

بسم الله الرحمن الرحيم



:

:

- /  
- /

2007 / 1428

قال الله تعالى:

"يَرْفَعُ اللَّهُ الَّذِينَ آمَنُوا مِنْكُمْ وَالَّذِينَ أُوتُوا الْعِلْمَ دَرَجَاتٍ  
وَاللَّهُ بِمَا تَعْمَلُونَ خَبِيرٌ" (١١) "سورة المجادلة

صدق الله العظيم



)

(

58

74

%78

:

.

.

## Abstract

This study aimed to identify the acquisition of the use of digital signature in the management of information technology centers (ICT) in the Palestinian universities in the Gaza Strip represented in the (development of the infrastructure of the ICT, the attention of senior management of the ICT using digital signature, a legal and legislative requirements for the use of digital signature, adoption electronic correspondence and policies to protect electronic information), the study aims to clarify the importance of preserving the security of information and the use of technologically advanced equipment to protect privacy through electronic correspondence.

The researcher used descriptive analytical approach, where a study of all the staff of information technology centers in the Palestinian universities in the Gaza Strip, has been relying on results of the data collection by a questionnaire distributed 75; respondents were 58 which mean 77%.

The study concluded that the digital signature technology is not used in the information technology centers at Palestinian universities in the Gaza Strip, and security hardware and software used in the information technology centers need to be updated to be able to protect electronic information efficiently and staff information technology centers have limitations of the concept of electronic signature.

The study came out several recommendations, including: the administrations of the Palestinian universities need to adopt electronic correspondence using official electronic signature, updating security equipment and the development of information security software currently used in the information technology centers, and special annual budget allocation to information security and training the staff of information technology centers in the area of information security, the formulation and development of policies for the security of information and using electronic signature constantly in the information technology centers.

## الإهداء

إلى روح والدي رحمه الله

إلى والدتي العزيزة التي حثتني دائماً على الدراسة وإلى زوجتي الغالية التي شجعتني كثيراً على  
الدراسة وقدمت لي كل عون

إلى أرواح الشهداء

إلى من يجب النجاح والتوفيق ويرجوه لي

إلى طلبة العلم في كل مكان

أهدي جهدي المتواضع لهم جميعاً

## الشكر و التقدير

أتوجه بالشكر لله سبحانه وتعالى الذي ارتضى أن يكون شكر الناس شكراً له . . . كما أتوجه بالشكر والتقدير لأعز وأغلى الناس والدتي العزيزة لدعائها لي في إنهاء هذه الدراسة، كما أتوجه بالشكر والتقدير لزوجتي الغالية لمساندتها لي لإنجاح وإتمام هذه الدراسة جزاهما الله عني خير الجزاء . .

كذلك أتقدم بجزيل الشكر وعظيم الامتنان إلى أساتذتي: الدكتور عصام البحيصي و الدكتور توفيق برهوم على نصائحهما السديدة و تفضلهما بالإشراف على هذه الدراسة جزاهما الله كل خير.

كما أشكر الاستاذ الدكتور/ يوسف عاشور، و الدكتور/ رشدي وادي لتفضلهما بقبول مناقشة هذه الرسالة .

وكذلك أتقدم بشكري للجامعة الإسلامية التي اعمل بها وتمنحني بهذه الدراسة الدرجة العلمية الثانية إن شاء الله . . . .

:	
<b>2</b>	
<b>3</b>	
<b>3</b>	
<b>4</b>	
<b>4</b>	
<b>5</b>	
<b>5</b>	
<b>5</b>	
<b>6</b>	
<b>6</b>	
<b>7</b>	
:	
<b>18</b>	<b>1.1</b>
<b>18</b>	1.1.1
<b>18</b>	1.1.2
<b>19</b>	1.1.3

20		1.1.4
24		1.1.5
26		1.1.6
29		1.2
30		1.2.1
31		1.2.2
31		1.2.3
32		1.2.4
32		<b>1.3</b>
33		1.3.1
34		1.3.2
35		1.3.3
35		1.3.4
36		<b>1.4</b>
37		<b>1.5.1</b>
38		<b>1.5.2</b>
39		<b>1.6</b>
39		<b>1.5.1</b>
40		<b>1.5.2</b>
44		<b>1.5.3</b>
45		<b>1.6</b>
46		1.6.1
47		1.6.2
		:
51		
52	-	
59	-	
61		-
63	-	



:	
68	
68	
68	
69	
70	
71	
74	
79	
82	
:	
90	
93	
:	
110	
111	
112	
114	
:	
118	
120	
122	
:	
124	
125	

الصفحة	
26	
27	
43	
56	
63	
70	
71	
72	
72	
73	
73	
74	
75	
76	
77	
77	
78	
79	
80	" "
80	
84	
84	
85	
85	
86	

86	
87	
88	
88	
90	
92	
94	
96	
98	
100	
102	" "
104	
105	" "
107	

	.
108	
112	
113	
114	
114	
115	
115	

#		
1	<b>authentication</b>	:
2	<b>authorization</b>	:
3	<b>password /username</b>	: ...
4	<b>VPN</b>	: (Virtual private networks)
5	<b>Biometrics Signature</b>	هو توقيع يتم باستخدام الخواص الذاتية للإنسان مثل بصمة الأصبع أو قزحية العين.
6	<b>Digital Certificates- DC</b>	:
7	<b>Certificate Authorities -CA</b>	:
8	<b>Smart Cards</b>	:
	<b>SIM Cards</b>	:
9	<b>Trusted third Party</b>	جهة ثالثة محايدة لتصديق الشهادات الرقمية

10	<b>Asymmetric Cryptography</b>	التشفير اللامتماثل ، هو عبارة عن تشفير زوج من المفاتيح الغير متماثلين تستخدم لتشفير ولفك تشفير الرسالة لتكون آمنة.
11	<b>Symmetric Cryptography</b>	التشفير المتماثل : هو التشفير الذي يستخدم نفس المفتاح من أجل التشفير و فك التشفير
12	<b>Web Server</b>	ويب سيرفر أو خادم الويب وهو جهاز مخصص لوضع وتشغيل موقع الإنترنت الخاص بالمؤسسة .
13	<b>Keys</b>	:
14	<b>Public Key</b>	:
15	<b>Private key</b>	:
16	<b>Cryptosystems</b>	:
17	<b>Encryption</b>	:
18	<b>Public-key cryptography</b>	digital : " IDs "
19	<b>Directory Service</b>	:
20	<b>Hacker</b>	:

#		
1	<b>PIN</b>	Personal Identification Number
2	<b>PKI</b>	Public Key Infrastructure
3	<b>HTTP</b>	hypertext transfer protocol
4	<b>B2B</b>	Business to Business
5	<b>CRL</b>	Certificates Rejected list
6	<b>PDA</b>	Personal digital assistant
7	<b>CA</b>	Certificate Authority
8	<b>E-Sign Act</b>	Electronic Signatures in Global and National Commerce

148	
153	

# الفصل الأول

## الإطار العام للدراسة

:

:

:

:

:

:

:

:

:

:

:

... (2003).

(2003).

(Authentication) :

(Integrity)

(Confidentiality or Privacy)

(teletrust.de) (non- repudation )

:

.(1999) Hassler



:

.

"

"

:

.1

.2

.3

.4

.5

.6

(  $\alpha \leq 0.05$  )

.1

(  $\alpha \leq 0.05$  )

.2

(  $\alpha \leq 0.05$  )

.3

(  $\alpha \leq 0.05$  )

.4

(  $\alpha \leq 0.05$  )

.5

(Hackers)

( )

■

■

:

-1

-2

-3

-4

:

:

-

-

-

-

-

:

.

:

:

)

:

.(

.(

):

■

■

■

■

■

■

:

:

:

:

:

.1 (1998) Wilson

( ... )  
▪  
▪  
▪  
(Integrity) :  
(Privacy) (Authentication)

.2 (1999) Borasky

.3 (2001) Godwin

-

-

-

-

**(2001) David .4**

**(2002) Kuechler .5**

"

:

"

■

■

■

■

**(2004) Gabor .6**

Electronic Identifier

Digital Signature

(computer algorithm)

.It is unique to the person using it

**(2004) Edward .7**

(2004) Rodney .8

(2006) Thomas .9

(2006) Shalhoub .10



•

•

**(2006) Kwo-Shing Hong** .11

**(2007) Christopher** .12

:

...

(Electronic Signatures in Global  
and National Commerce – ESIGN Act. , E-Sign Act.)

E-Sign Act.

:

◆

◆



:

:

■

■

■

:

(2004) Rodney

(2006) Thomas

(2006) shalhoub

:

(2002) Kuechler

(2004) Gabor Moroc

)

(

:

(1999) Borasky

(2004) Edward

(2007) Christopher

. Stephen(1998)

(2007) Whitaker

(2007) Christopher

:

Certificate Authority -CA

CA

PIN/Password

(

)

PIN/Password

:

❖

:

:

.1

.2

.3

.4

Certificate Authority -CA

CA

(

)

.(

)

:



# الفصل الثاني

## الإطار النظري

:  
:  
:  
:  
:  
:  
:

(Digital

Signature)

.(2006)

(Authentication) :

(non-

(Integrity)

(Confidentiality or Privacy)

(teletrust.de) repudiation )

Authentication :

non-

Data integrity

Data Authentication

.(Hassler, 1999) Repudiation

## 1.1 Digital Signature

: 1.1.1

(Marking a Document)

(Private key) (Public Key)  
(2001) Minihan

: 1.1.2

(2000) White

(Authenticity of Message)

(Unique digital representation) ( )

- (Hash)

(Signing Function)

( ) Private Key

Public Key hash

.(digital fingerprint)

:

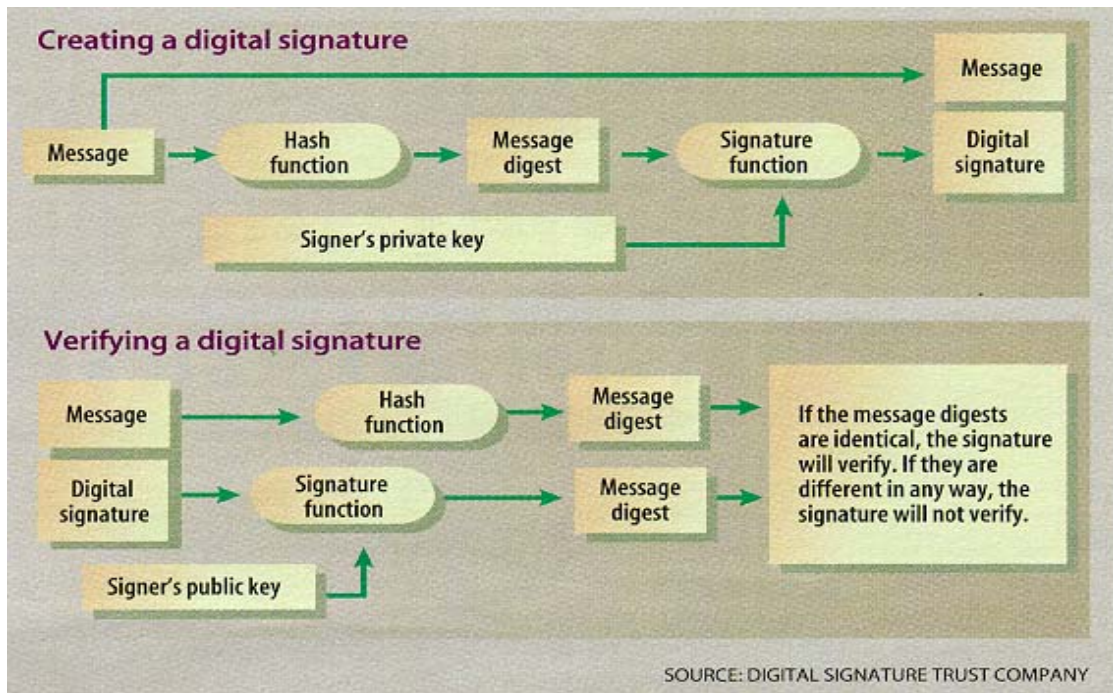
hash



hash

(1.1)

.(2000) Jones



.(2000) Jones

(2.1)

1.1.3

(2004) CGI

:

:(Message Signature) :\_\_

:Message Digest Evaluation (a

(Data " " "

-integrity)

**:Digest Signature (b**

(Hashing Algorithm) Hashing

**:(Message Encryption) :\_\_**

**:/ (a**

**Creation of a One time symmetric encryption/decryption key**

**:Message Encryption (b**

**:symmetric-Key(SymK) encryption (c**

**( )**

**1.1.4**

**:(2004) CGI**

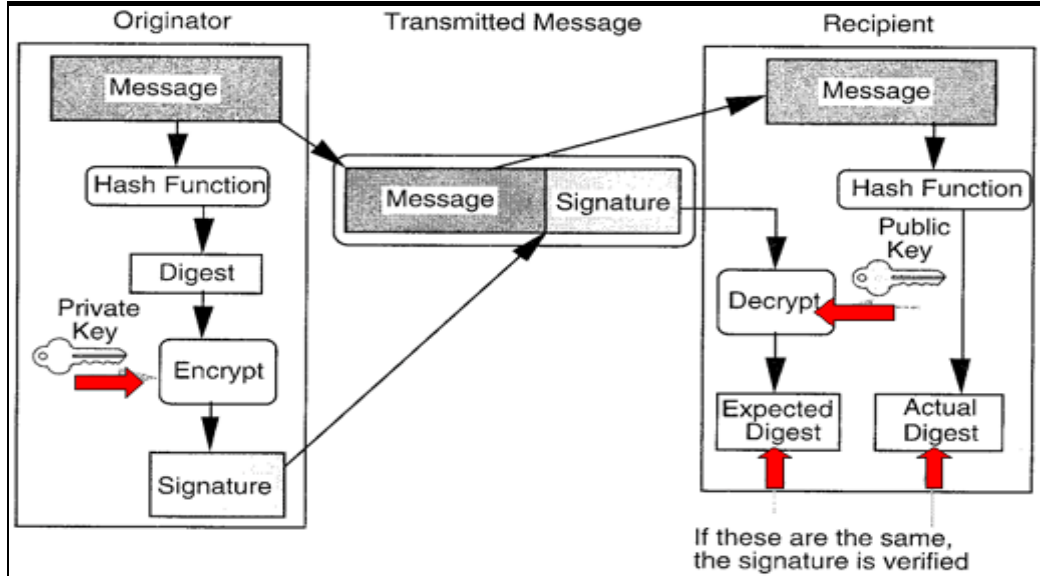
**:Message decryption :\_\_**

**:symmetric-key decryption (a**

**:Message Decryption**

(b)

.( )



(Arizona State University)

(2.2)

**:signature verification**

:\_

digest

**:Message digest decryption (a**

hashing

**:Digest evaluation (b**

digest

hashing

digest

Hash

:

**: Digests**

(c)

**Message digest decryption •**

**Digest evaluation •**

- :
- .1
  - .2
  - .3
- (1.2)

( Arizona State University )

:

hash :a message digest :\_\_

(Mathematical Formula)

(characters)

digest

digest :\_\_

(Public key Cryptography)

)

.(

"Certification Authority-CA " :\_\_

)

(

:

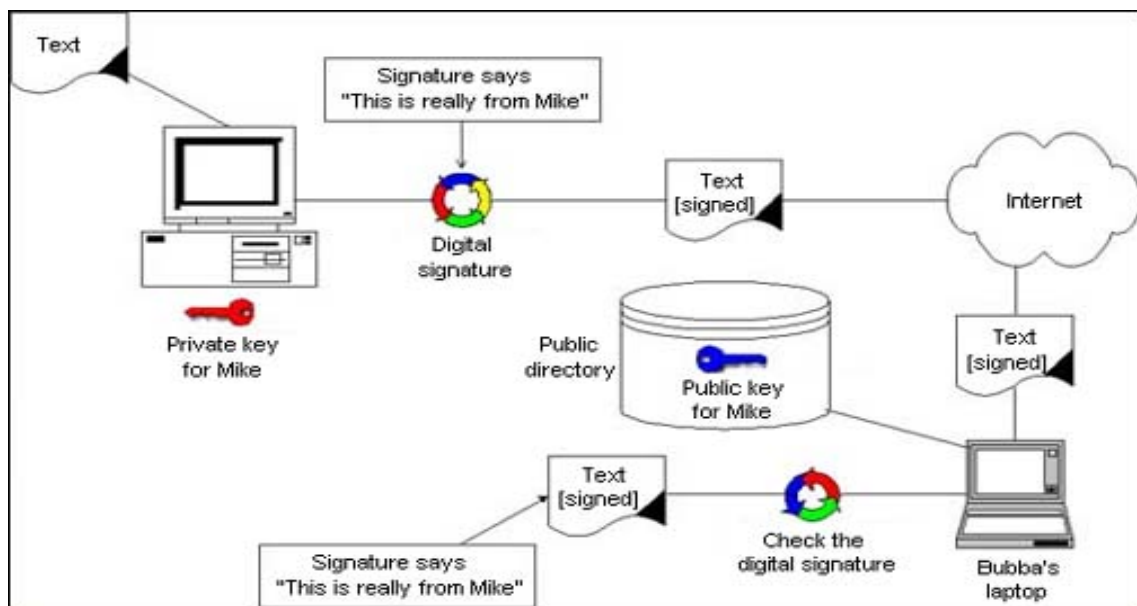
.1  
 (Message digest) (Hash )  
 Message digest

.2  
 " (1)  
 Message digest  
 Digest Message Hash  
 Digest Message

.3  
**Digest Message**

( )  
 IBM

(2.3)



(IBM.com)

(2.3)

## 1.1.5

(North Carolina)

:

**:Integrity ( ) .1**

(2000) Radcliff

(2003) Horton

**:Privacy .2**

(2003) Horton

" /

(teletrust.de)

"

(2003)

**:Non-repudiation .3**

(2000) Radcliff

(returns a proof of receipt-

)

)

-

.(

**: Authentication** .4

:  
: (teletrust.de)  
" (wikipedia.org)  
"

**: Date-Time stamping** .5

**:Speed & Accuracy** .6

(2.1 )

(2.1)

#			
1	Privacy	( ) •	( ) •
2	Authentication	•	•
		•	( )
3	Data Integrity	•	•
4	Non-Repudiation	•	•
		•	•
5	Date-Time stamping	•	•
6	Speed & Accuracy	•	•

(2.1)

(entrust.com)

.(2.2)

**1.1.6**

(attached)

(Codes)

"

.1

.(2000) Radcliff



(Entrust.com)

(2.2)

#	Property	Paper Signatures	Digital Signatures
1	Can be applied to electronic documents and transactions	No	Yes
2	Signature verification can be automated	No	Yes
3	Signature automatically detects alterations to the document	No	Yes
4	Can be used to signify a commitment to a contract or document	Yes	Yes
5	Can be augmented by use of a witness to the signature process	Yes	Yes
6	Recognized by legislation	Yes	Yes

" .2

.(entrust.com)

" (teletrust.de) .3

"

(Asymmetric Cryptography) .4

written

.(Wikipedia.org) Form

" (1999) Wilson .5

"

.6

(string of characters)

.(2004) Edward

" :  
 ."  
 "Electronic Signature "  
 " Digital Signature "  
 Electronic Signature  
 Digital Signature (2002)  
 : electronic Signature  
 Digital ) (Encryption) (Mathematical Algorithm)  
 (1999) Borasky (Certificate  
 .(2000) Radcliff

Digital Signature  
 Digital Electronic Signature  
 (1999) Borasky Signature  
 Electronic Signature  
 : (2002)

- 
- Personal Identification Number–PIN
- 
- (Biometrics Signature)
- -Digital Signature

(Stream of Digits)  
Digital Signature

(2002)

## 1.2 Public Key Infrastructure-PKI

- PKI-

" " " "

PKI

PKI

(2.4)

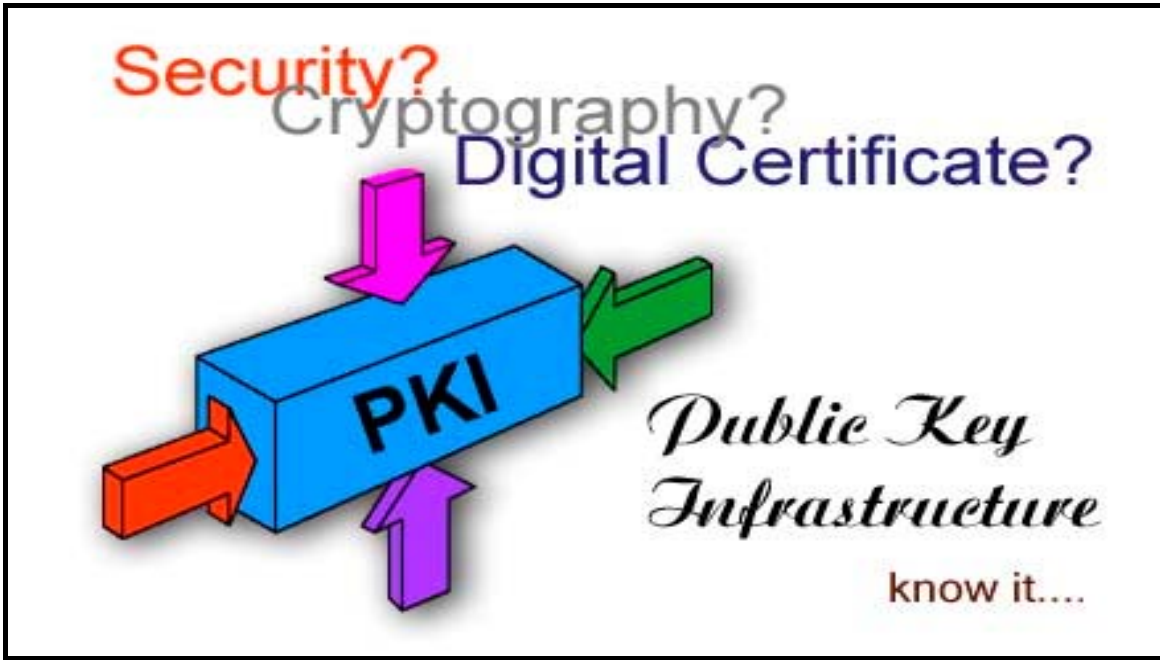
PKI – (Michigan Technological University)

: (2002) Kuechler

: **security** .1

: **Attribution** .2

: **Non-repudiation** .3



(Michigan Technological University)

(2.4)

:		
	(teletrust.de)	
:	(manage digital Certificates)	PKI
	. . .	•
		•
		•
	(Validity Period of the	
		.Certificate)
	:	<b>1.2.1</b>
:	(teletrust.de)	
	. (HTTP)	.1
	.Secure e-Mail	.2
	Secure payment transactions	.3
	.Signing electronic documents	.4
	Electronic signature equivalent to	.5
	handwritten signature	
	Banking applications	.6

Secure data transfer .7  
Virtual private networks - VPN .8

: 1.2.2

:

: (Password) •

: Private key encryption •

Yakal

.(2000)

1.2.3

: (teletrust.de)

: Encrypted storage on magnetic Media .1  
(Encrypted Form)

High level security .2

(Smart Cards)

:modules without read access  
(SIM Cards)

.(Howstaffworks.com)

## 1.2.4

:(us-cert.gov)

( )

.1

.2

.3

.4

## 1.3

### Digital Certificates(DC)

(Certificate  
(Trusted third

Authority-CA)  
Party)

PKI

.(Microsoft)

)

(

(2002)

Public key

Private Key

.(Cren.net)

( )

(Invalid signature)

(Cren.net)

PKI

.(2002) Department of Information Resources

: **1.3.1**

: ( Netscape.com)

.1

.2

: (cren.net)

WebServer .3

.4

( Netscape.com)

:

(Threat of  
Interception)

**: Server Certificates (a)**

(Private  
Account)  
(Business to Business - B2B)

**: Personal Certificates (b)**

- 1.3.2**
- (1999) Hassler
- :
- :Valid ( ) .1**
- :Expired .2**
- : Revoked .3**
- (Certificates Rejected list –CRL)
- : Archived .4**
- :Suspended .5**
- . CRL



: **Superseded** .6

...

Update

**1.3.3**

)

([cren.net](http://cren.net))

(

Texas-Houston Columbia

Dartmouth Minnesota

PKI Technology

(Web Server Certificates)

WebServer

: **1.3.4**

(Server Certificates)

(1998) Meckbach

(2004)

:

		:	.1
		:	.2
		:	.3
		- (Entrust.com)	
2.5		-	PKI-
		1.5	
40.000	"Scotiabank"		1998
	.1998		
		:	.4

## 1.5 Certification Authorities (CA)

(definethat.com)

Private

( )  
(2004)

Public key key

(Microsoft)

(Teletrust.de)

(Cren.net)

Identity

(2002) Kuechler

(Servers)

...

( )

:

1.5.1

:

Microsoft

(Applicant)

**Key Generation** ( ) .1

CA

:Matching of Policy Information	.2
.....	
:sending of Public keys & Information	.3
.	
:Verification of Information	.4
.	
: Certification Creation	.5
.	
:Sending/Posting of Certificate /	.6
.	
The Certificate is Loaded onto an (install) : individual's Computer	.7
.	
:	<b>1.5.2</b>

CRL

CRL

DCs

CAs

1.6  
**Information Security Polices – ISP**

(2006) Kwo-shing Hong  
(Guideline of Information Security)

(2004) Kostas

**1.5.1**

(teletrust.de)

Unique Private Key  
 Securely Managed ( )  
 PDA Laptop :  
 (teletrust.de) ,(cren.net)  
 Smart Cards . SIM Cards  
**1.5.2**  
 (2002) Michigan  
 :  
 (Privacy) .1  
 .  
 (Licensing of .2  
 programs)  
 .  
 .3  
 ) Users Accounts  
 .(  
 (electronic exchange .4  
 systems)

(Integrity of the systems) .5

(2002) Michigan  
: (2001) New York At Stony Brook

:(Privacy) .1

: (Copyright) .2

: (Software) .3

: (Harassment, Defamation)	.4
: (User Accounts)	.5
: (Permitting unauthorized access)	.6
(          )	
: (Termination of access)	.7
: (Circumventing Security)	.8
(system's security)	
: (Breaching Security)	.9
:	
(Creating or knowingly propagating viruses)	•
(Hacking)	•
>Password cracking)	•
Unauthorized viewing of )	•
(other's files	



" "

:

(2.3) (2001) New York At Stony Brook

(2.3)

#		New York	Michigan
1	<b>Privacy</b>	√	√
2	<b>Copyright</b>	√	
3	<b>Software(Licensing of programs)</b>	√	√
4	<b>electronic exchange systems</b>		√
5	<b>Integrity of the systems</b>		√
6	<b>Harassment, Defamation</b>	√	
7	<b>User Accounts</b>	√	√
8	<b>Permitting unauthorized access</b>	√	
9	<b>Termination of access</b>	√	
10	<b>Circumventing Security</b>	√	
11	<b>Breaching Security</b>	√	

1.5.3

:

:

(Morality)

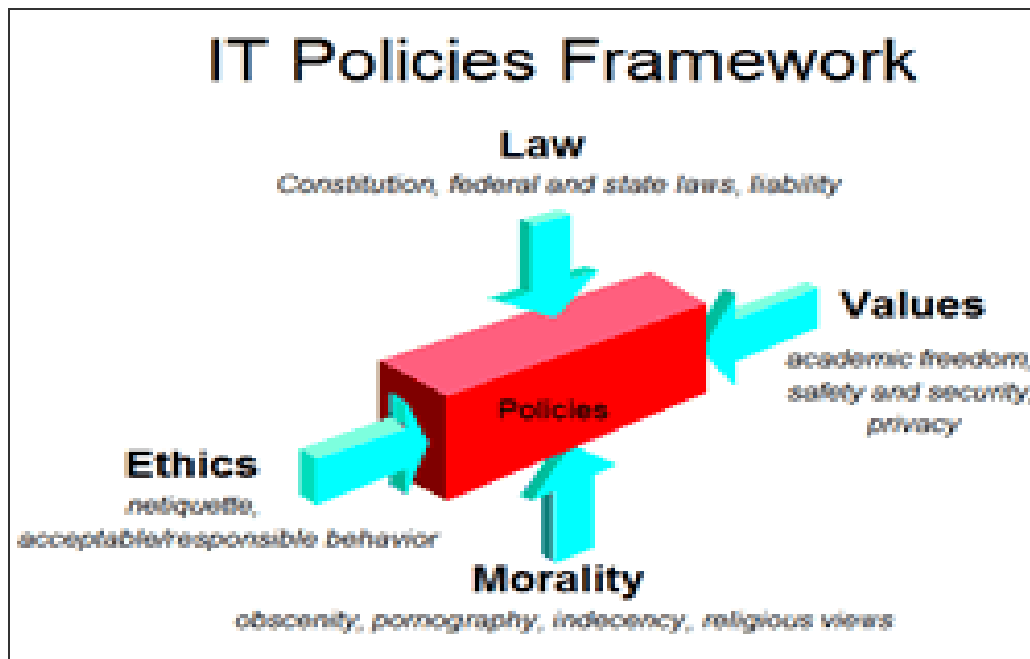
(Ethics)

(Values)

(LAW)

:

(2004) Rodney



(2004) Rodney

(1.4)

## 1.6

### The electronic signature legalism

- -

( )

:(2006)

1.6.1

:

.1 :

.2 :

(2005)

CA

.3 :

.4 :

(2005)

( )

## 1.6.2

(2005)

:

"

"

( )

"

"

•

•

:

"

"

(uncitral.org)

"

:

.1

.2

"

" " " "

:

.1

.2

(2004)

Private

Invalid

Key

.Signature

.(2004) Rodney

## الفصل الثالث

مراكز تكنولوجيا المعلومات في الجامعات الفلسطينية في قطاع غزة

أولاً: إدارة شؤون تكنولوجيا المعلومات في الجامعة الإسلامية

ثانياً: وحدة تكنولوجيا المعلومات – جامعة الأزهر

ثالثاً: مركز الحاسوب - جامعة الأقصى

رابعاً: مركز تكنولوجيا المعلومات والاتصالات – جامعة القدس المفتوحة



مراكز تكنولوجيا المعلومات في الجامعات الفلسطينية في قطاع غزة  
IT Centers at Palestinian Universities in Gaza Strip

:

" (Information technology centers)

" Computer Centers-

Update

.....

"

"

"

"

"

"

:

- .1
- .2
- .3
- .4 .( )
- .5

:

"

"

-

-

:

-

-

•

•

"

"

)

.( – 2005

:

.1

.2

.3

.4

.5

.6

.7

:

: ( – 2005 )

: .1

videoconference

: .2

:

:

: .3

Intel

-

-

-

.4

:

:

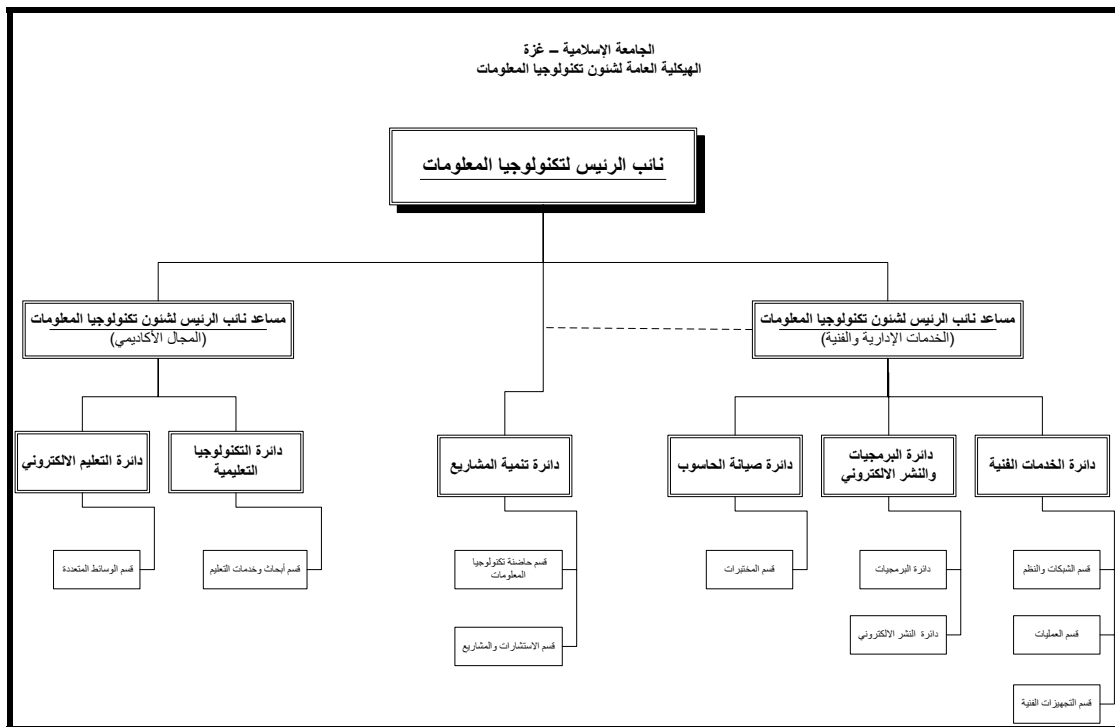
- 
- 
- 
- 

(3-1) :

-

-

- 
- 



( )

-

(3-1)

-  
( )

:

: .1

Restore Backup

(Availability)

1Gbps

Fiber Optics

10/100Mbps

FTP

.( - 2005 )

(Switch/Hub)

224 (Server)

17

10/100Mbps

(Antivirus)

(Firewall)

)

1.5Mbps

(2005

:

(3-1)

3650	
1784	
900	
224	
17	Servers
2700	

.(User Account)

.2

-:

Portal

- 2005

)

.(

.3

UPS

:

(

)

...

.(

- 2005

)

:

•

Open LAB

.(

)

.4

تتكون دائرة تنمية المشاريع من :

( : • )

: •

.5

:



-

.

)

.( - 2005

-

:

" 2001

"

.

.( )

500

:

( )

:

: .1

: .2

:

.

.3  
:  
○  
○  
:  
)  
:  
(  
.1  
:  
)  
Scanner UPS

.2

:

.3

( )

” ”

.

- :

2002

.( )

( ) ( )

.

:

: ( )

: .1

::

.

: .2

( )

)

: (2006-2005

.1

:

.2

Servers

.3

28 (Server) 14

10/100Mbps

(Switch/Hub)

( - )

:(5-3)

(3-5)

350	
398	
859	
28	
14	Servers
277	

. - :

20

)

(

(Information & )

Communications Technology Center - ITTC)

**:ICTC**

( )

.1

.2

(Self Service Provider)

.3

.4

( )

.5

.6

.7

.8

## ITTC

•

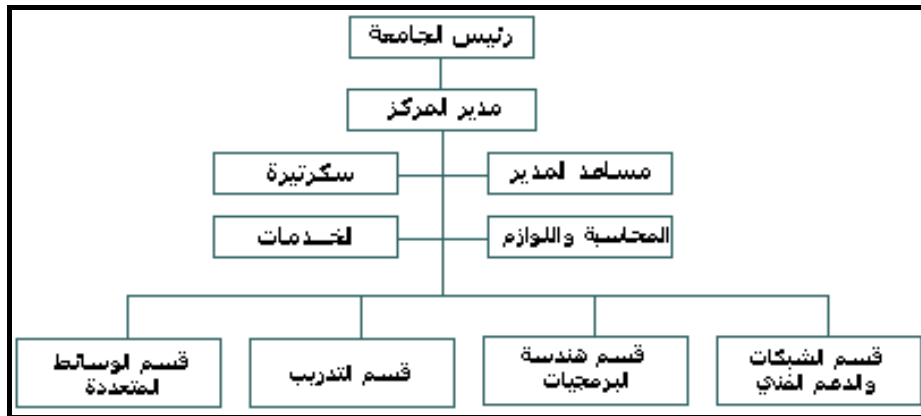
:

.1

.2

.3

.4



)

(3-5 )

(

## ITTC

:

...

.(Security Administrator)



:

:

:

:

:

:

:

:

: 4.1

(2006

(Questionnaire)

(Statistical Package for Social Science) SPSS

: 4.2

:

:

•

•

- 
- 

:

4.3

:

" "

:

:

1	2	3	4	5	

:

:

:

14

(32)

:

:

7

:

:

9

:

5

:

5

:

6

:

(74)

:

-  
-  
-  
-

(4.1)

		#
35		1
13		2
12		3
14	-	4
<b>74</b>		

:

(65)

(74)

(7)

(%78)

(58)

:

(4.2)

46.55	27	34	35	
18.97	11	11	13	
20.69	12	11	12	
13.79	8	9	14	
<b>100.00</b>	<b>58</b>	<b>65</b>	<b>74</b>	

%46.55

(4.2)

%13.79

%20.69

%18.97

(4.3)

24.14	14	
67.24	39	
8.62	5	
<b>100.00</b>	<b>58</b>	

%8.62 (4.3)

" "

(4.4)

20.69	12	
60.34	35	
18.97	11	
<b>100.00</b>	<b>58</b>	

%60.34 (4.4)

" "

(4.5)

27.69	11	5-
55.38	36	10-6
16.92	11	10
<b>100.00</b>	<b>58</b>	

55.38% (4.5)  
( 10 -6)

(4.6)

1.72	1	
50.00	29	5-
39.66	23	6-10
8.62	5	10
<b>100.00</b>	<b>58</b>	

5 - ) 50.00% (4.6)

10

(%8.62)

.(4.5)

(4.7)

5.17	3	
3.45	2	
12.07	7	
5.17	3	
55.17	32	
3.45	2	
5.17	3	
10.34	6	
<b>100.00</b>	<b>58</b>	

" %5.17 (4.7)  
%12.07 %3.45 "  
%5.17  
%55.17  
%5.17 %3.45  
%10.34  
" " " " " "

(20)

:

:

.1

.2

**:Validity Coefficient**

:

:



: -1

(8)

(6)  
 (3) (6-4)  
 (10) (56)  
 (46)

:( ) -2

(20)  
SPSS

:

(4.8)

0.05	0.510	
0.01	0.753	
0.05	0.547	
0.05	0.547	
0.01	0.856	
0.01	0.733	
0.01	0.820	

0.444 = (0.05)

(2-20)

r

0.561 = (0.01)

(2-20)

r

) (4.8)

(  
( 0.856 0.510 )

(0.01 0.05)

(4.9)

0.01	0.736	
0.05	0.486	
0.01	0.761	
0.01	0.790	
0.01	0.719	
0.05	0.504	
0.05	0.446	
0.01	0.614	
0.01	0.759	

0.444 = (0.05)

(2-20)

r

0.561 = (0.01)

(2-20)

r

(4.9)

0.446)

(0.01 0.05)

( 0.790

(4.10)

0.01	0.731	
0.01	0.691	
0.01	0.679	
0.01	0.808	
0.01	0.778	

0.444 = (0.05) (2-20) r

0.561 = (0.01) (2-20) r

(4.10)

0.679)

(0.01) (0.808

(4.11)

0.01	0.571	
0.01	0.827	
0.01	0.685	
0.01	0.587	
0.05	0.539	

0.444 = (0.05) (2-20) r

0.561 = (0.01) (2-20) r

(4.11)

0.539)

(0.01 0.05)

(0.827

(4.12)

0.01	0.684	.1
0.01	0.826	.2
0.01	0.782	.3
0.01	0.779	.4
0.01	0.700	.5
0.05	0.453	.6

0.444 = (0.05)

(2-20)

r

0.561 = (0.01)

(2-20)

r

(4.12)

0.453)

(0.01 0.05)

(0.826

(32)

(4.13)

**0.830	
**0.793	
*0.475	
**0.603	
**0.750	

0.444 = (0.05)

(2-20)

r \*

0.561 = (0.01)

(2-20)

r \*\*

(4.13)

: -1

:

: -

(4.14)

" "

" "			
0.858	0.854	7	
0.566	0.501	9	
0.748	0.715	5	
0.743	0.705	5	
0.674	0.508	6	
<b>0.584</b>	<b>0.412</b>	<b>32</b>	

(0.858 – 0.566)

(4.14)

(0.584)

(4.14)

(4.15)

0.807	7	
0.742	9	
0.784	5	
0.626	5	
0.730	6	
<b>0.873</b>	<b>32</b>	

(4.15)

(0.873)

(0.626)

:

:

-1

(20)

:

.1

.2

.3

.4

.5

.6

:

-2

.(58)

: -3

. (25)

:

:

-1

-2

-3

-4

-5

.SPSS





:

:

:

:

(5.1)

27.59	16	
72.41	42	
<b>100</b>	<b>58</b>	

%27.59 (5.1)

." "

(5.2)

6.90	4	
13.79	8	-
5.17	3	5-2
1.72	1	5
<b>27.59</b>	<b>16</b>	

(5.2)  
( - )

(4.6)

-3

(5.3)

74.14	43	
25.86	15	
<b>100.00</b>	<b>58</b>	

%74.14 (5.3)

" "

(5.1)

-4

(5.4)

48.28	28	
51.72	30	
<b>100.00</b>	<b>58</b>	

%51.72 (5.4)

-5

(5.5)

32.76	19	
29.31	17	
37.93	22	
<b>100.00</b>	<b>58</b>	

%32.76 (5.5)

(5.6)

(2) (1)

-6

(5.6)

				#
1	17.24	10		1
2	13.79	8		2
3	12.07	7		3
4	10.34	6		4
5	6.90	4		5

(5.5)

.....  
.....  
.....  
.....  
.....

-7

(

(5.7)

48.6	1.827	2.43	35	0	2	5	16	
			60.34	0	3.45	8.62	27.59	%

(5.7)

%60.34

.....

)

(

(7)

%50

%48.6

(5.8)

27.59	16	
72.41	42	
<b>100.00</b>	<b>58</b>	

%72.41 (5.8)

" " (7)

(5.9)

67.66	0.772	2.03	16	24	18	
			27.59	41.38	31.03	%

%31.03 (5.9)

%41.38

%27.59

%67.66

(oracle)

(5.4)

:

" .1

."

(5.10) t

"%60"

%71.72 % 61.38

.0.05

(5.10)

(58 = )

	مستوى الدلالة	قيمة t				
5	0.040	2.106	65.86	3.29		1
6	0.096	1.691	64.48	3.22		2
1	0.000	4.147	71.72	3.59		3
2	0.000	4.271	70.34	3.52		4
3	0.001	3.512	69.66	3.48		5
7	0.637	0.475	61.38	3.07		6
4	0.001	3.630	69.66	3.48		7
	<b>0.000</b>	<b>4.592</b>	<b>67.59</b>	<b>23.66</b>		

2.00 = (0.05)

(57)

t

2.66 = (0.01)

(57)

t

(3)

(5.10)

"

"

%71.72



	"		(4)	
	"		%70.34	
	"		(5)	
	"		%69.66	"
	"		(7)	
	"		%69.66	"
	(6)			
65.86	"	"	(1)	
		(6)		
"			(2)	
"			%64.48	

(15)  
 (8)  
 - (6) (5.10) " -  
 " -  
 %61.38

**23.66**

**0.05**

**0.000**

**% 67.59**

◆

: (5.11)

(5.11)

**0.821	

0.250 = (0.05)

(56)

\*

0.325 = (0.01)

(56)

\*\*

(5.11)

0.01

(  $\alpha \leq 0.05$  )

” ”

.2

”

(5.12)

t

”%60”

0.05

%74.83

% 53.10

:

(5.12)

(5.12)

(58 = )

	مستوى الدلالة	قيمة t				
7	0.141	1.493	64.14	3.21		8
8	0.695	0.394	61.03	3.05		9
2	0.000	4.018	71.03	3.55		10
4	0.005	2.899	67.93	3.40		11
5	0.110	1.622	65.52	3.28		12
1	0.000	5.203	74.83	3.74		13
9	0.006	- 2.829	53.10	2.66		14
6	0.118	1.585	64.48	3.22		15
3	0.000	3.762	70.34	3.52		16
	<b>0.003</b>	<b>3.141</b>	<b>65.82</b>	<b>29.62</b>		

2.00 = (0.05)

(57)

t

2.66 = (0.01)

(57)

t

(13)

(5.12)

"

"

%74.83

.

"

"

(10)

%71.03

	(16)	(5.12)	
"			"
		%70.34	
	"	(11)	
%67.93	"		"
		(10)	
	"	(12)	
"			
		%65.52	
		(7)	(6)
	(1)		
	"	(15)	
%64.48	"		"

	"		(8)	
	"			%64.14
		.(15)		
	(9)		(12)	"
		%61.03		"
			.(15) (8)	
"	-	-	(14)	
		%53.10		"
Digital				Certificates
(24)				
29.62			(5.12)	
	0.05	0.003		% 65.82
				◆
(5.13)				
				:

(5.13)

**0.885	

0.250 = (0.05) (56) \*

0.325 = (0.01) (56) \*\*

(5.13)

0.01

(2006) Thomas

( $\alpha \leq 0.05$ )

" .3

"

(5.14)

t

"%60"

%78.62 % 66.21

: (5.14) 0.05

(5.14)

(58 = )

5	0.069	1.856	66.21	3.31		17
1	0.000	7.304	78.62	3.93		18
4	0.004	2.972	68.28	3.41		19
2	0.000	7.046	78.62	3.93		20
3	0.003	3.056	69.66	3.48		21
	<b>0.000</b>	<b>5.870</b>	<b>72.28</b>	<b>18.07</b>		

2.00 = (0.05)

(57)

t

2.66 = (0.01)

(57)

t

(18)

(5.14)

"

% 78.62

"

- -

(Rodney, 2004)

"

(20)

%78.62

"



	"		(21)	
	%69.66		"	
	(19)		(5.14)	"
	%68.28	"		
	"		(17)	
	"			%66.21
(20)	(19)	(18)		
	"			
% 72.28	18.07	"	0.05	0.000

(20)

(1999) Borasky

(2004) Edward

(2004) Rodney

(18) (19).



: (5.15)

(5.15)

**0.625	

0.250 = (0.05) (56) \*

0.325 = (0.01) (56) \*\*

(5.15)

0.01

(2004) Rodney

(19) (17)

(  $\alpha \leq 0.05$  )

(  $\alpha \leq 0.05$  )

:

"

"

.4

"

(5.16)

t

"%60"

0.05

%76.21

% 52.41

:

(5.16)

(5.16)

(58 = )

	مستوى الدلالة	قيمة t				
4	0.002	3.236	70.34	3.52		22
1	0.000	5.457	76.21	3.81		23
5	0.006	- 2.872	52.41	2.62		24
3	0.000	3.970	71.72	3.59		25
2	0.000	4.169	72.76	3.64		26
	<b>0.000</b>	<b>4.065</b>	<b>68.69</b>	<b>17.17</b>		

2.00 = (0.05)

(57)

t

2.66 = (0.01)

(57)

t

(23)

(5.16)

%76.21

(26)

%72.76

(18)

(25)

				%71.72
.	(19)	(18)		
	(22)		(5.16)	"
			%70.34	"
				.(28)
"	-	-	(24)	
"				%52.41
			(5.16)	
	0.05	0.000	%68.69	17.17

(2007) Christopher

(1998) Wilson



: (5.17)

(5.17)

**0.763	

0.250 = (0.05)

(56)

\*

0.325 = (0.01)

(56)

\*\*

(5.17)

0.01

)

( ....

.(2001) David

(  $\alpha \leq 0.05$  )

" :

"

(  $\alpha \leq 0.05$  )

" " .5  
 (5.18) t  
 "%60"  
 0.05 %76.90 % 53.45  
 : (5.18)

(5.18)  
 " "  
 (58 = )

1	0.000	6.897	76.90	3.84		27
5	0.074	-1.821	55.52	2.78		28
6	0.007	-2.815	53.45	2.67		29
2	0.000	7.254	75.86	3.79		30
3	0.892	-0.136	59.66	2.98		31
4	0.898	-0.129	59.66	2.98		32
	<b>0.023</b>	<b>2.338</b>	<b>63.51</b>	<b>19.05</b>		

2.00 = (0.05) (57) t  
 2.66 = (0.01) (57) t

(27) (5.18)

" " %76.90

	"			(30)	
	%75.86	"			
				(31)	
	%59.66	"			
				(32)	
	%59.66	"			
				(28)	
	"			(5.18)	"
					%55.52
	(38)				
	"	-	-	(29)	
	"				%53.45
0.023	"			19.05	"
	%63.51				0.05
	"				"



.(2006) Shalhoub



: (5.19)

(5.19)

**0.719	

0.250 = (0.05) (56) \*

0.325 = (0.01) (56) \*\*

(5.19)

0.01

(2001) Godwin

(  $\alpha \leq 0.05$  )

:

(  $\alpha \leq 0.05$  )

: (5.20)

(5.20)

67.59	4.40	23.66	1372	7	:
65.82	6.35	29.62	1718	9	:
72.28	3.98	18.07	1048	5	:
68.69	4.07	17.17	996	5	:
63.51	3.43	19.05	1105	6	:
<b>67.23</b>	<b>16.58</b>	<b>107.57</b>	<b>6239</b>	<b>32</b>	

(5.20)

%67.23

# Digital Signature program

## Digital Signature program

( ) - -  
) (Text Files) (  
) (  
)

Functions

6.1

.C++

Ms. Access

Microsoft Office

(williamstallings.com)

Hashing

:

: Hashing

:

Hashing

:

:Snerfu .1

:

Hashing

256Bit

128Bit

.1

XOR

.2

:

Hashing Birthday

.1

:MD2, MD4, MD5 .2

:

.1

Hashing

128BIT

.2

4

.3

:

Hashing

.1

Hashing Birthday

.2

Snerfu

.MD5

MD2

MD4

MD2

.3

MD5

**SHA1 .3**

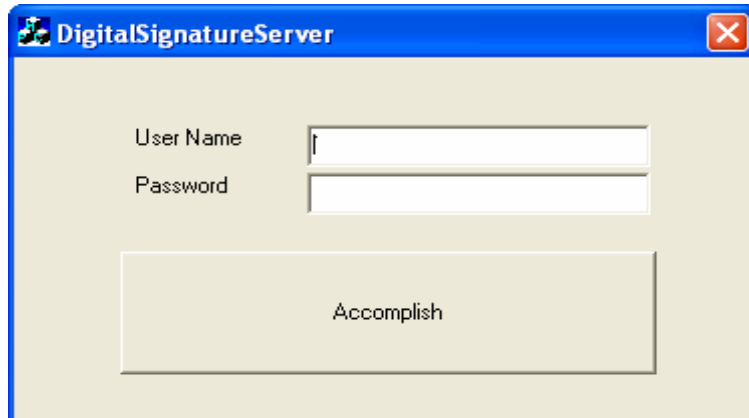
NSA NIST  
: MD5  
Hashing 160Bit .3  
.4  
.Hashing Birthday  
.5

**(SHA1)**

RSA

**6.3**

.1



(6.1)

)

.(

:

User Name

(6.1)

Password

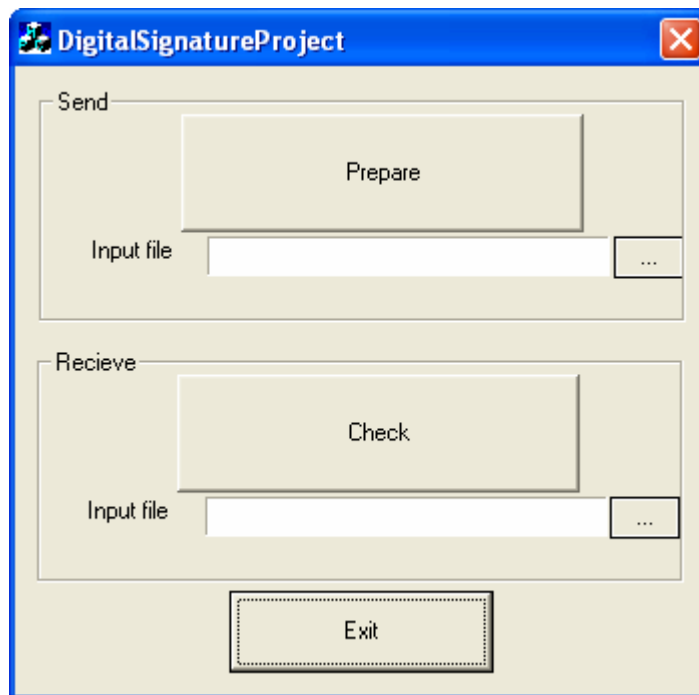
Accomplish

(6.2)

digest

:

.2



(6.2)

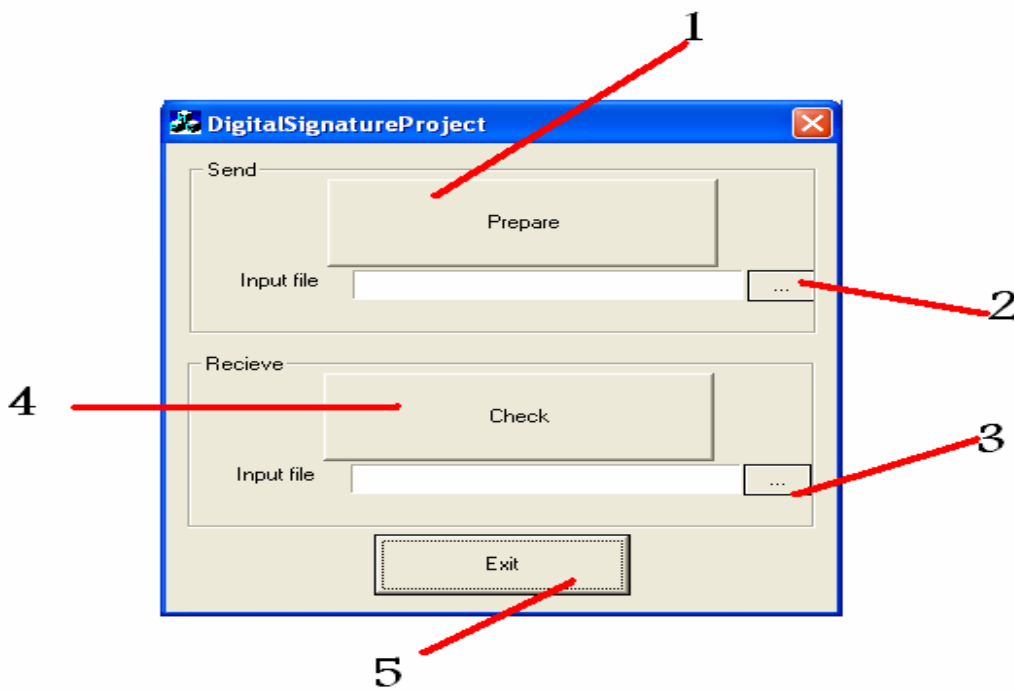
(6.2) (6.1)



(6.3)

(2)  
(6.4)

.1



(6.4)





(6.5)

(1)

(.out)

(3)

(6.6)

: .3

(6.4)



(6.6)

(4)

(6.4)

:

.

## الفصل السابع

:  
:  
:

:

.1

" "

.2 72.41%

.3 74.14%

.4 51.72%

.5 32.76%

)

.6 60.34%

(

.8

.9

.10

.11

.12

.13

	.14	% 78.62
	.15	
	.16	%72.76
	.17	%71.72
	.18	
	.19	
( $\alpha \leq 0.05$ )	.20	
	.21	
( $\alpha \leq 0.05$ )	.22	
	.23	
$\alpha \leq 0.05$ )	.24	
"		
( $\alpha \leq 0.05$ )		

:

-:

•

•

•

•

•

•

•

•

•

•

•

- 
- 
- 
- 
- 
- 
- 
- 
- 
- 
-

:

"

"

"

"

.

:

:

•

•

.



## المراجع

:

:

206

- ) :
- . (
- :" :
- " (2004) •
- 2005 " " •
- 2004 •
- 2006 -2005 •
- " " (2001) •
- :" (2006) •
- " " " /
- 2006 /
- " (2003) •
- " " " •
- " - " (2005) •
- 2005

:

Borasky, Danielle V. ,(1999), "**Digital Signatures**", Jul/Aug 99, vol.23, Issue 4, P47, 4P, 1 diagram. ISSN: 0146-5422.

CGI,(2004),"**Public Keys Encryption and digital signature: How do they work?**"

[http://www.cgi.com/web/en/library/white\\_papers/p1.htm](http://www.cgi.com/web/en/library/white_papers/p1.htm)

Christopher Ashley, (2007), "**MEMORANDUM :E-Signatures in University Transactions**", [www.uww.edu/Adminforms/esigauthor.doc](http://www.uww.edu/Adminforms/esigauthor.doc) , (last Access 1 March 2007).

David C. Chou, David C. Yen, (2001)," **intranets for organizational innovation**", Information Management & Computer Security, 2001, Vol. 9/2, P80- 87.

Department of Information Resources (2002), "**Guidelines for the Management of Electronic Transactions and Signed Records**", Prepared by the UETA Task Force of the Department of Information Resources and the Texas State Library and Archives Commission ,September 2002.

[http://www.tsl.state.tx.us/slr/recordspubs/UETA\\_guidelines.doc](http://www.tsl.state.tx.us/slr/recordspubs/UETA_guidelines.doc)

Edward H. , Freeman J.D. ,(2004)"**Digital Signatures and Electronic Contracts**", Information Systems Security , May/June 2004.

Fredric G., (1994),"**Protection of Electronic Mail and electronic Messages Challenges and Solutions**", Security and Control Products, Nashua, New Hampshire,USA.

Gabor Moroc, (2004), "**Summary of Current Law on Electronically Generated and Digital Signatures**" , [http://www.csdaca.org/conferences/Automation%20impacts%20on%20Attorney%20Ethical%20Issues\\_MCLE/DCSS%20Digitizd%20signature%20Summary.do](http://www.csdaca.org/conferences/Automation%20impacts%20on%20Attorney%20Ethical%20Issues_MCLE/DCSS%20Digitizd%20signature%20Summary.do) (Last Access Marsh 2007 )

Godwin J. Udo, (2001)," **Privacy and Security concerns as major**

**barriers for e-commerce: a survey study"**, Information Management & Computer Security, 2001, Vol. 3/8 , P143-13.

Vesna Hassler, Helmut Biely, (1999), "**Digital Signature Management**", Electronic Networking Applications and Policy, Volume 9 Number 4, 1999 pp. 262-271, ISSN 1066-2243

Mike Horton, Clinton Mugge ,(2003), "**Hack Notes**, Network Security – Potable Reference ", McGraw-Hill companies, ISBN: 0-07-2227883-4, USA.

Jones, Jennifer, Johnston, Margret, (2000), "**Digital signature bill enables e-commerce**", InfoWorld; 06/19/2000, Vol. 22 Issue 25, p8, 1/2p, 1 diagram.

Judith V. Boettcher and Amanda Powell , "**Digital Certificates, What Are They, and What Are They Doing in My Browser?**" [www.cren.net/crenca/docs/syllabus.doc](http://www.cren.net/crenca/docs/syllabus.doc) , (last Access March 2007)

Kostas Moulinos & others,(2004),"**Towards Secure sealing of Privacy Policies**", Information Management & Computer Security, Volume 14 Number 2, 2006 pp. 104-115, Emerald Group Publishing limited, 0968-5227.

Kuechler W. , Grupe F. ,(2002),"**Digital Signatures: A Business View**", Information System Security, Mar/April 2002, Vol.11 Issue 1, P23, 13P.

Kwo-shing Hong & others, (2006), "**An empirical study of information security policy on information security elevation in Taiwan**", Information Management & Computer Security, Volume 12 Number 4, 2004 pp. 350-361, Emerald Group Publishing limited, 0968-5227.

Meckbach, Greg, (1998),"**Digital signature reaches new level**", Computing Canada; 06/22/98, Vol. 24 Issue 24, p13, 2p, 1c.

Minihan, Jim , (2001),"**Electronic Signature Technologies: A Tutorial**", Information Management & Computer Security, Volume 9 Number 4 2001 pp. 165-174, ISSN 0968-5227.

Radcliff, Deborah, (2000),"**Digital Signatures**", Computerworld; 4/10/2000, Vol. 34 Issue 15, p64, 1p.

Rodney J. Petersen, (2004), " **A Framework for IT Policy Development**" *Educause Review*, vol. 39, no. 2 (March/April 2004): P54–55.  
<http://www.educause.edu/apps/er/erm04/erm0428.asp>

Stephen Wilson, (1999), "**Digital Signatures and the Future of Documentation**", *Information Management & Computer Security*, Volume 7 Number 2 1999 pp. 83-87, ISSN 0968-5227.

Whitaker, David, (2007), "**E-Signatures in the Higher Education Environment**", [www.abanet.net](http://www.abanet.net) (last Access 1 March 2007).

White, Ron, (2000), "**Digital signatures**", *PC Computing*; Mar2000, Vol. 13 Issue 3, p152, 2p, 4c, ISSN: 0899-1847.

Yakal, Kathy, (2000), "**Make Approvals Bulletproof**", *PC Magazine*; 09/19/2000, Vol. 19 Issue 16, p34, 1/3p.

[http://www.uncitral.org/uncitral/ar/uncitral\\_texts/electronic\\_commerce/1996Model.html](http://www.uncitral.org/uncitral/ar/uncitral_texts/electronic_commerce/1996Model.html) ,(Last Access Nov.2007).

(Wikipedia)

[http://en.wikipedia.org/wiki/digital\\_signature](http://en.wikipedia.org/wiki/digital_signature) (Last access March 2007).

(entrust)

<http://www.entrust.com/digitalsig/description.htm> (Last access April 2007)

<http://www.alaqsa.edu.ps/arabic/ccenter/default.asp> , (Last access April 2007)

<http://www.iugaza.edu.ps/ara/it/> (Last access April 2007)

- [http://www.alazhar.edu.ps/arabic/Centers/Information\\_Technology\\_Unit/ITU.htm](http://www.alazhar.edu.ps/arabic/Centers/Information_Technology_Unit/ITU.htm) (Last access April 2007)
- <http://www.qou.edu/homePage/arabic/index.jsp?pageId=93>, (Last access April 2007)
- (Teletrust )  
[www.teletrust.de/fileadmin/files/AG7\\_Flyer2001\\_e\\_24.doc](http://www.teletrust.de/fileadmin/files/AG7_Flyer2001_e_24.doc) ,(Last access April 2007)
- (Microsoft)  
<http://support.microsoft.com/kb/195724> ,Article ID 195724, Last Review January 23, 2007, (Last access September 2007).
- (IBM)  
<http://www.ibm.com/developerworks/lotus/library/securemessaging/>
- (entrust)  
[www.entrust.com/resources/pdf/cryptointro.pdf](http://www.entrust.com/resources/pdf/cryptointro.pdf) - 2007-01-23
- North Carolina University  
<http://www.unc.edu/~dvb/cyberlaw/digitalsignatures/security.htm#Advantages> , (Last access April 2007).
- University of Michigan  
**"Guidelines for Implementing the Proper Use Policy of the University of Michigan: Responsible Use of Technology Resources"** , December 2002, <http://spg.umich.edu/> , (last Access 1Sep. 2007).
- University of New York  
**"State University of New York at Stony brook responsible use of information technology policy"** , April 2001,  
<http://naples.cc.sunysb.edu/Admin/policy.nsf/pages/P109#web#web>  
(last Access 1Sep. 2007).
- Michigan Technological University  
<http://www.cs.mtu.edu/~yinma/Index.html> (last Access 11Nov. 2007).

- Arizona State University  
[www.asu.edu/ecure/2002/dollar/index.html](http://www.asu.edu/ecure/2002/dollar/index.html)
- <http://www.definethat.com/define/5691.htm>
- <http://computer.howstuffworks.com/question571.htm> ,(Last access February 2007).
- <http://www.us-cert.gov/cas/tips/ST04-018.html> ,(Last access march 2007)
- [www.e-signature.gov.eg/materials/hwhab\\_e-signatuere\\_different.ppt](http://www.e-signature.gov.eg/materials/hwhab_e-signatuere_different.ppt)  
(Last access April 2007)
- [http://searchsecurity.techtarget.com/sDefinition/0,,sid14\\_gci11953,00.html](http://searchsecurity.techtarget.com/sDefinition/0,,sid14_gci11953,00.html) (Last access April 2007)
- [http://pr-qa.com/solutions/qacpp\\_benefits2.htm](http://pr-qa.com/solutions/qacpp_benefits2.htm)
- <http://williamstallings.com/Extras/Security-otes/lectures/authent.html#fn>

## ملاحق الدراسة





(1)

:

-

.. ...

.

.

- - \_\_\_\_\_

.

\_\_\_\_\_ :

( )

.

.

x

:

- . :

-

.

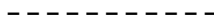
:  
-1

:  
-2

.  
-3

.  
-4

( )



-5

10  10 - 6  5 -

-6

10  10 - 6  5 -

.( ) -7

-----:( ) 


 .( ) 




-8

5  5 - 2  - " "  -

-9

-10

-11

:(

):

" "


:( )

-12

)

(

-13

-14

:

:						
						.15
						.16
						.17
						.18
						.19
						.20
						.21

:						
						.22
						.23
						.24
						.25
						.26
						.27
					(Digital Signatures)	.28
						.29
						.30

						:
						.31
						.32
						.33
						.34
						.35

						:
						.36
						.37
						.38
						.39
						.40

						:
						.41
						.42
						.43
						.44
						.45
						.46

(2)

7.1

```
void CDigitalSignatureServerDlg::OnAccomplish()
{
    UpdateData();
    //Step-1 Generating Random Numbers
    GeneratePrimeNumbers();

    //Step-2 calculating n=p*q
    m_n = m_Prime1 * m_Prime2;
    //Step-3  $\phi=(p-1)(q-1)$ 
    m_Undef = (m_Prime1-1) * (m_Prime2-1);
    //Step-4 Selecting 'e'
    SelectE();
    //m_e=17;
    //Step-5 Calculate 'd'
    CalculateD();
    //Displaying private and public keys
    //(1) Public key KU={e,n}

    //(2) Private key KU={d,n}

    if (ptrCDigitalSignatureDB.IsOpen())
    ptrCDigitalSignatureDB.Close();

    ptrCDigitalSignatureDB.Open();

    ptrCDigitalSignatureDB.AddNew();

    ptrCDigitalSignatureDB.m_DS_User=m_UserName;
    ptrCDigitalSignatureDB.m_DS_Password=m_Password;
    ptrCDigitalSignatureDB.m_DS_d=m_d;

    ptrCDigitalSignatureDB.m_DS_e=m_e;
    ptrCDigitalSignatureDB.m_DS_n=m_n;
    ptrCDigitalSignatureDB.Update();

    MessageBox("Done Successfully");
}
```

7.2

Sha1.cpp

Sha1.h

Hashing

Class

**Sha1.h .a**

```
#if !defined(AFX_SHA1_H_892E0361_14F2_4E6C_85C9_853CF05AC5F8__INCLUDED_)
#define AFX_SHA1_H_892E0361_14F2_4E6C_85C9_853CF05AC5F8__INCLUDED_

/*
100% free public domain implementation of the SHA-1 algorithm
by Dominik Reichl <dominik.reichl@t-online.de>
Web: http://www.dominik-reichl.de/

Version 1.6 - 2005-02-07 (thanks to Howard Kapustein for patches)
- You can set the endianness in your files, no need to modify the
  header file of the CSHA1 class any more
  - Aligned data support
- Made support/compilation of the utility functions (ReportHash
  and HashFile) optional (useful, if bytes count, for example in
  embedded environments)

Version 1.5 - 2005-01-01
- 64-bit compiler compatibility added
- Made variable wiping optional (define SHA1_WIPE_VARIABLES)
  - Removed unnecessary variable initializations
- ROL32 improvement for the Microsoft compiler (using _rotl)

===== Test Vectors (from FIPS PUB 180-1) =====

SHA1("abc") =
A9993E36 4706816A BA3E2571 7850C26C 9CD0D89D

SHA1("abcdcbcdcedefdefgefghfghighijhijkjklklmklmnlmnomnopnopq") =
84983E44 1C3BD26E BAAE4AA1 F95129E5 E54670F1

SHA1(A million repetitions of "a") =
34AA973C D4C4DAA4 F61EEB2B DBAD2731 6534016F

*/

#ifdef __SHA1_HDR__
#define __SHA1_HDR__

#if !defined(SHA1_UTILITY_FUNCTIONS) && !defined(SHA1_NO_UTILITY_FUNCTIONS)
#define SHA1_UTILITY_FUNCTIONS
#endif

#include <memory.h> // Needed for memset and memcpy

#ifdef SHA1_UTILITY_FUNCTIONS
#include <stdio.h> // Needed for file access and printf
#include <string.h> // Needed for strcat and strcpy
#endif

#ifdef _MSC_VER
```

```

#include <stdlib.h>
#endif

// You can define the endian mode in your files, without modifying the SHA1
// source files. Just #define SHA1_LITTLE_ENDIAN or #define SHA1_BIG_ENDIAN
// in your files, before including the SHA1.h header file. If you don't
// define anything, the class defaults to little endian.

#if !defined(SHA1_LITTLE_ENDIAN) && !defined(SHA1_BIG_ENDIAN)
#define SHA1_LITTLE_ENDIAN
#endif

// Same here. If you want variable wiping, #define SHA1_WIPE_VARIABLES, if
// not, #define SHA1_NO_WIPE_VARIABLES. If you don't define anything, it
// defaults to wiping.

#if !defined(SHA1_WIPE_VARIABLES) && !defined(SHA1_NO_WIPE_VARIABLES)
#define SHA1_WIPE_VARIABLES
#endif

/////////////////////////////////////////////////////////////////
// Define 8- and 32-bit variables

#ifndef UINT_32

#ifdef _MSC_VER

#define UINT_8 unsigned __int8
#define UINT_32 unsigned __int32

#else

#define UINT_8 unsigned char

#if (ULONG_MAX == 0xFFFFFFFF)
#define UINT_32 unsigned long
#else
#define UINT_32 unsigned int
#endif

#endif

#endif
#endif

/////////////////////////////////////////////////////////////////
// Declare SHA1 workspace

typedef union
{
    UINT_8 c[64];
    UINT_32 l[16];
} SHA1_WORKSPACE_BLOCK;

class CSHA1
{
public:
#ifdef SHA1_UTILITY_FUNCTIONS
// Two different formats for ReportHash(...)
enum
{
    REPORT_HEX = 0,

```



```

REPORT_DIGIT = 1
};
#endif

// Constructor and Destructor
CSHA1();
~CSHA1();

UINT_32 m_state[5];
UINT_32 m_count[2];
UINT_32 __reserved1[1];
UINT_8 m_buffer[64];
UINT_8 m_digest[20];
UINT_32 __reserved2[3];

void Reset();

// Update the hash value
void Update(UINT_8 *data, UINT_32 len);
#ifdef SHA1_UTILITY_FUNCTIONS
bool HashFile(char *szFileName);
#endif

// Finalize hash and report
void Final();

// Report functions: as pre-formatted and raw data
#ifdef SHA1_UTILITY_FUNCTIONS
void ReportHash(char *szReport, unsigned char uReportType = REPORT_HEX);
#endif

void GetHash(UINT_8 *puDest);

private:
// Private SHA-1 transformation
void Transform(UINT_32 *state, UINT_8 *buffer);

// Member variables
UINT_8 m_workspace[64];
SHA1_WORKSPACE_BLOCK *m_block; // SHA1 pointer to the byte array above
};

#endif
#endif // !defined(AFX_SHA1_H_892E0361_14F2_4E6C_85C9_853CF05AC5F8_INCLUDED_)
//{{AFX_INSERT_LOCATION}}
// Microsoft Visual C++ will insert additional declarations immediately before the previous line.

#endif // !defined(AFX_SHA1_H_892E0361_14F2_4E6C_85C9_853CF05AC5F8_INCLUDED_)
#endif // !defined(AFX_SHA1_H_892E0361_14F2_4E6C_85C9_853CF05AC5F8_INCLUDED_)

```

## Sha1.cpp .b

```

/*
100% free public domain implementation of the SHA-1 algorithm
by Dominik Reichl <dominik.reichl@t-online.de>
Web: http://www.dominik-reichl.de/

```

```

Version 1.6 - 2005-02-07 (thanks to Howard Kapustein for patches)
- You can set the endianness in your files, no need to modify the
header file of the CSHA1 class any more

```

- Aligned data support
- Made support/compilation of the utility functions (ReportHash and HashFile) optional (useful, if bytes count, for example in embedded environments)

Version 1.5 - 2005-01-01

- 64-bit compiler compatibility added
- Made variable wiping optional (define SHA1\_WIPE\_VARIABLES)
- Removed unnecessary variable initializations
- ROL32 improvement for the Microsoft compiler (using \_rotl)

===== Test Vectors (from FIPS PUB 180-1) =====

SHA1("abc") =  
A9993E36 4706816A BA3E2571 7850C26C 9CD0D89D

SHA1("abcdcbcdcedefdefgefghfghighijhijkijkljklmklmnlmnomnopnopq") =  
84983E44 1C3BD26E BAAE4AA1 F95129E5 E54670F1

SHA1(A million repetitions of "a") =  
34AA973C D4C4DAA4 F61EEB2B DBAD2731 6534016F

```

*/
#include "stdafx.h"
#include "DigitalSignatureProject.h"
#include "SHA1.h"

#ifdef _DEBUG
#define new DEBUG_NEW
#undef THIS_FILE
static char THIS_FILE[] = __FILE__;
#endif

#ifdef SHA1_UTILITY_FUNCTIONS
#define SHA1_MAX_FILE_BUFFER 8000
#endif

// Rotate x bits to the left
#ifdef ROL32
#define ROL32(_val32, _nBits) _rotl(_val32, _nBits)
#else
#define ROL32(_val32, _nBits) (((_val32)<<(_nBits))|((_val32)>>(32-(_nBits))))
#endif

#ifdef SHA1_LITTLE_ENDIAN
#define SHABLK0(i) (m_block->l[i] = \
(ROL32(m_block->l[i],24) & 0xFF00FF00) | (ROL32(m_block->l[i],8) & 0x00FF00FF))
#else
#define SHABLK0(i) (m_block->l[i])
#endif

#define SHABLK(i) (m_block->l[i&15] = ROL32(m_block->l[(i+13)&15] ^ m_block->l[(i+8)&15] \
^ m_block->l[(i+2)&15] ^ m_block->l[i&15],1))

// SHA-1 rounds
#define _R0(v,w,x,y,z,i) { z+=((w&(x^y))^y)+SHABLK0(i)+0x5A827999+ROL32(v,5); \
w=ROL32(w,30); }

```

```

#define _R1(v,w,x,y,z,i) { z+=((w&(x^y))^y)+SHABLK(i)+0x5A827999+ROL32(v,5);
                                                                    w=ROL32(w,30); }
#define _R2(v,w,x,y,z,i) { z+=(w^x^y)+SHABLK(i)+0x6ED9EBA1+ROL32(v,5); w=ROL32(w,30); }
#define _R3(v,w,x,y,z,i) { z+=(((w|x)&y)|(w&x))+SHABLK(i)+0x8F1BBCDC+ROL32(v,5);
                                                                    w=ROL32(w,30); }
#define _R4(v,w,x,y,z,i) { z+=(w^x^y)+SHABLK(i)+0xCA62C1D6+ROL32(v,5); w=ROL32(w,30); }

CSHA1::CSHA1()
{
    m_block = (SHA1_WORKSPACE_BLOCK *)m_workspace;

    Reset();
}

CSHA1::~CSHA1()
{
    Reset();
}

void CSHA1::Reset()
{
    // SHA1 initialization constants
    m_state[0] = 0x67452301;
    m_state[1] = 0xEFCDAB89;
    m_state[2] = 0x98BADCFE;
    m_state[3] = 0x10325476;
    m_state[4] = 0xC3D2E1F0;

    m_count[0] = 0;
    m_count[1] = 0;
}

void CSHA1::Transform(UINT_32 *state, UINT_8 *buffer)
{
    // Copy state[] to working vars
    UINT_32 a = state[0], b = state[1], c = state[2], d = state[3], e = state[4];

    memcpy(m_block, buffer, 64);

    // 4 rounds of 20 operations each. Loop unrolled.
    _R0(a,b,c,d,e, 0); _R0(e,a,b,c,d, 1); _R0(d,e,a,b,c, 2); _R0(c,d,e,a,b, 3);
    _R0(b,c,d,e,a, 4); _R0(a,b,c,d,e, 5); _R0(e,a,b,c,d, 6); _R0(d,e,a,b,c, 7);
    _R0(c,d,e,a,b, 8); _R0(b,c,d,e,a, 9); _R0(a,b,c,d,e,10); _R0(e,a,b,c,d,11);
    _R0(d,e,a,b,c,12); _R0(c,d,e,a,b,13); _R0(b,c,d,e,a,14); _R0(a,b,c,d,e,15);
    _R1(e,a,b,c,d,16); _R1(d,e,a,b,c,17); _R1(c,d,e,a,b,18); _R1(b,c,d,e,a,19);
    _R2(a,b,c,d,e,20); _R2(e,a,b,c,d,21); _R2(d,e,a,b,c,22); _R2(c,d,e,a,b,23);
    _R2(b,c,d,e,a,24); _R2(a,b,c,d,e,25); _R2(e,a,b,c,d,26); _R2(d,e,a,b,c,27);
    _R2(c,d,e,a,b,28); _R2(b,c,d,e,a,29); _R2(a,b,c,d,e,30); _R2(e,a,b,c,d,31);
    _R2(d,e,a,b,c,32); _R2(c,d,e,a,b,33); _R2(b,c,d,e,a,34); _R2(a,b,c,d,e,35);
    _R2(e,a,b,c,d,36); _R2(d,e,a,b,c,37); _R2(c,d,e,a,b,38); _R2(b,c,d,e,a,39);
    _R3(a,b,c,d,e,40); _R3(e,a,b,c,d,41); _R3(d,e,a,b,c,42); _R3(c,d,e,a,b,43);
    _R3(b,c,d,e,a,44); _R3(a,b,c,d,e,45); _R3(e,a,b,c,d,46); _R3(d,e,a,b,c,47);
    _R3(c,d,e,a,b,48); _R3(b,c,d,e,a,49); _R3(a,b,c,d,e,50); _R3(e,a,b,c,d,51);
    _R3(d,e,a,b,c,52); _R3(c,d,e,a,b,53); _R3(b,c,d,e,a,54); _R3(a,b,c,d,e,55);
    _R3(e,a,b,c,d,56); _R3(d,e,a,b,c,57); _R3(c,d,e,a,b,58); _R3(b,c,d,e,a,59);
    _R4(a,b,c,d,e,60); _R4(e,a,b,c,d,61); _R4(d,e,a,b,c,62); _R4(c,d,e,a,b,63);
    _R4(b,c,d,e,a,64); _R4(a,b,c,d,e,65); _R4(e,a,b,c,d,66); _R4(d,e,a,b,c,67);
    _R4(c,d,e,a,b,68); _R4(b,c,d,e,a,69); _R4(a,b,c,d,e,70); _R4(e,a,b,c,d,71);
    _R4(d,e,a,b,c,72); _R4(c,d,e,a,b,73); _R4(b,c,d,e,a,74); _R4(a,b,c,d,e,75);
    _R4(e,a,b,c,d,76); _R4(d,e,a,b,c,77); _R4(c,d,e,a,b,78); _R4(b,c,d,e,a,79);
}

```

```

// Add the working vars back into state
state[0] += a;
state[1] += b;
state[2] += c;
state[3] += d;
state[4] += e;

// Wipe variables
#ifdef SHA1_WIPE_VARIABLES
a = b = c = d = e = 0;
#endif
}

// Use this function to hash in binary data and strings
void CSHA1::Update(UINT_8 *data, UINT_32 len)
{
    UINT_32 i, j;

    j = (m_count[0] >> 3) & 63;
    if((m_count[0] += len << 3) < (len << 3)) m_count[1]++;

    m_count[1] += (len >> 29);

    if((j + len) > 63)
    {
        i = 64 - j;
        memcpy(&m_buffer[j], data, i);
        Transform(m_state, m_buffer);
    }
    for (; i + 63 < len; i += 64) Transform(m_state, &data[i]);

    j = 0;
    else i = 0;

    memcpy(&m_buffer[j], &data[i], len - i);
}

#ifdef SHA1_UTILITY_FUNCTIONS
// Hash in file contents
bool CSHA1::HashFile(char *szFileName)
{
    unsigned long ulFileSize, ulRest, ulBlocks;
    unsigned long i;
    UINT_8 uData[SHA1_MAX_FILE_BUFFER];
    FILE *fIn;

    if(szFileName == NULL) return false;

    fIn = fopen(szFileName, "r");
    if(fIn == NULL) return false;

    fseek(fIn, 0, SEEK_END);
    ulFileSize = (unsigned long)ftell(fIn);
    fseek(fIn, 0, SEEK_SET);

    if(ulFileSize != 0)
    {

```

```

        ulBlocks = ulFileSize / SHA1_MAX_FILE_BUFFER;
        ulRest = ulFileSize % SHA1_MAX_FILE_BUFFER;
    }
    else
    {
        ulBlocks = 0;
        ulRest = 0;
    }

    for(i = 0; i < ulBlocks; i++)
    {
        fread(uData, 1, SHA1_MAX_FILE_BUFFER, fIn);
        Update((UINT_8 *)uData, SHA1_MAX_FILE_BUFFER);
    }

    if(ulRest != 0)
    {
        fread(uData, 1, ulRest, fIn);
        Update((UINT_8 *)uData, ulRest);
    }

    fclose(fIn); fIn = NULL;
    return true;
}
#endif

void CSHA1::Final()
{
    UINT_32 i;
    UINT_8 finalcount[8];

    for(i = 0; i < 8; i++)
        finalcount[i] = (UINT_8)((m_count[((i >= 4) ? 0 : 1)]
        >> ((3 - (i & 3)) * 8) ) & 255); // Endian independent

    Update((UINT_8 *)"\200", 1);

    while ((m_count[0] & 504) != 448)
        Update((UINT_8 *)"\0", 1);

    Update(finalcount, 8); // Cause a SHA1 Transform()

    for(i = 0; i < 20; i++)
    {
        m_digest[i] = (UINT_8)((m_state[i >> 2] >> ((3 - (i & 3)) * 8) ) & 255);
    }

    // Wipe variables for security reasons
    #ifdef SHA1_WIPE_VARIABLES
        i = 0;
        memset(m_buffer, 0, 64);
        memset(m_state, 0, 20);
        memset(m_count, 0, 8);
        memset(finalcount, 0, 8);
        Transform(m_state, m_buffer);
    #endif
}

#ifdef SHA1_UTILITY_FUNCTIONS
// Get the final hash as a pre-formatted string

```

```

void CSHA1::ReportHash(char *szReport, unsigned char uReportType)
{
    unsigned char i;
    char szTemp[16];

    if(szReport == NULL) return;

    if(uReportType == REPORT_HEX)
    {
        sprintf(szTemp, "%02X", m_digest[0]);
        strcat(szReport, szTemp);

        for(i = 1; i < 20; i++)
        {
            sprintf(szTemp, " %02X", m_digest[i]);
            strcat(szReport, szTemp);
        }
    }
    else if(uReportType == REPORT_DIGIT)
    {
        sprintf(szTemp, "%u", m_digest[0]);
        strcat(szReport, szTemp);

        for(i = 1; i < 20; i++)
        {
            sprintf(szTemp, " %u", m_digest[i]);
            strcat(szReport, szTemp);
        }
    }
    else strcpy(szReport, "Error: Unknown report type!");
}
#endif

// Get the raw message digest
void CSHA1::GetHash(UINT_8 *puDest)
{
    memcpy(puDest, m_digest, 20);
}

```

## Digest ◆

```

void CDigitalSignatureProjectDlg::OnMakeDigest()
{
    char InFile[200];
    char OutFile[200];

    UpdateData();
    strcpy(InFile, m_InputFile);
    strcpy(OutFile, m_OutputFile);

    HashFile(InFile, OutFile);
}

//2- Get the public key of the current user from the database server
//search for the user name in the database
UpdateData();
if(ptrDigitalSignatureDB.IsOpen())
ptrDigitalSignatureDB.Close();

```

```

ptrDigitalSignatureDB.Open(0,"SELECT * FROM Users WHERE
                        Users.DS_User='"+m_UserName+"'");
ptrDigitalSignatureDB.MoveFirst();

}

Encrypt("c:\\Temp.txt", m_OutputFile);

ptrDigitalSignatureDB.Close();

}

: ◆

void CDigitalSignatureProjectDlg::OnPrapare()
{
    char InFile[200];

    UpdateData();
    strcpy(InFile,m_InputFile);

    //1-Doing Hash Code////////////////////////////////////
    HashFile(InFile,"c:\\TempSource1.hc");

    //2- combine the original file with the hash code file////////////////////////////////
    Combine(InFile,"c:\\TempSource1.hc","c:\\TempSource2.cmb");

    //4- Doing RSA encryption //////////////////////////////////
    Encrypt("c:\\TempSource2.cmb", "c:\\TempSource2.enc");

    //5- Combine the user name and the enc block

    char Buffer[200];
    strcpy(Buffer,InFile);

    Buffer[strlen(InFile)-3]=NULL;
    strcat(Buffer,"out");

    Combine(1,"c:\\TempSource2.enc",Buffer);

}

◆

void CDigitalSignatureProjectDlg::Oncheck()
{
    char InFile[200];

    UpdateData();
    strcpy(InFile,m_InputReceivingFile);

```

```

//0- seperate the original file and the User Name////////////////////
CString SenderUser;
Seperate(1 ,&SenderUser,InFile ,"c:\\TempDist1.inc");//org=original one

if(MessageBox("Do you want to proceed with a letter comes from:"+SenderUser,"",4)!=6)
return;

//0- Quering from the database////////////////////////////////////
QeringForUser(SenderUser);

//1-Decrypt the incomming file //////////////////////////////////////
Decrypt("c:\\TempDist1.inc","c:\\TempDist1.dec");//dec= decrypted one

//2- seperate the original file and the hash code file////////////////////////////////////
Seperate("c:\\TempDist1.dec" ,"c:\\TempDist1.org","c:\\TempDist1.hc");//org=original one

//3- Doing Hashing for original one////////////////////////////////////
HashFile("c:\\TempDist1.org","c:\\TempDist2.hc");

//4- compare the 2 hashing code files
int CheckFlag=Compare("c:\\TempDist2.hc","c:\\TempDist1.hc");

if ( CheckFlag==1)
MessageBox("File error");
else if (CheckFlag==2)
MessageBox("Not indetical");
else if(CheckFlag==1)
{
MessageBox("OK");
ShellExecute(NULL,"open","c:\\TempDist1.org",NULL,NULL,SW_MAXIMIZE);
}
}

◆

void CDigitalSignatureProjectDlg::Encrypt(CString infile, CString outfile)
{
int NO_BITS=32;
double c=0,d=1;
char bits[100];
double n=(double)m_n;
unsigned char ch;
double data;//19;
long i,k=NO_BITS;
CString str;
int sizeof_d=sizeof(double);

FILE *inFP, *outFP;
inFP=fopen(infile,"r");

```



```

        if(inFP==NULL)
        {
str.Format("\nERROR: Couldn't open input file: %s\nAborting operation",infile);
        DumpNotes(str); //
        return;
        }

        outFP=fopen(outfile,"w");
        if(outFP==NULL)
        {
str.Format("\nERROR: Couldn't create output file: %s\nAborting operation",outfile);
        DumpNotes(str); //
        fcloseall();
        return;
        }

        //change integer 'm' to binary
        D_to_B(m_e,32,bits);
        GetOnlyProperBits(bits);
        k=NO_BITS = strlen(bits)-1;

        while(1)
        {
//read from ip file into 'data'
        if(fread(&ch,1,1,inFP)==0)
            break;
        data =(double)ch;

//calculate ((data)^e mod n) .i.e the remainder
        c=0;d=1;
        for(i=k;i>=0;i--)
        {
            c=2*c;
            d=fmod(d*d,n);

            if(bits[NO_BITS-i] == '1')
            {
                c=c+1;
                d=fmod(data*d,n);
            }//end of IF
        }//end of for loop

        //write to op file from 'd'
        fwrite(&d,sizeof_d,1,outFP);

        }//end of while loop

        fcloseall();
    }//end of encrypt

```



void CDigitalSignatureProjectDlg::Decrypt(CString infile, CString outfile)

```

int NO_BITS;
double c=0,d=1;
char bits[100];
double n=(double)m_n1;
double data;//19;
long i,k;
CString str;
int sizeof_d=sizeof(double);

FILE *inFP, *outFP;
inFP=fopen(infile,"r");
if(inFP==NULL)
{
str.Format("\nERROR: Couldn't open input file: %s\nAborting operation",infile);
return;
}

outFP=fopen(outfile,"w");
if(outFP==NULL)
{
str.Format("\nERROR: Couldn't create output file: %s\nAborting operation",outfile);
fcloseall();
return;
}

//change integer 'd' to binary
D_to_B(m_d,32,bits);
GetOnlyProperBits(bits);
k=NO_BITS = strlen(bits)-1;

while(1)
{
//read from ip file into 'data'
if(fread(&data,sizeof_d,1,inFP)==0)
break;

//calculate ((data)^e mod n) .i.e the remainder
c=0;d=1;
for(i=k;i>=0;i--)
{
c=2*c;
d=fmod(d*d,n);

if(bits[NO_BITS-i] == '1')
{
c=c+1;
d=fmod(data*d,n);
} //end of IF
} //end of for loop

//write to op file from 'd'
unsigned char ch=(unsigned char)d;
fprintf(outFP,"%c",ch);
//write(&ch,1,1,outFP);
}
}

```

```

} //end of while loop

fcloseall();

} //end of encrypt

```

## Digital Signature Project – Use Case Diagram

Digital Signature Project – Use Case Diagram

