Modify PGP Cryptography Protocol Using Hash Visualization Technique

Dr. Hala Bahjat Abdul Wahab
Computer Science Depart.
University of Technology
Baghdad, Iraq
hala_bahjat@yahoo.com

Abstract-

The strong cryptography employed by PGP is one of the best available today. The PGP protocol is a hybrid cryptosystem that combines some of the best features of both conventional and public-key cryptography. This paper aim to modify PGP protocol by combined between Hash Visualization Technique, parametric curve technique and PGP protocol stages to increase PGP protocol and make the protocol more difficult in front of the counterfeiter, by use generated digital images capability for Hash visualization (random art technique) as input stage for PGP protocol and select the key according parametric curve equations ,instead of use random movements of mouse to generate the key in standard PGP protocol.

Keywords: Applied Cryptography, has visualization, root key validation, and PGP Protocol.

Modify PGP Cryptography Protocol Using Hash Visualization Technique

د.هاله بهجت عبد الوهاب
مدرس
قسم علم الحاسبات
الجامعه التكنولوجيه
العراق/بغداد
hala_bahjat@yahoo.com

الخلاصه:

هو واحد من اهم برتوكولات التشفير المتوفره حاليا والتي له ميزه السريه القويه, حيث انه يجمع افضل الصفات PGP الموجوده
في كل من الانظمه التشفير التقليديه (نظام المفتاح الواحد) وانظمة التشفير ذات المفتاحين. هذا البحث يطرح فكره تحسين هذا البرتوكول من خلال طرح فكره جديد في اشتقاق مفتاح الادخال المستخدم في هذا البرتوكول عن طريق دمج بين عدة تقنيات ومنها تقنيه فن توليد الصور الرقميه العشوائيه واستخدام هذه الصور كمرحله ادخال بديله عن مرحله ادخال البرتوكول التقليدي واشتقاق المفتاح المستخدم باعتماد على تقنية معادلات منحي ذات سريه عاليه لسحب المفتاح من الصور المولده بدل من اعتمادطريقةتحريك الماوس بشكل عشوائي لتوليد المفتاح في البرتوكول سابقا والتي تعتبر من الطرق القديمه وذات سريه ضعيفه, مما جعل اضافه هذه التقنيات تزيدد من سريه البرتوكول وجعله اكثر صعوبه امام المهاجم.

1. Introduction

Counterfeiting is a growing threat in recent years, especially with the ever-increasing growth of data communication, the need for security and privacy has become a necessity. Cryptography and data security are an essential requirement for communication privacy. Cryptography is the science of using mathematics to encrypt and decrypt data. Cryptography enables you to store sensitive information or transmit it across insecure networks (like the Internet) so that it cannot read by anyone except the intended recipient [1].

The new security primitive hash visualization, to establish the necessary requirements, to propose Random Art as a prototypical solution. *Random Art* was developed by Andrej Bauer, and is based on an idea of genetic art by Michael Witbrock and John Mount. Originally Random Art was conceived for automatic generation of artistic images. A brief overview and demonstration of Random Art can be found at [2].

In this paper added the image that generated form random art technique between the steps of PGP protocol in order to reach to cryptography protocol more robust in front of the counterfeiter.

2. Hash Visualization Algorithms

A hash function is a function h which has, as aminimum, the following two properties:

1. Compression: h maps an input x of arbitrary fnite length, to an output h(x) of fixed bit length n.
2. Ease of computation: given h and an input x,
   h(x) is easy to compute.

- Three most desired properties:

1. Preimage resistance: for any pre-specified output y, it is computationally infeasible to find the input x such that h(x) = y.
2. 2nd-preimage resistance: given any input x, it is computationally infeasible to find an input x' such that h(x') = h(x).
3. Collision resistance: it is computationally infeasible to find any two distinct inputs x; x'which hash to the same output, h(x) = h(x').

- A one-way hash function is a hash function h with two additional properties: pre-image resistance and 2nd-preimage resistance. A collision resistant hash

function is a hash function h with the additional property of collision resistance[3].

2.1 Requirements for hash visualization Algorithms.

A hash visualization algorithm(HVA) is a function hI which has, as a minimum, the following two properties:

1. Image-generation: hI maps an input x of arbitrary finite length, to an output image $h_I(x)$ of fixed size.
2. Ease of computation: given h and an input x, $h_I(x)$ is easy to compute[2].

3. Random Art[2].

Random Art, is define as a technique that converts meaningless strings into abstract structured images. The basic idea is to use a binary string s as a seed for a random number generator.

The randomness is used to construct a random expression, which describes a function generating the image mapping each image pixel to a color value.

The pixel coordinates range continuously from -1 to 1, in both x and y dimensions. The image resolution defines the sampling rate of the continuous image. For example, to generate a 100×100 image, we sample the function at 10000 locations. ***Random Art*** is an algorithm that give a bit-string as an input, it will generate a function $F :[-1,1]^2 \rightarrow [-1,1]^3$, which defines an image.

The bit-string input is used as a seed for the pseudo-random number generator, and the function is constructed by choosing rules from a grammar depending on the value of the pseudorandom number generator.

The randomness is used to construct a random expression which describes a function generating the image mapping each image pixel to a color value.

The pixel coordinates range continuously from -1 to 1, in both x and y dimensions. The image resolution defines the sampling rate of the continuous image. For example, to generate a 100×100 image, we sample the function at 10000 locations. ***Random Art*** is an algorithm that give a bit-string as an input, it will generate a function $F :[-1,1]^2 \rightarrow [-1,1]^3$, which defines an image.

The bit-string input is used as a seed for the pseudo-random number generator, and the function is constructed by choosing rules from a grammar depending on the value of the pseudorandom number generator.

3.1 Random Art algorithm [2].

The grammar used in the Random Art implementation is too large to be shown in this paper. Other functionsincluded are: sin, cos, exp, square root, division, mix.

The function mix (a; b; c; d) is a function which blends expressions c and d depending on the parameters a and b.

We show an example of an expression tree of depth 5 in figure 3, along with the corresponding image. For the other images in this paper, we used a depth of 12.

Pseudo-code for the Random Art algorithm is shown in Figure 4. The function rnd() used in the algorithm returns a random number in the range [0; 1). The purposeof the 'while' statement in step 5 is to make sure that the

expressions do not grow too fast with respect to depth d. in [3] more detail *Random Art algorithm.*

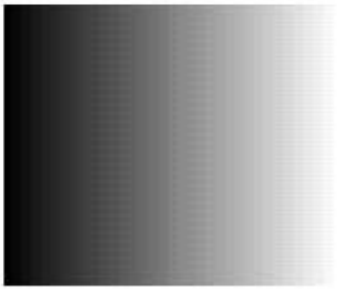
Algorithm RandomArt(G; i; d) [2].

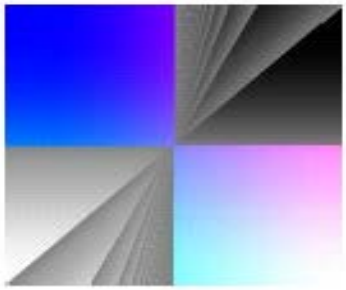
input: grammar G = [r1; : : : ; rn]
initial rule i
depth d
output: expression E
begin
(1) Suppose ri = [(a1; p1); : : : ; (ak; pk)].
(2) If d _ 0 then let a = a1 and goto step (4).
(3) Let a be one of (a1; : : : ; ak), picking ai with probability pi.
(4) If a is a terminal rule let E = a and go to step (6).
(5) Suppose a = f(ri1 ; : : : ; rim) where m is the arity of f.
While d _ 0 and rnd() _ 0:5 do d : = d _ 1.
For each j = 1; : : : ;m let Ej = RandomArt(G; ij; d_ 1).
Let E = f(E1; : : : ;Em).
(6) Return E.
end

Example [2]:

For example, the expression F(x, y) = (x, x, x) produces a horizontal gray grade, as shown in figure (1 -a). A more complicated example is shown in the following expression figure (1 -b).

If xy>0 then (x,y,1)
else (fmod(x, y),fmod(x, y),fmod(x, y))

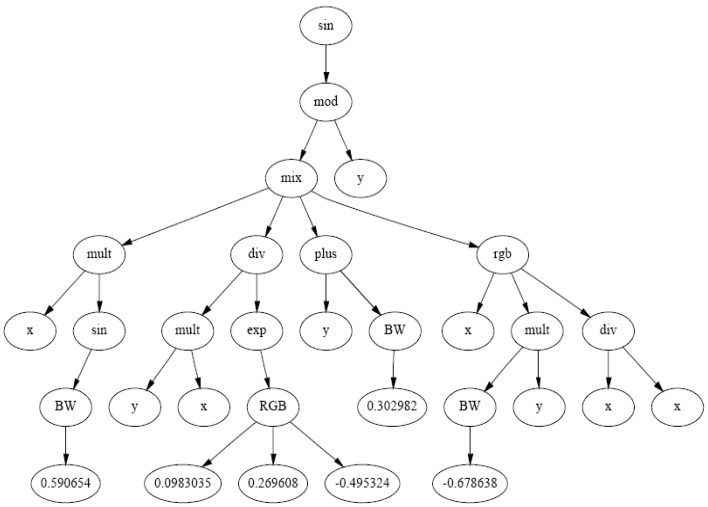




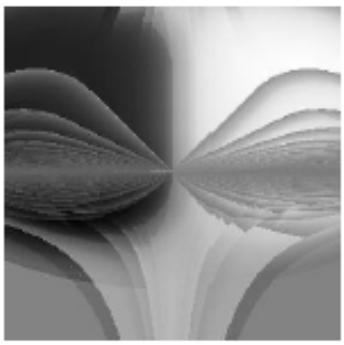
(a) Image for expression (x, xx) (b) Image for expression more complicated

.

Figure (1): Examples of images and corresponding expression.



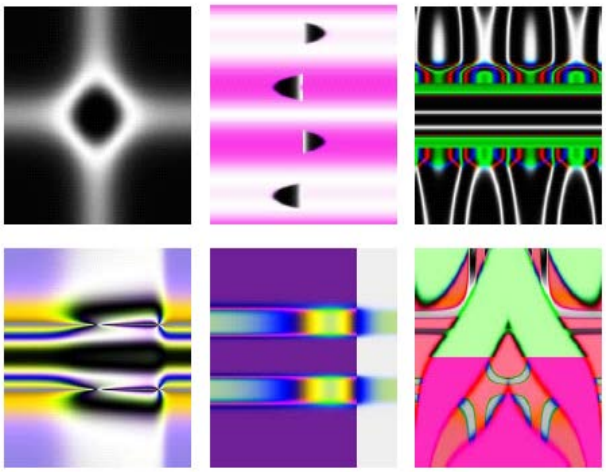
(a) Sample Random Art expression tree.



(b) Generated image

Figure (2): Random Art expression tree and the corresponding image [3].

The function F can also be seen as an expression tree, which is generated using a *grammar* G and a *depth parameterd*, which specifies the minimum depth of the expression tree that is generated. Figure (2) shows examples for other images using random art expression tree used a depth of 12, [2].



Figure(3): Examples of random art images.

Random mathematical formula to red, green and blue seed ,for rxample formules[6]:

Red=
cos(cos(arctan(cos(tan(mult(sin(tan(arctan(mult(mult(rgb(y[ ],r,x[ ]),(rgb(r,x[],x[ ])),bw(y[])))))),bw(y[])))))).

Green=
arctan(mult(cos(arctan(tan(mult(cos(tan(tan(arctan(mult(mult(tan(y[]),arctan(r)),cos(r))))), r)))), cos(y[ ]))).

Blue= if(y[],x[],mult(cos(mod(mix(if(reverse(exp(mod(div(sin(mix(sin(r),sin(x []),mix(y [],y[],r,y[]),bw(r))),bw(x[ ])),rgb(y[],y[ ],r)))),y [],add(x[],r)),y[],x[ ],add(x[ ],x[])),add(y[ ],y[ ])))),bw(y[]))).

The ***BW function*** is a user-defined function,which determines the gray color of input value and it can be calcuted using the equation is defined as follows:

G=trunc((R*0.30)+(G*0.11)+(B*0.59))

The ***revers function*** is a user-defined function,which determines the revers color of the input value and it can be calculated using the equation is defined as follows:

R=255-X

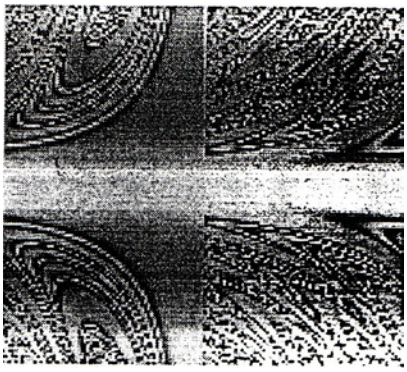
Where x is the input value to the reverse functionand'255' is the upper rangeof the colors range, whichis betwwen 0 and 255.

The *mix function* is user-defined function, which determines the mixingof two colors'a'and'b' depending on the value of 'c' and 'd'. it can be calculated using the equation is defined as follows:

M=(a*c)+(b*d)/(a+b)

Where a,b,c,d are the input color values.

Each formula is evaluated to produce one component value for each pixel (x,y). the three-color-component values define the R-value,G-value and b-value, they are mixed toproduce the final RGBvalue for each pixel.this willcreat the image.since each formula can be represented as tree with 13 depth, the red formula is of depthequal to12, green formula is of depth equal to12and blue formala is depth equal 13as shown in figure(4)[6].



Figure(4):The generated image .

4. Cryptographic Protocol (Pretty Good Privacy)[7].

Pretty Good Privacy (PGP) is a public key system for encrypting electronic mail using the RSA public key cipher. PGP combines some of the best features of both conventional and public-key cryptography. PGP is a hybrid cryptosystem. When a user encrypts plaintext with PGP, PGP first compresses the plaintext. Data compression saves modem transmission time and disk space and, more importantly, strengthens cryptographic security. Most cryptanalysis techniques exploit patterns found in the plaintext to crack the cipher. Compression reduces these patterns in the plaintext, thereby greatly enhancing resistance to cryptanalysis. (Files that are too short to compress or which do not compress well are not compressed.) PGP then creates a session key, which is a one-time-only secret key. This key is a random number generated from the random ovements of your mouse and the keystrokes you type. The session key works with a very secure, fast conventional encryption algorithm to encrypt the plaintext; the result is ciphertext. Once the data is encrypted, the session key is then encrypted to the recipient's public key. This public key-encrypted session key is transmitted along with the ciphertext to the recipient. Figure (5) shows the send process.

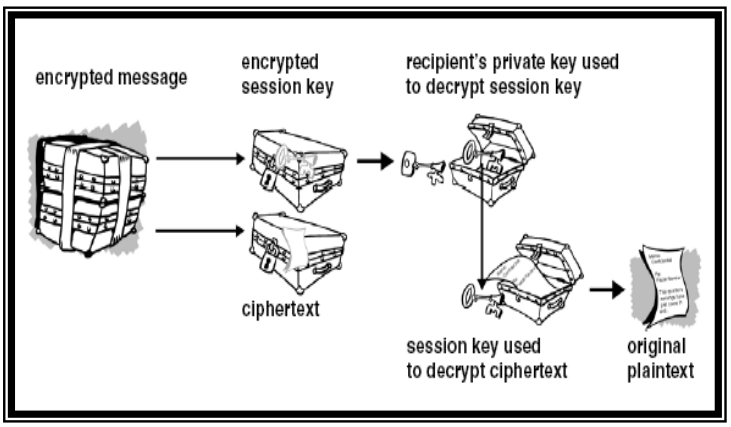


Figure (5): Send Process.

Decryption works in the reverse. The recipient's copy of PGP uses his or her private key to recover the session key, which PGP then uses to decrypt the conventionally encrypted ciphertext.

In figure (6) shown the received process.

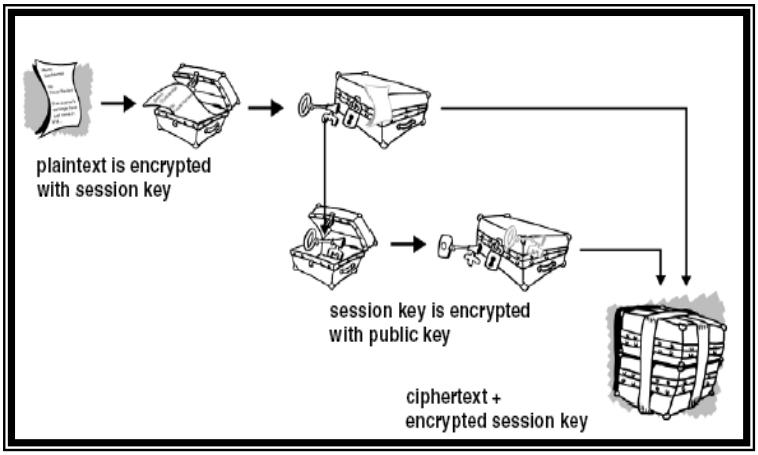


Figure (6): Received Process.

The combination of the two-encryption methods combines the convenience of public-key encryption with the speed of conventional encryption. Conventional encryption is about 10,000 times faster than public-key encryption. Public-key encryption in turn provides a solution to key distribution and data transmission issues. Used together, performance and key distribution are improved without any sacrifice in security[8].

5. Parametric Curves.

5.1 introduction

There are two principal ways to describe the shape of a curved line: **implicitly** and **parametrically**. The implicit form describes a curve by a function f(x, y) that provides a relationship between the x and y coordinates [9].

An alternative way of describing lines and curves which treats the coordinates x and y symmetrically is the parametric form.

The coordinate x and y are expressed as function of an auxiliary parameter t, so that x=x(t), y=y(t). For example, the circle $x^2+y^2-1=0$ can be expressed parametrically by equations:

$x=\cos t$ and $y=\sin t$ --------1

Where t takes, values in the range $0 \le t \le 2\pi$. Although we normally need to prescribe the range of the parameter t. This can be an advantage if we want to describe a segment of a curve. For example, the arc ABC of the circle in figure (7) is completely described by the parametric equations (1) and condition $2\pi/3 \le t \le 7\pi/6$.

The parametric equations enable us to plot points on the curves by evaluating x (t) and y (t) for successive values of t [10].

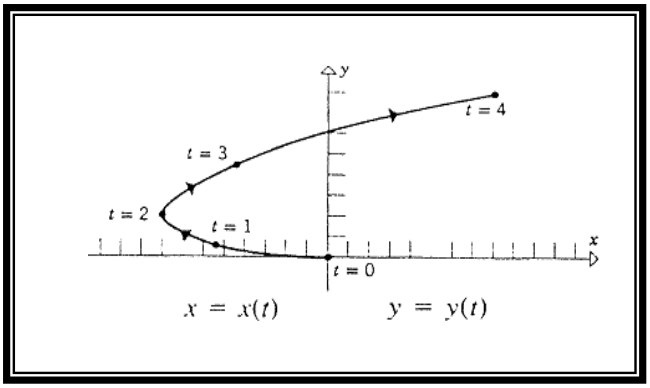


Figure (7): Example to describe the parametric equations.

A parametric form for a curve produces different point on the curve, based on the value of a parameter. A parametric form suggests the movement of a point through time, which we can translate into the motion of a pen as it sweeps out the curve. The path of a particle traveling along the curve is fixed by two functions, x( ) and y( ),and we speak of (x(t),y(t)) as the position of the particle at time t.

The curve itself is the totality of points "visited" by the particle as t varies over some interval. Figure (8) shows a plane curve to be traced by a moving point. If we use the parameter t to denote time, then the parametric equations x=x(t), y=y(t) specify how x- and y-coordinates of the moving point vary with time.

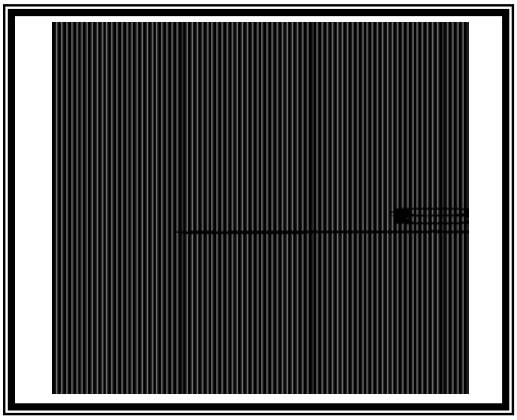


Figure (8): Plane curve.

Parametric form for curve produce different points on the curve, based on the value of parameter.

Parameterization greatly affects on curve shape, and thus adds a security feature that cannot be predicted by any counterfeiter designer. For this reason we proposed to parametric curve form to increase the curve security, in the following the algorithm is to generate *Bezier curve(* i.e one of curves have parametric property)[11,12].

Algorithm: generate Bezier curve.

Input: Given four points; $(x_i, y_i), i = 0,\ldots,3$

Output: Interpolate the get new values for each of x and y to draw.

Process:

Step1: for u= 0 to 1 step 0.01

Step2:let $X = (1-u)^3 x_0 + 3(1-u)^2 u x_1 + 3(1-u)u^2 x_2 + u^3 x_3$

Step3:let $Y = (1-u)^3 y_0 + 3(1-u)^2 u y_1 + 3(1-u)u^2 y_2 + u^3 y_3$ Step4: Plot (x, y)

Step5: Next u

Step6: End.

6. The proposed method to modify the PGP protocol using Random Art technique.

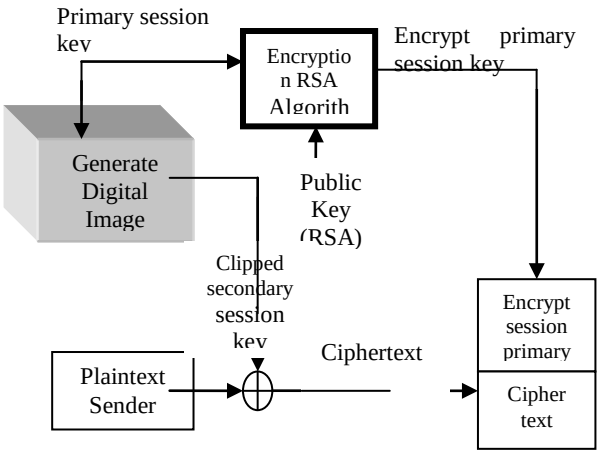
The strong cryptography employed by PGP is the best available today. The PGP protocol is a hybrid cryptosystem that combines some of the best features of both conventional and public-key cryptography. In this section propose insert the random art technique for generated digital images capability to the PGP protocol stages to increase protocol robustness and make the protocol more difficult in front of the counterfeiter.

This paper aim used the generated images facility that generate from random art as a session key instead of generating the session key by using movement of the mouse or the keystrokes type. The clipped session key from the generated image, in this case consist of two session keys, *primary session key* and *secondary session key*.

Primary session key represent the function F for algorithm of random art that refer to it in section (3.1) and secondary session key represent the stream of randomness bits sequence that is clipped according to curve equations for Bezier *curve* to increase the randomness bits sequence. The works with New-PGP begin when a user encrypts plaintext. *First,* compress the plaintext . *Second,* creates a session key select function F to generate images according to the algorithms in section(3.1). *Third,* the user enters primary session key(control points coordinates) to the clip secondary key from digital mage(pixels depending on the coordinate for bezier curve generation). *Forth,* the user XOR the stream of the secondary session key bits sequence with the plaintext after compression process. *Fifth,* the sender uses the public key from RSA algorithm to encrypt the primary session key. *Sixth,* the sender transmits the encrypted primary session key along with the ciphertext to the recipient. Decryption works in the reverse order. The recipient's copy of new-PGP uses his or her private key from RSA algorithm to recover the primary session key that is used to generate the secondary key from generated image to decrypt the conventionally encrypted ciphertext.

Example

To explain how the New-Protocol works, and indicate the protocol behavior. We used the result to generate a image use a primary session key (PSK) that consists of function F with 12 depth . According to the primary session key we clipped a secondary session key (SSK) of size equal to 260 random bits using bezier curve equation for clipping. The public key (PK) of the RSA algorithm consist of (n=997517,e=193) where (secret p =977) and (secret q=1021), and the private key of RSA algorithm equal (d=727297).



New- PGP cryptography protocol (Sending process)

7. Conclusions

From the New-PGP method, reached to the following conclusions:-

a. The process to guesses the primary session key from the secondary key is infeasible, because there is no correlation between the two session keys.
b. If the counterfeiter succeed to solve the factorization problem from RSA , and find the private key from public key , the key that is obtained can not help him to recover the plaintext from primary session key unless knowing the secondary session key.
c. The New-PGP increased secure condition to the PGP protocol that made the protocol more robust and efficient.

7. References

[1] Alfred J.M., Paul V. C. and Scott A. V.,"***HandBook of Applied Cryptography***", Fifth Addition,(2001).

[2] Andrej Bauer." ***Gallery of random ar"t***. WWW at http://www.cs.cmu.edu/~andrej/art/, 1998.

[3] Adrian Perrig." ***Hash Visualization: a New Technique to improve Real-World Security"***

[4] Dhamija Rachna,. "***Hash visualization in user authentication"***. In proceedings of the computer Human Interaction Conference,(2000).

[5] Newman W.M., "***Sproull R.F.,"Principles of Interactive Computer Graphics"***; Mc Graw-Hill Book Company London, 1981.

[6] Akhlas Abbas Al Bahrany,***" StructuredImage Authentication System"***, M.Sc. Thesis, University of technology at Computer Science,(2001***)***.

[7] Schaefer E. D. "An introduction to Gryptography"; Santa Clara University, 1999.

[8] GPG Corporation,"An Introduction of Cryptography***";*** www.pgp.com, 2004.

[9] Hill F. S."***Computer Graphics Using OPENGL***", second edition, Prentice Hall,(2001).

[10] Wiliam J. M. , Robin S. L.. "***The Art of Computer Graphics Programming***", Van Nostrnd Reinhold Company, Inc, (1987).

[11] Jean Gallier, "***Curves and Surfaces in Geometric Modeling",*** Morgan Kaufman Publishers,(2000).

[12] Richardson M "***Modren Computer Graphics***". Blackwell Scientific Publications, .,(1989).