INFORMATION SECURITY APPLICATION DEVELOPED USING CRYPTOLOGY AND STEGANOGRAPHY TECHNIQUES

KRİPTOLOJİ VE STEGANOGRAFİ TEKNİKLERİ İLE GELİŞTİRİLMİŞ BİR BİLGİ GÜVENLİĞİ UYGULAMASI

Murat AYDOĞAN*

Dr., Bingol University, Department of Computer Technologies, Bingol, Turkey

Engin AVCI

Prof. Dr., Fırat University, Department of Software Engineering, Elazig, Turkey

**Corresponding Author: maydogan@bingol.edu.tr*

Geliş Tarihi / Received: 29.11.2020
Kabul Tarihi / Accepted: 08.12.2020

Araştırma Makalesi/Research Article
DOI: 10.38065/euroasiaorg.384

ÖZET

Teknolojinin gelişmesiyle birlikte dijital ortamların kullanımı her geçen gün artmış ve günümüzde çok yüksek boyutlara ulaşmıştır. Artık her kullanıcının kendi ihtiyacına göre bir şekilde kullandığı bu platformlar üzerinde kullanım sayısına bağlı olarak gün geçtikçe güvenlik açıkları meydana gelmiştir. Günümüzde bilgisayar teknolojilerinin yaygın olarak kullanılmaya başlanmasıyla birlikte bilgi güvenliğine olan ihtiyaçta önemli derecede artmıştır. Bu çalışma güvenlik konusuyla yakından alakalı olan kriptoloji ve steganografi tekniklerinden faydalanılarak bilgi güvenliğinin korunmasına katkı sağlamak amacıyla yapılmıştır. Literatürde genellikle bilgi güvenliğinin sağlanması için birbirlerinin alternatifi olarak gösterilen veri şifreleme ve veri gizleme konuları, bu çalışmada birlikte kullanılmış ancak bu kez alternatif olarak değil birbirlerinin tamamlayıcısı olmuşlardır. Bu bağlamda, kriptoloji ve steganografi tekniklerinden faydalanılarak bilgi güvenliğinin artırılmasına önemli bir katkı sağlayacağı düşünülen yeni ve özgün bir uygulama geliştirilmiştir. Geliştirilen uygulama ile sayısal resim içerisine gizlenen verilerin her defasında farklı ve rasgele noktalara gizlendiği ardından gizlenmiş verilerin başarıyla deşifre edilerek orijinal verilere ulaşıldığı görülmüştür. Uygulama C# programlama dili ile geliştirilmiştir.

Anahtar Kelimeler: Bilgi Güvenliği, Kriptoloji, Veri Şifreleme, Steganografi, Veri Gizleme

ABSTRACT

The use of digital platforms popularized as a result of technological developments and reached very high points recently. However, these platforms that are used by many people to meet their needs are open to security vulnerabilities due to the number of uses they get. The need for information security has increased significantly since computer technologies are widely used today. This study aims to contribute to the protection of information security using cryptology and steganography techniques that are closely linked with security issues. In this study, data encryption and data hiding, which are generally thought of as alternatives, are used as supplements of each other. Thus, a new and unique application, which is thought to contribute to increasing information security, was developed using cryptology and steganography techniques. It is observed that with the usage of the new application, the data embedded in a digital image that is hidden to different and randomly selected positions each time can be decrypted successfully to obtain the original data. The application was developed using the C# programming language

Keywords: Information Security, Cryptology, Data Encryption, Steganography, Data Hiding

I. INTRODUCTION

Security is one of the most crucial needs of humanity from past to present. In today's world, this need has increased, and the scientific world takes a close interest in this matter (Tuncal, 2008).

Following the rapid development of computer technologies, computer systems' security, especially data security, has become crucial. The users' needs and information sharing increased in parallel with the development of the Internet. Users get the opportunity to shop, communicate, share different types of data regardless of time and space, thanks to the Internet (Buluş, 2006).

The fact the Internet is a common platform that facilitates the lives of users and is used by many people has, of course, caused security vulnerabilities. These vulnerabilities make the Internet quite attractive for ill-intentioned people who want to have illegal benefits (Ulutürk, 2010).

Although not authorized, not only can third parties be able to access the communication between two users who correspond over a shared network, but also the information in this environment can become changeable. Therefore, cryptology and steganography, in other words, data encryption and data hiding, are the concepts that emerged and developed rapidly; thenceforth, becoming highly popular (Kim et al, 2003).

II. METHODS

1. Cryptology

The security vulnerability is crucial when the constant development of today's technology is taken into consideration. Cryptology is an important discipline that is used in interpersonal communication or communication between government agencies, and it fills the security vulnerabilities in communications, and this branch of science has been used in different ways since the past (Bahçetepe, 2006; Sarıtaş 2010).

Cryptography (encryption) is carried out to prevent unauthorized people from using the data, or to prevent undetectable changes on the data, or to convert the data into a meaningless form. In this sense, cryptography is concerned with encrypting data and converting it back to its original format (Başkök, 2007).

1.1. Public-key Cryptography (Asymmetric cryptography)

Key distribution is a problem in symmetric encryption technique. Different keys are used for encryption and decryption. In this system, the encryption is carried out by using a public-key that is known to everyone. This technique is known as asymmetric cryptography since encryption and decryption are carried out using asymmetric algorithms (Denton, 2011). RSA encryption algorithm can be given as an example (Rivest et al, 1978).

1.2. Private-key Encryption (Symmetric Encryption)

A private key is used for encryption and decryption for symmetric encryption. After encryption, while the ciphertext is being sent to the receiver, the private-key should also be sent to the receiver safely. Symmetric encryption algorithms are fast. DES and AES encryption systems can be given as examples (Tsunoo et al, 2002; Çağlar, 2004).

2. Steganography

Steganography is one of the oldest methods of data hiding, dating back to Ancient Greece and Ancient Times. Digital steganography is generally divided into three in terms of usage. These are as follows (Şahin, 2007):

- Text steganography
- Image steganography

- Audio steganography

III. EXPERIMENTAL STUDIES

Crimes are being committed using identity fraud, causing many innocent people to suffer. The application developed to prevent fraud operates on the principle of first encrypting one's personal ID number, and hiding these numbers in the personal photographs on the national ID card of the person, finally reading the encrypted data through the photograph and verifying the national ID number of the person by comparing it to the one on the national ID card, respectively.

Data Encryption

Two different encryption algorithms, one being authentic, are used for data encryption. In the application, these algorithms are named as the First Encryption Step and Second Encryption Step. The encryption algorithms are explained in detail in the following sections; however, briefly, the first encryption step is a distinct encryption algorithm that is developed for this application specifically. The data encryption is completed once the data that were encrypted using the first algorithm are encrypted using the RSA encryption algorithm as a second step.

Step 1: Resequencing

While developing the encryption algorithm mentioned in the introduction, each of the 11 digits of Turkish ID number are encrypted and are handled one by one. The table, called the Character Set shown below is used for encrypting these digits (Table 1). The reason to prefer the Character Set to ASCII code is that each character of the set is entirely random, unlike ASCII code, whose values act according to a certain logic.

Table 1. Character Set

0	n	**16**	/	**32**	h	**48**	P	**64**	!	**80**	p
1	9	**17**	0	**33**	,	**49**	r	**65**	K	**81**	v
2	*	**18**	=	**34**	%	**50**	6	**66**	U	**82**	b
3	E	**19**	space	**35**	B	**51**	)	**67**	\	**83**	Y
4	d	**20**	m	**36**	ı	**52**	S	**68**	^	**84**	1
5	+	**21**	-	**37**	Ş	**53**	H	**69**	ü	**85**	"
6	O	**22**	z	**38**	'	**54**	i	**70**	C	**86**	&
7	R	**23**	Ğ	**39**	>	**55**	Ü	**71**	A	**87**	Ö
8	f	**24**	t	**40**	ö	**56**	_	**72**	G	**88**	u
9	;	**25**	y	**41**	o	**57**	s	**73**	Ç	**89**	<
10	Z	**26**	2	**42**	T	**58**	.	**74**	:	**90**	İ
11	e	**27**	L	**43**	5	**59**	7	**75**	V	**91**	ş
12	k	**28**	F	**44**	4	**60**	D	**76**	(	**92**	@
13	ç	**29**	N	**45**	M	**61**	c	**77**	8		
14	l	**30**	I	**46**	J	**62**	#	**78**	j		
15	a	**31**	?	**47**	ğ	**63**	3	**79**	g		

The number used in the application is a standard 11-digit number, just like the Turkish ID number, and determined as 29614534824. As it is stated in the introduction, the method to be used for encryption is to resequencing the digits one by one and hiding the numbers in the picture after the necessary steps are followed.

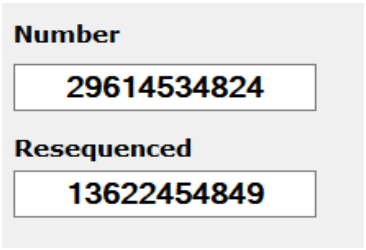

Figure 1. Random sequential Turkish ID number

The software randomly reordered the Turkish ID number determined as 29614534824, as 13622454849, as shown in Figure 1. After this step, the numerical expression to be used for encryption and decryption is 13622454849.

Table 2. Data values in Character Se

Number	Character Set
1	84
3	64
6	50
2	26
2	26
4	44
5	43
4	44
8	78
4	44
9	93

Step 2: Character Set

The exact equivalents of the sequenced number set in Table 2 were created using the Character Set. After this stage, where the application sets the new sequence, and the equivalent values of this new ID number, the equivalent values of each number is now as determined in the table.

In short, from now on, the Turkish ID number is no longer the group of numbers of the system, but it is a set of digits that is broken down one by one using the newly assigned values. The reason for not choosing ASCII code equivalents, at this stage, is that the ASCII code table follows a certain system, whereas the Character Set is designed based on randomness. For example, the ASCII equivalent of 0 (zero) is 48. The ASCII equivalents of numbers from zero to nine continues consecutively, and this is a problem in terms of password security. As stated before, the Character Set is preferred because the number set's code equivalents are chosen to be completely random, and therefore there is no closeness and systematicity among them.

Step 3: Control Bit

Each 11 digit of the Turkish ID number, encrypted in two steps using the developed encryption algorithm, was handled one by one and reordered. Then the equivalent values of each digit were identified using the Character Set developed for the application. The security enhancement process, the third step, is titled as the Control Bit step. The equivalents of each number in the Character Set were used in the Control Bit step. These values were transformed into the binary code system. Within the eight-bit system that consists of 1 and 0 values, the equivalent of the control bit was identified by determining the number of 1-values. Even though the importance and necessity of the control bit within the encrypted system will be elaborated in the section about decryption the data, here it can be said that the control bit checks whether the decrypted data is correct or not. The

control bit is identified using the binary value obtained. A new number is generated using the equivalent value in the Character Set and the control bit (Figure 2).

1. COLUMN	2. COLUMN	3. COLUMN	4. COLUMN
1	84	01010100	3
3	64	01000000	1
6	50	00110010	3
2	26	00011010	3
2	26	00011010	3
4	44	00101100	3
5	43	00101011	4
4	44	00101100	3
8	78	01001110	4
4	44	00101100	3
9	93	01011101	5

Figure 2. Number Transformations

After the identification of control bits, new numbers are generated using the identified values. For example, in the second step, the Character Set equivalent of number 1, which is the first number of the sequence obtained in the first step during which a number set is generated by resequencing the original Turkish ID numbers, is 84. The control bit control takes place in the third step. Thus, first, 84 is transformed into the binary system, and the control bit is generated by counting the amount of the bit value of 1 out of the 8-bit value. The control bit of number one is identified as number three after this process. The process of converting the obtained value into a meaningful number is done by writing the control bits between the equivalents of the original numbers in the Character Set. If this is to be done for number one, 834 would be obtained using 84, which is the equivalent of number one in the Character Set, and three, which is the control bit. It should be noted that the control bit value, number three, is written in the tens digit. Hence, a new three-digit number is generated. If the encryption algorithm is assessed, number one transformed into 84, first, then into 834. Therefore, 834 is the systemic equivalent of number one. Figure 3 presents the reordered and generated numbers after the control bits were calculated.

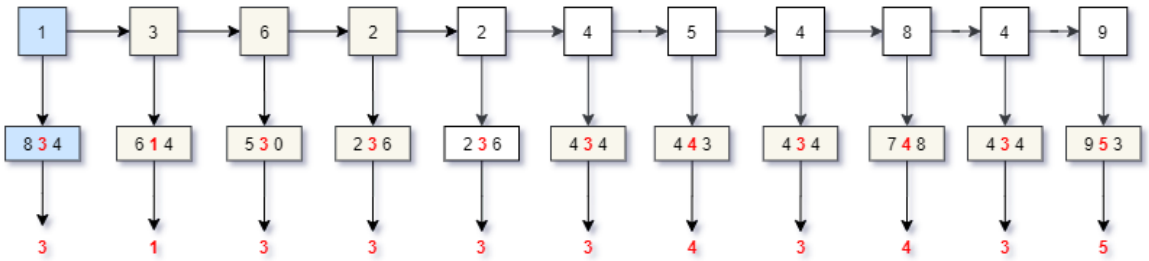

Figure 3. Control Bits

RSA Encryption Algorithm

In this application, where the RSA encryption algorithm is used, which is based on the generation of two large prime numbers and the mathematical operations applied based on the obtaining of private-

key and public-key, the P and Q numbers are first randomly generated by the software. Next, the private-key and public-key are obtained using the following algorithm.

1. The software randomly generated prime number 7013 as the P value and prime number 9871 as the Q value.
2. N key is generated by multiplying P and Q numbers. N key is 69225323, which is calculated by multiplying 7013 and 9871.
3. The value of φ (n) is calculated by multiplying minus 1 of the numerical values of numbers P and Q. Calculating the φ (n) value is important to generate E and D keys. The φ (n) value is calculated as 6908440 by multiplying (7013 -1) and (9871 –1).
4. The following is the method of identifying key E: The value for key E should be selected suitable for the 1<E<N format and the selected value must be a prime number between E and φ (n). As it is shown in the screenshot below, the system identified key E as 8779, which will be used for encrypting the values.
5. The key D to decipher the values is generated using the E*D≡1(mod φ (n)) formula. When the previously generated keys were used as their respective values within the formula, 21324619 was calculated as the result of 8779*D=1(mod 6908440).

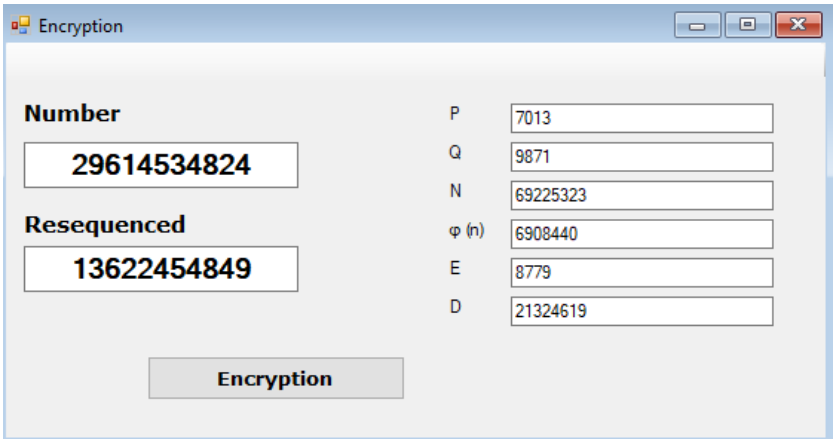

Figure 4. Key Generation Interface

The keys to be used for encrypting and decrypting the messages using the application developed in line with the RSA encryption algorithm, were generated as shown in Figure 4. Further, the method of rewriting the text to be encrypted in random sequence and encrypting the new text generated according to the new randomization was preferred to increase security.

Encrypting Data Using RSA

The secret-key and public-key to be used for encryption and decryption the data were identified after generating the keys in line with the RSA encryption algorithm. Encryption can be the first step after this process. As it can be seen in the following screenshot, the numbers were handled one by one as in the first encryption step. These numbers that are in the encrypted format generated as a result of the first step, are encrypted again in the second step using the RSA method.

$C = M^{e} \pmod{N}$ formula is used for the RSA encryption algorithm.

According to this formula, the M value is the message to be encrypted, E and N values are public-keys. 434 is the first step-encrypted version of number 4 in the sixth row; Figure 5 shows the 21681179 as a result after the operation $434^{8779} \pmod{69225323}$.

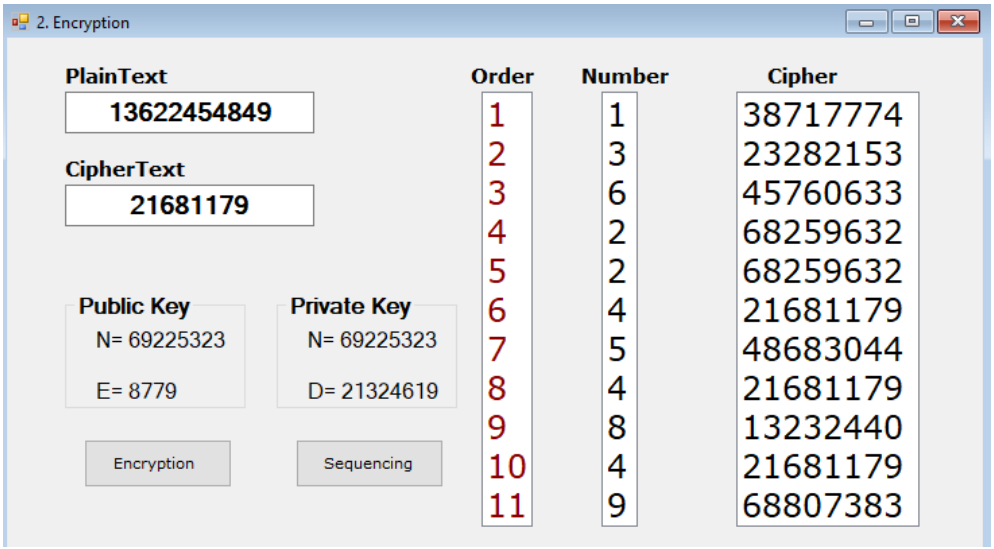

Figure 5. Encryption Interface

The Turkish ID numbers that were encrypted by random sequencing in the first encryption step were once again encrypted using the RSA encryption algorithm in the second encryption step.

Table 3. Encryption Steps

ORDER	RANDOM ORDER	1. ENCRYPTION	2. ENCRYPTION (RSA)
2	1	834	38717774
9	3	614	23282153
6	6	530	45760633
1	2	236	68259632
4	2	236	68259632
5	4	434	21681179
3	5	443	48683044
4	4	434	21681179
8	8	748	13232440
2	4	434	21681179
4	9	953	68807383

The first two columns of Table 3 show the original and randomized numbers; the third column shows the encrypted version of these numbers according to the first encryption step. In the last column, the encrypted values using the RSA encryption algorithm of the data that were encrypted in the first step are shown. This final step concludes the data encryption.

Data Hiding

The second module, data hiding, started after completing the two-step encryption, which is the data encryption step of the application. This section elaborates on hiding the encrypted data into passport-sized colored photographs used as an example in the Fraud Identity Detection application.

The most important superiority of the developed algorithm is randomized geolocation. In other words, it is the randomization of the address of the numbers that are already reordered; only the system knows which data is hidden in which address. As expected, the locations will be randomized and changed for each hiding. As it was explained in detail in the data encryption section, the

superiority and the advantage of the system is that the data were encrypted twice, one of which was by a powerful encryption algorithm RSA were embedded randomly into a photo to be decrypted again.

The standard passport photographs on the ID cards are 177x236. Since the developed application will be used for identifying fake ID cards, the photographs used on the data hiding step should fit these measurements. In order not to interfere with the data integrity of the 175x234 sized photographs, two columns, and two rows were drawn around the photograph; the new size of the picture is 177x236, as shown in Figure 6.

175 x 234 177 x 236

Figure 6. Part used for data hiding

Figure 7 presents the original pixel values of the passport photograph examples, whose dimensions were given above, to be used in the application before data hiding. Only the pixel values of a part of the photograph that is going to be used are given. Still, the same part was selected for the following processes, and the pixel values for the same part were presented so that the changes in the pixel values can be noted and compared.

18	17	21	16	10	16	14	14	13	16	14	15	13	13
13	14	16	15	20	15	13	13	14	14	12	14	13	12
13	14	15	15	18	16	13	12	13	14	13	12	15	10
12	15	17	15	14	13	13	14	14	11	11	11	15	14
11	14	15	15	12	15	14	12	13	13	15	16	9	11
14	15	13	14	14	16	13	14	14	17	13	14	13	11
17	16	14	14	13	15	13	13	13	14	15	12	11	16
13	19	13	11	11	14	14	12	13	12	14	9	10	10
14	14	12	13	12	12	13	13	13	17	11	10	11	7
15	15	14	13	13	13	11	16	14	11	14	15	15	14
11	12	14	11	11	13	16	15	15	14	11	16	13	14
13	15	14	12	14	14	14	16	17	20	14	14	13	13
18	15	16	15	16	18	20	18	20	23	20	15	13	14
19	16	15	15	16	18	17	19	19	22	18	15	12	11
20	13	14	16	15	15	13	16	18	19	17	12	12	12
19	17	18	17	12	12	9	15	17	14	14	14	14	12
18	17	16	14	11	14	11	15	13	15	11	12	12	10
17	19	16	14	16	16	15	16	14	15	14	15	10	11
18	14	17	16	17	16	15	17	17	18	16	16	13	14
18	15	16	17	17	17	13	15	15	15	16	15	15	15

Figure 7. Original Pixel Values

Step 1: It is the process of adding a frame of two rows and two columns on a passport photograph with standard dimensions of 175x234. The first process on the new photograph with the new dimensions of 177x236 is to color code the added rows and columns, namely the structure, referred to as the frame in the following sections.

Step 2: Using the frame starts after the color-coding in the range of 0-255. A part of the pixels in the upper side of the frame is determined as the pixels that are the hiding locations of the encrypted versions of the Turkish ID numbers and the keys. These locations were designed as follows:

- The first 10 pixels [0-10 range] were assigned as the hiding location of the N, E, and D key values.
- Next, the system identifies a number between 0 and 10, and a gap was created by skipping the same number of pixels. The random value was five in this application.
- Next, since an 11-digit encrypted number was to be hidden, 22 addressing points were assigned, where the abscissa and ordinate of the 11 numbers were located.

Step 3: First, the keys that are using the first 10 pixels should be hidden, because the keys are crucial for decryption after data hiding. Since the system randomly identifies the key values, as it was stated in the first sections, there is a high risk that the decrypted values do not match the original values if the key values are not recorded. This causes the system to malfunction.

If the assigning process is analyzed within the key hiding process, N=69225323, E=8779, and D=21324619 keys were identified according to the processes described above.

The assigning process is based on switching the values to be assigned with the original pixels in groups of two. The point to note here is to replace directly the original pixel if it has two digits, and the ones and tens digits are changed if it has three digits. If the generated value after the change surpasses 255, controls should be carried out since a problem might be encountered, and in case of a problem, the value on the hundred's digits should be changed.

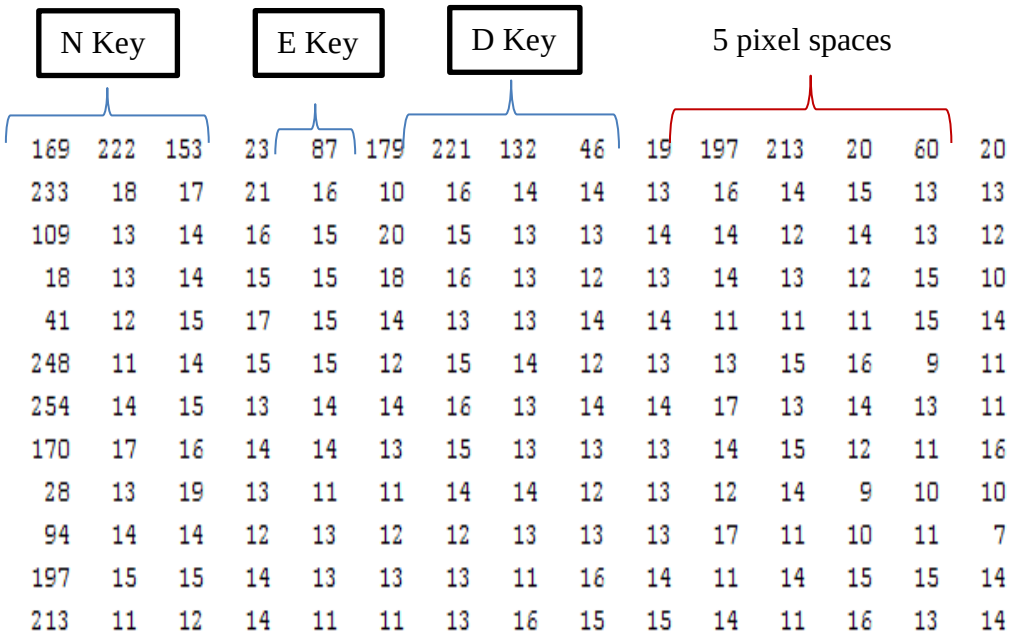

Figure 8. Key Hiding

The screenshot in Figure 8 shows the pixel values of the points located in the row, which are called the upper side of the frame. According to this, the first 10 pixels are used for hiding keys. As the first step, filling the values in the 0-255 range of the entire frame created the color tones in the range of black and white. Afterwards, the keys were placed and a gap in which a random value between 0 and 10, as it was mentioned in the second step, can fit was left; in other words, the pixels were not

assigned, the random values that were assigned to these addresses from the values between 0-255 that were randomly filled were kept.

Table 4. The Address and Values of Hidden Keys

N KEY			**E KEY**			**D KEY**	
ADDRESS	*VALUE*		*ADDRESS*	*VALUE*		*ADDRESS*	*VALUE*
(1,0)	**1**69		(5,0)	87		(7,0)	**2**21
(2,0)	**2**22		(6,0)	**1**79		(8,0)	**1**32
(3,0)	**1**53					(9,0)	46
(4,0)	23					(10,0)	19
69225323			**8779**			**21324619**	

Table 4 shows the address of the hiding points of N, E, and D keys, that is, expressions in the form of the x and y-axis and their values at these addresses. The full values of the keys are presented in the bottom row, and the colors show the hidden keys on the pixels.

Step 4: After the first 10 pixels in the upper row of the added frame and used to hide the keys, a total of 15 pixels were used by adding a random value as described in the previous steps. The system keeps the records of the encrypted versions of the 11 numbers that were described in detail in the data encryption section. The hiding of the encrypted version of 11 numbers should be carried out after completing the hiding of keys. As it was mentioned before, one of the most important advantages of the application is that the points in which the numbers will be hidden are not identified, and the system randomly assigns these points and that these can be decrypted again. Hence, the addresses in which the numbers will be hidden should be known and recorded by the system. Since there are 11 numbers to be hidden, 11 points should be identified on the x-axis, and 11 points should be identified on the y-axis. If the total would be called as the address, 22 points should be assigned as addresses. The upper row of the frame, the addressing line, was used for this assignment process, and 37 pixels in total were used as key values, security value (random value), and addressing points.

First, the addresses of the points where the numbers will be hidden must be defined in the process of hiding the encrypted 11-digits that constitute the Turkish ID number. As it was stated before, since the geolocation is randomized, the system randomly assigns addresses. After the frame is added, values in the range of 0-236 according to the size of the picture produced by the system are used as addressing points for the numbers to be hidden on the frame on the photograph, which is 177x236.

Step 5: This step is the hiding of the encrypted numbers in the frame placed on the photograph, after recording the key values and addresses of the numbers. Two principles are important at this point. The first one is that data should not be written on the addressing line, which consists of 37 pixels. Therefore, addressing was permitted on the entire frame, but the 37 pixels on the x-axis. The second point is to use four cells since the encrypted numbers consist of eight characters in total, and the pixel values are exchanged in groups of two as in the third step.

For data hiding, for example, if number one, the first character, is going to be hidden, the first step should be identifying the address of the location where the data will be hidden. A 22-cell addressing line, where the abscissa and ordinate values are located, should be identified. Since the number one is given as an example, it is known that the point (16,0), which is the abscissa of the first character in the first cell of the addressing line, and (27,0), which is the ordinate, pixels are assigned. Hence, it is concluded that the (161,235) point is the address of the first character on the photograph within this addressing algorithm.

If the process of encrypting data and hiding the number 1, the first character of the reordered ID number in the photograph, is analyzed in more detail;

Character one transformed into 834 after the first encryption algorithm, and this value was transformed into 38717774 using the RSA encryption algorithm, which is the second encryption step. Now, 38717774 is the equivalent of number 1 for the system. The hiding of this number is as follows:

38-71-77-74

(161,235) Point-> **188** is the original pixel value, 1**38** is the value obtained after the last two pixels are exchanged.

(162,235) Point--> **97** is the original pixel value, the changed value is **71**

(163,235) Point--> **220** is the original pixel value; however, 277 would be obtained after exchanging the last two digits and this value surpasses the upper limit of **255**. Hence, 100 is subtracted from the value to obtain 1**77** using the algorithm.

(164,235) Point-> **49** is the original value, **74** is the result.

Decryption Data

Decryption and transform the encrypted data hidden in the digital photograph into the original Turkish ID number, more information than the data hiding is needed. The most important data at this stage would be the key values stored in the first 10 pixels, and 22 pixels (11 abscissae and 11 ordinate) used for the addressing since the locations of the encrypted data are stored within the address data. After the data are encrypted using the address information, secret keys (N and D) should be known to decipher the data that were encrypted using the RSA encryption algorithm. The operations performed in 5 steps in total during the data hiding phase are decrypted with two basic steps in this section.

Step 1: Reading the Data

As the first step of the decryption process, the address where the data is hidden in the photograph should be found and this encrypted data should be read. The first thing to do for this is to identify the addressing line as in the data hiding step.

At this stage, the decryption of the 7^{th} character of the 11-digit Turkish ID number, number five, and number four, the 8^{th} character, can be given as an example.

After identifying the addressing line located within the first 22 pixels, the abscissa and ordinate of the 7^{th} and 8^{th} numbers should be identified. Since the first 15 pixels were reserved for key values, and the addressing starts from the 16^{th} pixel, the abscissa of the 7^{th} number is the 32nd pixel, and the abscissa of the 8^{th} number is the 33^{rd} pixel. The ordinate values are the 43^{rd} and 44^{th} pixels, respectively. Therefore, the address of the 7^{th} number would be (117,235) point using the 32^{nd} and 43^{rd} addressing pixels; and the address of the 8th number would be (111,235) point using the 33^{rd} and 44^{th} addressing pixels.

After the locations of the addresses where the numbers were hidden are determined, a process like the method applied in the data hiding step was followed. This time, data was used directly instead of being exchanged. Since it was known that the 7^{th} character was located at the point (111,235), once reached, what needs to be done was to move four more pixels to find 111, 112, 113, and 114 pixels located on the x-axis, as explained in the sections above. Similarly, the 8^{th} character was in 117, 118, 119, and 120 pixels.

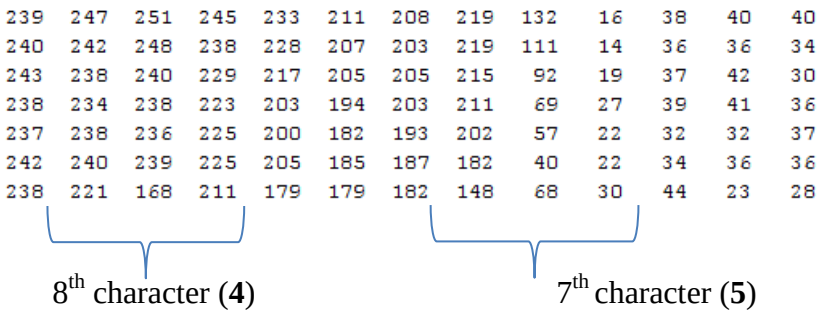

Figure 9. Data decryption

The cells that the 7th and 8th digits were hidden in are identified as shown in Figure 9. The details of the identified addresses can be found in the table below. When the screenshot above and the table are assessed together, the principles of data hiding, the logic of addressing, and data reading will be understood better.

7th Character		8th Character	
(117,235)	148	(111,235)	221
(118,235)	68	(112,235)	168
(119,235)	30	(113,235)	211
(120,235)	44	(114,235)	179

Figure 10. Reading Character Values

As it is shown in Figure 10, like the data hiding process, the encrypted data was obtained when the entire number was formed by taking the last two steps of the pixels reached within the data set consisting of 11-digit;

48683044 is the resulting value using the data obtained from the [117-120,235] address of the 7th character, and **21681179** is the resulting value using the [111-114,235] address of the 8th character.

Step 2: Data Decryption

In data decryption, the steps followed for hiding encrypted data, which are the steps of the data reading process, are followed in reverse order. It can also be said that data decryption is reversing the data encryption. If the previous sections are remembered, a 2-stage encryption method was applied to encrypt the data: first, a special value named control bit was obtained by the conversion of the character set values to two-digit codes and then in the second step, another three-digit number was attained using these two values. Next, as a second encryption step, the obtained number was encrypted again, by the RSA encryption algorithm, and therefore the data encryption was completed.

The decryption of encrypted data obtained by the previous data reading step is also a two-step process and the steps should be applied in reverse order, as it was mentioned before. First, the encrypted data should be decrypted using the RSA encryption algorithm, which is the second step of encryption. N and D keys, which were explained in detail in the encryption section, are needed for decryption using the RSA encryption algorithm. Hence, the keys, as well as the numbers, were hidden in the photograph in the data hiding step. Otherwise, decryption may result in an error.

When the keys located in the first 10 pixels are read, the N key is identified as 69225323, and the D key is identified as 21324619. Since the encrypted forms of the 7th and 8th characters were used as an example in the first step, seeing the conversion of the same numbers to their original values may be necessary to understand how the system works.

$M = C^{d} \pmod{N}$ formula is used for decryption encrypted data according to the RSA encryption algorithm. Therefore;

48683044 is the value of the 7th character obtained in the first step, and 21681179 is the value of the 8th character. If the key values and numbers identified through data reading, are to be put in places in the formula;

7th character would be: $48683044^{21324619} \pmod{69225323} = \mathbf{443}$

8th character would be: $21681179^{21324619} \pmod{69225323} = \mathbf{434}$

The encryption is decrypted because the encryption step, the RSA encryption algorithm, was followed in the reverse order. The next step is deciphering the encryption of the first encryption step. It is known the tens digit of the three-digit number created after the encryption, is used as the control bit. This step is testing the control bit. When the control bit value on the tens digit is subtracted from the number, the result is 43 for the 7th character. If 43 is transformed into the binary code, the result would be 00101011, and number **four** would be obtained when the 1-values are counted in this binary code. Hence, thus it is tested that the correct number is read and decryption from the photograph. In other words, with the help of the control bit, it is concluded that the number **43** is the correct number. Finally, number five is the equivalent of 43 in the Character Set (Table 1).

The coordinates of the points where all the numbers in the addressing line are hidden in the photograph should be identified first and then the encrypted data should be read. After this process, the two encryption steps should be processed in reverse, decrypted, and then the original numbers should be decrypted by testing via the control bit.

IV. CONCLUSION

Data encryption and data hiding techniques were used in this study that aims to develop new methods and algorithms in parallel with data and information security and test these through applications. In information security, two important scientific branches, cryptology, and steganography, are generally regarded as alternatives to each other. In the field of study, generally, the one suitable for the aim is selected and applied based on research. However, the main aim of this study is to use data hiding and data encryption techniques together. Data hiding and data encryption were used together in the application developed within the scope of this study using different methods and techniques. Therefore, the aim of enhancing security is achieved as a result of the studies conducted.

An application that detects identity fraud is developed. Today, even though technology facilitates everyday life, it can also cause damage. Fraud and forgery crimes are on the rise in parallel with the technology, according to the legal records; hence, causing many innocent people to suffer. The application developed checks whether the encrypted Turkish ID number that is hidden in the ID card photograph matches the actual ID number of the person by deciphering the encrypted numbers. The authentic encryption and hiding techniques were used for the application, and satisfactory results were achieved since the cryptography and steganography techniques were used together.

V. REFERENCES

Bahçetepe, H.: ” Modüler Çarpma Algoritmalarının İncelenmesi ve Kriptolojide Uygulamaları ”, *Yüksek Lisans Tezi*, İstanbul Üniversitesi Fen Bilimleri Enstitüsü, İstanbul, Türkiye, (2006).

Başkök, M.: ” AES Şifreleme Algoritmasının Modellenmesi ”, *Yüksek Lisans Tezi*, Gazi Üniversitesi Fen Bilimleri Enstitüsü, Ankara, Türkiye, (2007).

Buluş, H. N.: ” Temel Şifreleme Algoritmaları ve Kriptanalizlerin İncelenmesi ”, *Yüksek Lisans Tezi*, Trakya Üniversitesi Fen Bilimleri Enstitüsü, Edirne, Türkiye, (2006).

Çağlar, E., ‘’Açık Anahtarlı Kriptografi ve Ağ Güvenlik Uygulamaları’’, Yüksek Lisans Tezi, Çanakkale Onsekiz Mart Üniveristesi, Fen Bilimleri Enstitüsü,Çanakkale, (2004).

Denton, B. : " Evaluation of Cryptographic Construction Properties and Security Requirements of Modern Secure Hashing Algorithms", *Master Thesis*, The Department of Electrical and Computer Engineering of Alabama University, Huntsville, Alabama, USA (2011).

Kim Y.S., Kim Y.M., Choi J.Y., Baik D.K., "Information Hiding System StegoWaveK for Improving Capacity", International Symposium, ISPA 2003 Aizu-Wakamatsu, Japan, July 2-4, Proceedings, Springer Berlin/ Heidelberg, ISSN 0302-9743, vol. 2745/2003, 2003.

Rivest, R..; Shamir, A.; Adleman, L.: "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems,", *Communications of the ACM*, v. 21, n. 2, (Feb, 1978), pp. 120-126.

Sarıtaş, H.: " Dijital İmza Uygulamasının Eliptik Eğri Şifreleme Yöntemi Kullanılarak Gerçekleştirilmesi ", *Yüksek Lisans Tezi*, Marmara Üniversitesi Fen Bilimleri Enstitüsü, İstanbul, Türkiye, (2010).

Şahin, A.'' Görüntü Steganografi de Kullanılan Yeni Metodlar ve Bu Metodların Güvenilirlikleri''. Doktora Tezi., Trakya Üniversitesi, Bilgisayar Mühendisliği Anabilim Dalı, Edirne, 2007.

Tsunoo Y.; Tsujihara E.; Minematsu K.; Miyauchi H.:"Cryptanalysis of Block Ciphers Implemented on Computers with Cache", ISITA, (2002).

Tuncal, T.: " Bilgisayar Güvenliği Üzerine Bir Araştırma ve Şifreleme- Deşifreleme Üzerine Uygulama ", *Yüksek Lisans Tezi*, Maltepe Üniversitesi Fen Bilimleri Enstitüsü, İstanbul, Türkiye, (2008).

Ulutürk, A., '' Gelişmiş Şifreleme Standardı'', Yüksek lisans Tezi, Gazi Üniversitesi Fen Bilimleri Enstitüsü, Ankara, Türkiye (2010).