World bibliography of translations of the meanings of the Holy Qur'an: printed translations, 1515-1980 (Vol. 1): Research Centre for Islamic History, Art, and Culture.
- Blaze, M., & Keromytis, A. D. (2000). DSA and RSA key and signature encoding for the KeyNote trust management system.
- definition MD5. (2011, march 12). retrieved from <http://searchsecurity.techtarget.com/definition/MD5>
- Den Boer, B., & Bosselaers, A. (1994). Collisions for the compression function of MD5.
- Imad F. Alshaikhli, M. A. A. (2011). Security Threats of Finger Print Biometric in Network System Environment. [Journal]. Advanced Computer Science and Technology Research, 1(1), 15.
- Josefsson, S. (2006). The base16, base32, and base64 data encodings.
- Klima, V. (2006). Tunnels in hash functions: MD5 collisions within a minute.
- Morin, R. C. (2001). How to base64.
- Quran, H., & Ahmad-UK, F. (1996). Al Islam. The Review of Religions.
- Rivest, R. (1992). The MD5 message-digest algorithm.
- Touch, J. D. (1995). Performance analysis of MD5. ACM SIGCOMM Computer Communication Review, 25(4), 77-86.
- Tuszynski, J. (2008). caTools: Tools: moving window statistics, GIF, Base64, ROC AUC, etc. R package version, 1.
- Wang, X., & Yu, H. (2005). How to break MD5 and other hash functions. Advances in Cryptology–EUROCRYPT 2005, 561-561.



## Appendix A

### The Components of the Program





**Fig. 1.** Screenshot of the “Encoding a Text File” Dialog Box

**Table 8:** Illustrating the function of the Browse file button


	<p>The name of this button is cmBrows and it is used to browse to the file that needs to be encrypted.</p>
	<p>The name of this text box is txtpath. When you choose the file that needs to be encrypted, the path and filename will be written to txtpath.</p>




**Table 9:** Illustrating the function of the Save To file button

	<p>The name of this button is cmdPathTo. It is used to save the encrypted text to the file whose path is written in textpathTo.</p>
	<p>The name of this text box is cmdPathTo. The path that the encrypting file is saved to will be written to cmdpath.</p>


**Table10:** Illustrating the function of the Clear button

	<p>The name of this button is cmdClear, and it is used to clear all text boxes.</p>
---	---

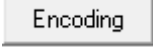
**Table 11:** Illustrating the function of the Decrypt button

	<p>The name of this button is cmdDecrypt, and it is used to decrypt the text written to text4.</p>
---	--


**Table 12:** Illustrating the function of the Encrypt button

	<p>The name of this button is cmdEncrypt, and it is used to encrypt the text written to text1.</p>
---	--


**Table 13:** Illustrating the function of the Encoding button

	<p>The name of this button is Command1, and it is used to encode the text in the selected file.</p>
---	---

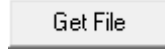

**Table 14:** Illustrating the function of the Decoding button.

	<p>The name of this button is Command2, and it is used to decode the text that encrypted and write it in other file.</p>
---	--

**Table 15:** Illustrating the function of the Exit button.

	The name of this button is command3. Use it to exit from the program.
---	---

**Table16:** Components for file hashing in the program

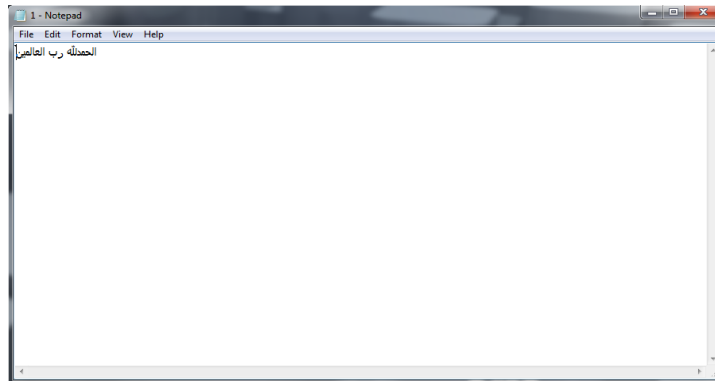
	Use this button to select the file to hash.
	Use this button to apply hashing for the selected file .

## Appendix B

### The Program Work

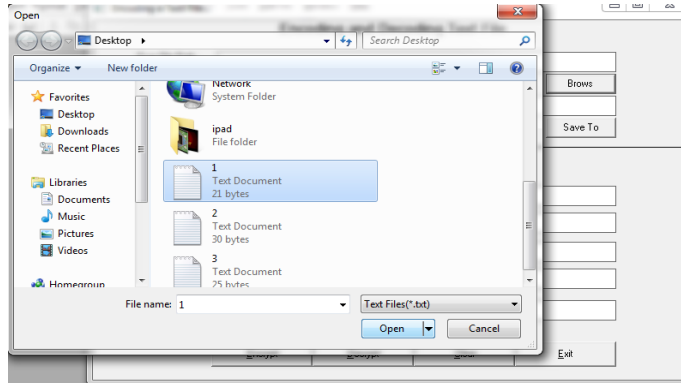
#### - How to Encrypt

As shown in Fig.2 below, the file that needs to be encrypted consists of plain text.



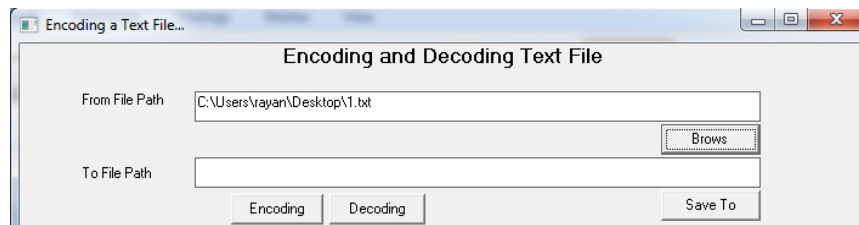
**Fig. 2.** Plain text file

As shown in Fig.3, select the file that you want to encrypt by pressing the Browse button.



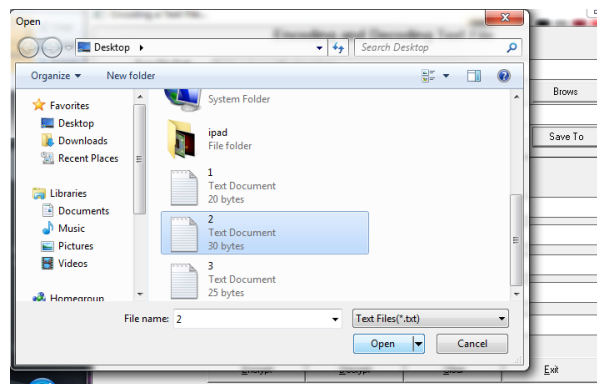
**Fig. 3.** Browse button dialog box

As shown in Fig.4, the path of the file that needs to be encrypted is entered in the From File Path field.



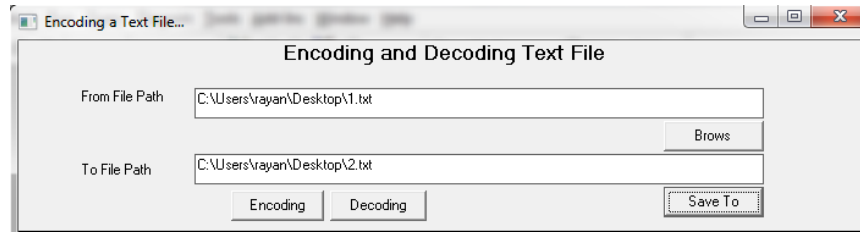
**Fig. 4.** Specifying the browse path

As shown in Fig.5, select the file in which you want to save the encoded text.



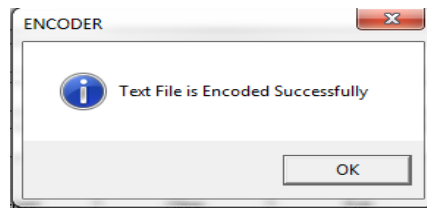
**Fig. 5.** Save As dialog box

As shown in Fig.6, the path of the saved file is entered in the To File Path field.



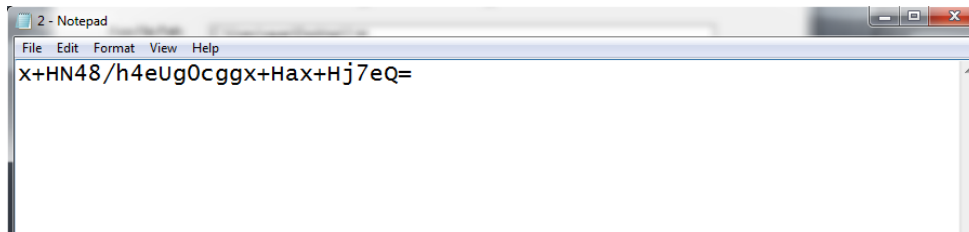
**Fig. 6.** Saving the text file to a given file path

When encryption is finished, the user will see a message box like the one in Fig.7.



**Fig. 7.** Encoder message box

After encoding the text in file 1 (see Figure 2), the encrypted result is saved in file 2, as shown in Figure 8.

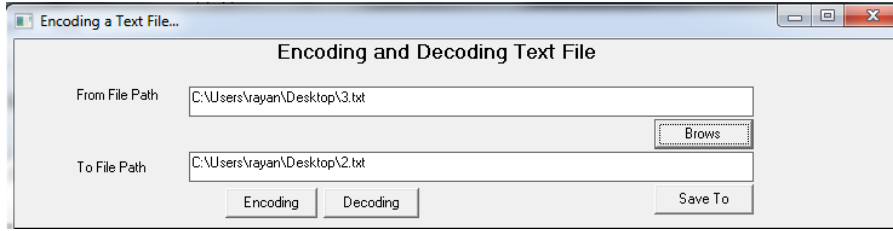


**Fig. 8.** Encrypting the file

### - How to Decrypt

There are several steps necessary to decrypt a file that has been encrypted previously using this method. Figure 8 shows the encrypted file that needs to be decrypted.

To perform the decryption process, browse to the file in the To Path File field, then indicate where the decrypted text will be saved in the From Path File field, as shown in Figure 9.

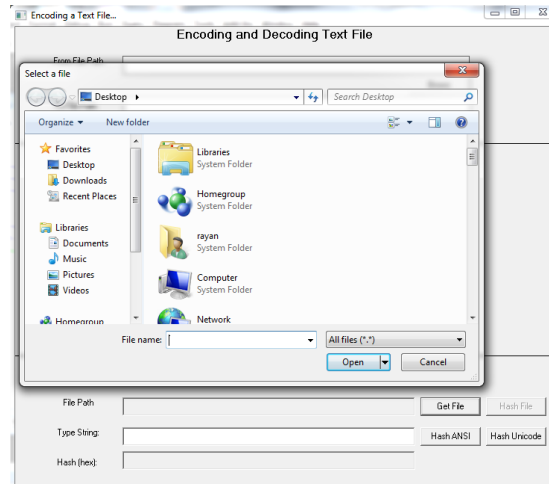


**Fig. 9.** Specifying the path of the file to decrypt and where the file will be saved

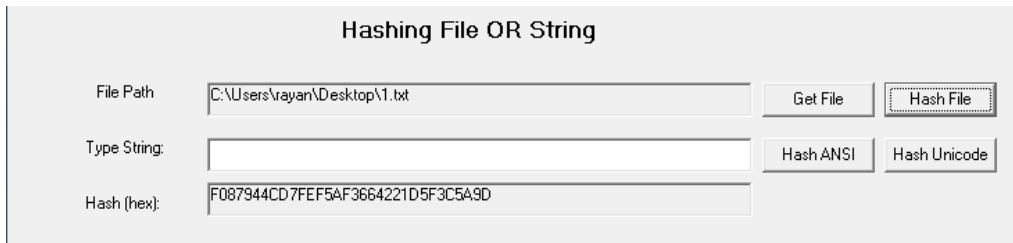
When decryption is finished, the user will see a message box like that shown in Figure 7. After that, the decrypted text will appear and can be read as shown in Figure 2.

### - How to Hash the File

As shown in Figure 10, select the file that you want to hash.



**Fig. 10.** Browsing for the file to be hashed As shown in figure 11, after selecting the file path using the Get File button, the user can click the Hash File button to generate the hashed hexadecimal code for the file.



**Fig. 11.** The hash for the file

Note: you can download this application from the URL <http://quran.alahmad.net/>