2016

Using graphic methods to challenge cryptographic performance

Brian Cusack
Digital Forensics Research Laboratories, Institute for Radio Astronomy and Space Research, Auckland University of Technology

Erin Chapman
Digital Forensics Research Laboratories, Institute for Radio Astronomy and Space Research, Auckland University of Technology

DOI: 10.4225/75/58a6991e71023
Cusack, B., & Chapman, E. (2016). Using graphic methods to challenge cryptographic performance. In Johnstone, M. (Ed.). (2016). *The Proceedings of 14th Australian Information Security Management Conference, 5-6 December, 2016, Edith Cowan University, Perth, Western Australia.* (pp.30-36).

USING GRAPHIC METHODS TO CHALLENGE CRYPTOGRAPHIC PERFORMANCE

Brian Cusack, Erin Chapman
Digital Forensic Research Laboratories, AUT
brian.cusack@aut.ac.nz, erinchapman@xtra.co.nz

Abstract

Block and stream ciphers have formed the traditional basis for the standardisation of commercial ciphers in the DES, AES, RC4, and so on. More recently alternative graphic methods such as Elliptic Curve Cryptography (ECC) have been adopted for performance gains. In this research we reviewed a range of graphic and non-graphic methods and then designed our own cipher system based on several graphic methods, including Visual Cryptography (VC). We then tested our cipher against RC4 and the AES algorithms for performance and security. The results showed that a graphics based construct may deliver comparable or improved security and performance in many of the required areas. These findings offer potential alternative avenues for post-quantum cryptographic research.

Keywords
Cryptography, performance, security, graphs, testing, word-oriented ciphers

INTRODUCTION

The demand for cryptographic methods has always been strong. The ever-expanding use of technology for communications, banking and financial transactions of diverse types, secure communications, and many other Internet applications is driving current demand for security and performance. The consumers of cryptography products require ever-increasing protection at lower cost (Thakur et al., 2011). Algorithms must maintain the confidentiality of communications, the integrity of the messages, and the accessibility of information to the users. The requirement for privacy of information has become increasingly challenging, caught in inter-jurisdictional debates of legality and the ability of developers to provide the levels of protection required (Bhat et al., 2015). The implementation of cryptographic algorithms in modern networked systems is crucial to ensure the users of information are satisfied with the service they receive. Many standardised algorithms have come and gone as vulnerabilities have been exploited to make algorithms unusable in the current cryptographic climate.
Events such as the theoretical cracking of the data encryption standard (DES), revisions including triple DES, and the major competition that resulted in the adoption of the Advanced Encryption Standard (AES) (Fluhrer et al., 2011; Singhal & Raina, 2011), illustrate the constant evolution of cryptography. While much research has been done to improve the security of traditional ciphers such as the AES and the now-defunct Rivest Cipher 4 (RC4) (Klien, 2008), there are opportunities for the development and improvement of alternative ciphers (Ustimenko, 2007). Our research focused on the potential of graphic methods. Encryption using Visual Cryptography (VC) and Elliptic Curve Cryptography (ECC), is well-established and has been shown to give high levels of security, improved performance, and reduced resource requirements. It also shows that that alternative competitors can be found in graphic schemes. To demonstrate that there are alternative approaches to achieve secure methods for the ever-expanding online world we constructed a word-oriented symmetric stream cipher. It was tested against AES, RC4, ECC, and VC algorithms, and the results demonstrated that alternatives are possible using graphic schemes.

The proposed system was termed Coordinate Matrix Encryption (CME) to reflect the graphic construct behind the algorithm (Galbraith & Menezes, 2005; Hou et al., 2014). It was implemented in Java along with the four competing algorithms, and we tested both graphic-based and traditional cryptographic algorithms against our construct. The algorithms were all tested for security, efficiency and resource consumption. The comparative results show the high levels of security achievable by alternative graphic-based ciphers and the potential for alternative innovations. The resistance of the proposed 8-bit CME system to brute force attacks was shown to be 157,899 orders of magnitude higher than that of a 128-bit key in traditional ciphers such as AES. Examination of the avalanche effect of the CME scheme showed that less than 0.5% of all bytes within the cipher text remained in the same position when a single bit of the plaintext was altered. While the RC4 scheme offered the best efficiency in terms of time required to encrypt and decrypt the data, it has been proven vulnerable; and the CME comparison showed lower memory requirements and faster setup execution. This offers the potential for research, testing and implementation of different approaches to make traditional cryptography adaptable to the new high-speed cyber connected world (Vigila et al., 2009; Tawalbeh et al., 2013).

GRAPHIC METHODS

Graphic-based systems rely on group theory and graph theory to create secure algorithms for encryption. Some of the more popular graphic-based methods are ECC (Akhter, 2015) and VC (Blundo et al., 2006). However, there are other algorithms that take advantage of the innate properties of group theory and families of graphs (Cohn, 2000; Polak et al., 2013). These graphic methods for encryption exploit particular traits of certain types of graphs, such as those using families of graphs of large girth, for example Cayley graphs. A Cayley graph is defined as a graph G(G,S) where S is a non-empty subgroup of the group G, such that S is equal to its own inverse $(S = S^{-1})$, and the set of vertices is equal to G, V=G, and the set of edge elements is:

$$E = \{\{x,y\} : x,y \in G, \exists s \in S : y = xs\}$$

A Cayley graph constructed in this manner is a regular graph, and the underlying algebraic structures of the family of Cayley graphs can be exploited for use in encryption (Priyadarsini, 2015).

Another family of graphs providing a possible route for cryptographic research is the family of directed graphs of large girth. The fact that there are only three families of undirected graphs of arbitrarily large girth limits their use, however there are infinite numbers of algebraically constructed families of such graphs. These can be converted to equivalent Turing machines of basic construction. The basic finite automaton is equitable to a directed graph, if the memory component is ignored. These graphs are part of the expander family of graphs. Cayley graphs can be used to describe a linear automata, while other graph families can be used to result in non-linear systems for encryption. Encryption systems based around groups of graphs such as Cayley or expander families use sequences of vertices or graph-colourings to create a cipher text. Others opt for using strongly regular graphs to generate a Hadamard matrix for encoding images (Davidoff et al., 2003). Some systems use the vertices to represent the plaintext space and the path within the graph becomes the password. Systems such as these based around walks along graph edges can be used in the construction of stream ciphers. Expander graphs are also of interest in cryptography; they are sparse, finite, and highly connected. Ramanujan graphs are a brand of expander graphs that are often used for encryption (Agnarsson & Greenlaw, 2007).

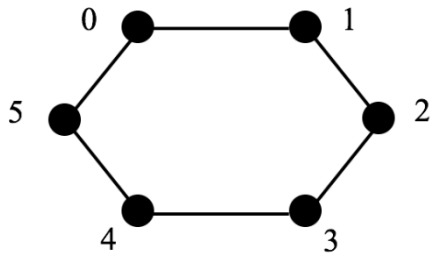

Figure 1. A simple Cayley graph described by G=Z/6Z,S={1,-1}

New systems have been proposed that utilise group theory and rings to create encryption that relies on the combining of two group elements. The reversing of these processes is impossible and hence establishes their cryptographic value. Multivariate cryptography is the set of cryptosystems which use polynomials and finite commutative rings for encryption, and these are part of the post-quantum cryptography movement (Ustimenko & Romanczuk, 2013). The concepts of quantum cryptography have been used and applied to theoretically break many standard cryptographic algorithms including the RSA. Post-quantum cryptography involves systems that are theoretically resistant to adversaries using quantum attacks. Graphic methods have hence been explored as possibilities in the post-quantum world, with the potential to define non-standard cryptographic methods that are better and stronger than the new adversaries. New developments in graphic methods have used the injection of algebraic geometry into the field of multivariate public key cryptosystems with outstanding effect (Blakley & Kabatiansky, 2011). The structures are a set of multivariate quadratic polynomial equations over finite fields and parameterised matrices for systems of para-unitary equations that deliver the cryptographic solution. Multivariate polynomials are a solution to the problems of RSA and an alternative to systems like ECC, using multivariate systems of equations over small fields, such as $GF(2^m)$ where m is some small number. The use of multivariate polynomials is a proposed solution to the issues with key size and set up time, both of which are high in computational complexity and require large amounts of data to communicate. Multivariate systems generally use quadratic polynomial fields. The multivariate systems rely on their own version of the one-way problem, in this case called the MQ problem, based on the computational complexity of solving many different quadratic equations over multiple different fields using many different variables (Chen et al., 2012; Sutter et al., 2013). The complexity of the MQ problem has led to these graphic methods being proposed as a possible quantum-resistant encryption method.

ISSUES WITH GRAPHIC METHODS

Encryption systems that use graphs for encoding, like those based around VC, can have very high computational overhead, due to the size of the graphs required to achieve the required levels of security (Klein & Wessler, 2007). Also, those encryption methods that base themselves around the special colourings of vertices and edges are vulnerable to cubical linearization attacks, which make decryption possible, and are costly in practice. For those graph-based systems that also rely on the DLP, the same vulnerabilities encountered by ECC encryption apply. The biggest problem within graph based systems is implementation challenges. Representing a graph within a computer program can be broken down into four possible objects for management: the adjacency list; the adjacency matrix; the incidence list; and the incidence matrix. Each object lists either vertices or edges, and they are either enumerated fully – in a matrix – or only where a connection occurs. These implementations affect the use of a particular system, especially with larger connected graphs, with many entries in its matrix or list (Riaz & Ali, 2011).
The security of ECC relies on computational complexity to assure that it is intractable to compute the Elliptic Curve Discrete Logarithm Problem (ECDLP) (Yan, 2008). This reliance means that the security would be severely compromised should the ever-increasing speed of technology provide a method of computing the solution to the ECDLP in less than the current exponential time. On the realization of quantum computers, the Elliptic Curve Discrete Logarithm problem will no longer be computationally infeasible to compute and exposing the algorithm to an intractable vulnerability (Kramer, 2015). The weakness surrounding ECC in a post-quantum world is based on Shor's algorithm, operating on a quantum computer, which is capable of solving problems such as discrete logarithms in minimal time. Aside from the possibility of breaking the Discrete Logarithm Problem, ECC also has disadvantages in its implementation. It is highly complex to implement, and the resulting cipher text message is increased in length from the original plaintext significantly (Akhier, 2015).

Advances in fields such as index calculus and number-field sieves have shown possible weaknesses in systems based around the problem of computing discrete logarithms (Joux & Vitse, 2012). Index calculus, as a method of computing discrete logarithms using probability and field arithmetic, which has been used by mathematicians to exploit characteristics of groups and to then solve the original discrete logarithm problem (Agnarsson & Greenlaw, 2007). While classic index calculus has not been implemented successfully against general ECC systems, and exponential time square root attacks are more efficient against these general ECC algorithms, the reduction in computing time for solving the discrete logarithm problem in other systems may suggest weakness in the overall computational complexity of DLP-based systems.

VC schemes encounter difficulties due to pixel expansion, the number of subpixels required to encode the correct level of contrast in each share (Shyu et al., 2007). This expansion greatly affects the required overhead of VC schemes, and as such is the target of much research. While there have been schemes proposed that give a constant pixel expansion, such as graph-based extended VC (Lu et al., 2011), many schemes require linear, or even polynomial pixel expansion based on the number of nodes within the scheme, making them infeasible for larger implementations. Within the schemes which ensure pixel expansion remains constant, the overhead for the encoding of the shares is still computationally high for large images with a greater numbers of pixels. These systems which constrain pixel expansion also degrade the contrast of an image, as there are fewer subpixels differentiating dark and light in the image, making it more difficult for the human eye to visually decode. Once multiple colours are introduced to the scheme, pixel expansion becomes even more complex, and overall image contrast is lowered further (Liu et al., 2008). A colour VC scheme will also require higher overall time complexity, as each colour within the image must have a different threshold for contrast.

VC is also open to malicious man-in-the-middle attacks, during the transfer of shares to participants. If the shares are intercepted, the malicious intermediary could keep the original share, and forward a new, false share to the intended participant. The interception of the share would as such result in the security of the scheme being completely undermined. Attacking a VC scheme in this manner is generally referred to as cheating. While this risk can be decreased by the implementation of a filter where each participant is assigned a specific target image, cheating is still possible, by a malicious participant. Cheating prevention VC schemes have been proposed that use specific basis matrices in the generation of both the secret shares, and a set of verification shares, to counter the ability to generate fake shares. These matrices add an extra column to the original matrices and hence extra cost. The problem still remains however, that these basis matrix schemes are not immune to cheating. To prevent this type of cheating, it is necessary to introduce multiple extra zero columns into the basis matrices that adds further costs. As a result, cheating prevention VC schemes result in higher overheads and increased pixel expansion when compared regular VC algorithms, which delivers a lower level of utility in real-world application. The proposal of adding tags to individual shares to allow for the identification of false or forged shares may offer additional protection against cheating, however it is still vulnerable to attack if an attacker is in possession of a genuine share, and can therefore find and replicate the security tag (Wang & Hsu, 2011).

ALGORITHM DESIGN

The algorithm design for the CME scheme was based around a square coordinate matrix and transformations in a finite Galois field $F(2^n)$ (Martin, 2012; Martinez & Encinas, 2013). The coordinate matrix design was structured by the concepts delivered in error-correcting codes, in which sparse matrices and code words are used to eliminate noise from the transmissions. In addition, security principles from VC were applied. The main encryption process uses a randomized coin toss style procedure, which is similar to the VC method of choosing whether a given pixel is black or white. This coin toss decides if the next section of the cipher text is to be a blank padding section, or if it is the next section of the plaintext message. If it is a blank padding section, one of the locations containing an empty entry is picked at random from a blank list, and the binary or integer coordinates (depending on the implementation) of that location are then input as the next part of the cipher text. Else, if the section is a part of the plaintext message, then a location containing that bit string is randomly chosen from the list of locations for the string. The location is then translated into the corresponding coordinates and concatenated to the cipher text. The scheme involves the addition of exactly the same number of blank coordinates as enciphered message coordinates. As a result of the addition of padding characters, the resulting cipher text is exactly four times the length of the plaintext, with two coordinates for every message or padding character, and exactly the same number of padding and message characters. The style of encryption means that the total length of the outputted cipher text is fixed at exactly four times the length of the plaintext, which may prove to result in undesirable overheads for transmission.

```
Total strings: 8
Number of occupied spaces: 32
Number of blank spaces: 32
Total matrix size: [8,8]
[---][---][101][---][---][---][---][---]
[---][010][011][---][---][---][000][---]
[000][111][---][001][---][101][---][---]
[110][000][001][---][---][---][---][111]
[---][---][---][010][111][100][100][---]
[010][100][001][---][---][100][101][011]
[---][---][101][010][---][110][---][---]
[110][---][011][011][111][000][001][110]
Set up complete, time taken: 19 ms.
Total memory used: 0.43109130859375 MB
```

Figure 2: A randomly generated key matrix for a 3-bit coordinate CME matrix scheme.

The use of multiple locations for each bit string and the addition of an equal number of padding coordinates at random locations in the ciphertext provide resistance to cryptanalysis, and particularly to known and chosen plaintext attacks, as the encryption process therefore results in a non-singular mapping. This non-singular mapping means each plaintext input has many possible ciphertext outputs for any one key matrix. The multiple locations also result in far more of the overall matrix being taken up by bit strings than would be the case if each string appeared only once. Again, this helps prevent cryptanalytic attacks, as it increases the likely occurrence of the same of padding coordinates appearing more than once, which is helpful in further confusing any analysis of the resulting data. A sample of a 16-bit plaintext and the corresponding 64-bit ciphertext resulting from encryption using a 4-bit coordinate matrix scheme is shown in Figure 3.

```
Plaintext:
0101000110111011
Ciphertext:
1110011001001011101101001100010001100111101100111010010000000010
```

Figure 3. Example plaintext ciphertext pair output from a 4-bit CME scheme.

The decryption process uses the same key matrix as in the encryption process and looks up each of the coordinates. If a given coordinate is an empty padding variable, it is discarded. If not, the value of the coordinate is combined with the next character of the key string using exclusive-OR, and the resulting value is added to the plaintext output. In this manner, the extra noise generated by the encryption process to ensure security is efficiently removed during decryption. Because each step of the decryption process consists only of simple entry check and exclusive-OR operation, the overall efficiency for decrypting the ciphertext is theoretically higher than that of the encryption

process.

TESTING AND RESULTS

The algorithm implementations were tested using Java, on an Intel i7 3.1GHz machine with 16GB of RAM. All algorithms were tested for resistance to brute force, avalanche effect, set up requirements and encryption/decryption time. Equation 1 shows the brute force analysis based on the key space for traditional 128-bit keys, while Equation 2 compares the resistance of the CME 8 and 4 bit schemes, based on the key space of the relevant matrix sizes. The results of avalanche effect testing showed that when the 8-bit CME algorithm was trialled against RC4 and AES it not only performed very well but outperformed the traditional algorithms, demonstrating the resilience of the CME algorithm, shown in Table 1. Table 2 shows the results of avalanche testing of 4-bit CME against VC, in which both algorithms achieved the maximum Hamming distance. Table 3 shows the set up time and memory requirements for ECC, 8-bit CME, AES, RC4, and Table 4 shows the set up time and memory for VC and 4-bit CME. Encryption/decryption times are given in Table 6 for AES, RC4, and 8-bit CME, and in Table 6 for 4-bit CME and VC. These results appear to show CME as a potential competitor for streaming encryption.

Equation 1 $$brute\ force_{128-bit} \approx 1.7014118 \times 10^{38}$$

Equation 2 $$brute\ force_{8-bit\ CME} \approx 1.19162785 \times 10^{157937}$$

Table 1: Comparative avalanche effect results with AES and RC4

Data Size (bits)	**Same Bytes (%)**			**Same Position (%)**		
	RC4	AES	CME	RC4	AES	CME
304	97.668	37.767	44.839	97.368	24.779	0.414
928	99.472	62.777	84.026	99.145	38.905	0.388
3024	99.940	87.935	99.713	99.735	45.857	0.422
4408	99.979	94.276	99.984	99.819	48.227	0.404
8144	99.997	99.100	100	99.902	48.593	0.395

Table 2: Comparative avalanche effect results with 4-bit CME and VC

Data Size (bits)	**% of Bits Unchanged**	
	VC	**CME 4-bit**
16	49.275	50.319
32	50.169	50.619
64	50.005	50.499
128	49.934	50.286
256	49.981	50.337
512	50.072	50.120

Table 3: Set up and memory requirements for 8-bit CME, AES, ECC and RC4

	ECC	CME	AES	RC4
Time taken (ms):	359.5	80.13	409	258.5
Memory used (MB):	1.192	1.217	2.364	2.340

Table 4: Set up and memory requirements for 4-bit CME and VC

	4-bit CME	VC
Memory used (MB):	0.448	0.456
Time taken (ms):	21.44	0

Table 5: Encryption/decryption time for AES, RC4 and 8-bit CME

Data Size (bits)	Encryption (ms)			Decryption (ms)		
	AES	Byte CME	RC4	AES	Byte CME	RC4
304	0.199	0.031	0.014	0.17	0.012	0.022
928	0.142	0.059	0.027	0.182	0.023	0.025
3024	0.173	0.136	0.023	0.179	0.065	0.016
4408	0.185	0.173	0.02	0.196	0.05	0.045
8144	0.148	0.262	0.024	0.25	0.093	0.032

Table 6: Encryption/decryption times for 4-bit CME and VC schemes

Bit String Length	Encryption (ms)		Decryption (ms)	
	VC	4-bit CME	VC	4-bit CME
16	0.056	0.02	0.01	0.02
32	0.08	0.066	0.014	0.036
64	0.196	0.104	0.052	0.06
128	0.368	0.214	0.088	0.074
256	0.868	0.369	0.214	0.136
512	2.822	1.13	0.492	0.328

CONCLUSION AND FURTHER RESEARCH

Further research into alternative graphic methods is required to explore the applications of alternative systems such as CME. The security offered by the proposed CME scheme makes it a potential candidate for post-quantum cryptographic research. The system's alternative key structure and non-singular mapping allow for resistance to a large key space and superb avalanche effect, while maintaining competitive efficiency. These features require further exploration. Comparative analysis between traditional and graphic-based ciphers is required to determine whether alternative graphic methods are able to offer higher security for lower overheads. Optimization of these schemes requires further research to ensure a lasting competitive advantage, and suitability for implementation in application development. There is currently little standardisation in stream ciphers to replace RC4, and as such the opportunity exists for an optimized version of CME to assist in this particular space in applications such as TLS that utilize stream ciphers for encryption on a day-to-day basis.

REFERENCES

Agnarsson, G., & Greenlaw, R. (2007). *Graph Theory: Modelling, Applications, and Algorithms*. New Jersey: Pearson Education Ltd.

Akhter, F. (2015). A novel Elliptic Curve Cryptography scheme using random sequence. Paper presented at the 2015 International Conference on Computer and Information Engineering (ICCIE).

Bhat, B., Ali, A. W., & Gupta, A. (2015). DES and AES performance evaluation. Paper presented at the International Conference on Computing, Communication & Automation (ICCCA), 2015.

Blakley, G. R., & Kabatiansky, G. (2011). Secret Sharing Schemes. In H. A. van Tilborg & S. Jajodia (Eds.), *Encyclopedia of Cryptography and Security* (pp. 1095-1097): Springer US.

Blundo, C., Cimato, S., & De Santis, A. (2006). Visual cryptography schemes with optimal pixel expansion. *Theoretical Computer Science*, 369(1–3), 169-182.

Chen, Y. C., Horng, G., & Tsai, D. S. (2012). Comment on Cheating Prevention in Visual Cryptography. *IEEE Transactions on Image Processing*, 21(7), 3319-3323.

Cohn, P. M. (2000). *Introduction to ring theory*. Springer Science & Business Media.

Davidoff, G., Sarnak, P., & Valette, A. (2003). *Elementary number theory, group theory and Ramanujan graphs* (Vol. 55): Cambridge University Press.

Fluhrer, S., Mantin, I., & Shamir, A. (2011). Weaknesses in the key scheduling algorithm of RC4. Paper presented at the International Workshop on Selected Areas in Cryptography.

Galbraith, S., & Menezes, A. (2005). Algebraic curves and cryptography. *Finite Fields and Their Applications*, 11(3), 544-577.

Hou, Y. C., Wei, S. C., & Lin, C. Y. (2014). Random-Grid-Based Visual Cryptography Schemes. *IEEE Transactions on Circuits and Systems for Video Technology*, 24(5), 733-744.

Joux, A., & Vitse, V. (2012). Cover and decomposition index calculus on elliptic curves made practical *Advances in Cryptology–EUROCRYPT 2012* (pp. 9-26): Springer.

Klein, A. (2008). Attacks on the RC4 stream cipher. *Designs, Codes and Cryptography*, 48(3), 269-286.

Krämer, J. (2015). *Why Cryptography Should Not Rely on Physical Attack Complexity*. Singapore: Springer.

Liu, F., Wu, C. K., & Lin, X. J. (2008). Colour visual cryptography schemes. *Information Security, IET, 2*(4), 151-165.

Lu, S., Manchala, D., & Ostrovsky, R. (2011). Visual cryptography on graphs. *J. Comb. Optim., 21*(1), 47-66.

Martin, K. M. (2012). *Everyday Cryptography: Fundamental Principles & Applications*. New York: Oxford University Press.

Martinez, V. G., & Encinas, L. H. (2013). Implementing ECC with Java Standard Edition 7. *International Journal of Computer Science and Artificial Intelligence*, 3(4), 134.

Polak, M., Romańczuk, U., Ustimenko, V., & Wróblewska, A. (2013). On the applications of Extremal Graph Theory to Coding Theory and Cryptography. *Electronic Notes in Discrete Mathematics*, 43, 329-342.

Priyadarsini, P. L. K. (2015). A Survey on some Applications of Graph Theory in Cryptography. *Journal of Discrete Mathematical Sciences and Cryptography*, 18(3), 209-217.

Riaz, F., & Ali, K. M. (2011, 26-28 July 2011). *Applications of Graph Theory in Computer Science.* Paper presented at the Third International Conference on Computational Intelligence, Communication Systems and Networks (CICSyN), 2011.

Shyu, S. J., Huang, S.-Y., Lee, Y.-K., Wang, R.-Z., & Chen, K. (2007). Sharing multiple secrets in visual cryptography. *Pattern Recognition*, 40(12), 3633-3651.

Singhal, N., & Raina, J. (2011). Comparative analysis of AES and RC4 algorithms for better utilization. International *Journal of Computer Trends and Technology*, 2(6), 177-181.

Sutter, G. D., Deschamps, J. P., & Imana, J. L. (2013). Efficient Elliptic Curve Point Multiplication Using Digit-Serial Binary Field Operations. *IEEE Transactions on Industrial Electronics*, 60(1), 217-225.

Tawalbeh, L., Mowafi, M., & Aljoby, W. (2013). Use of elliptic curve cryptography for multimedia encryption. *IET Information Security*, 7(2), 67-74.

Thakur, J., & Kumar, N. (2011). DES, AES and Blowfish: Symmetric key cryptography algorithms simulation based performance analysis. *International Journal of Emerging Technology and Advanced Engineering*, 1(2), 6-12.

Ustimenko, V. (2007). On Graph-Based Cryptography and Symbolic Computations. *Serdica Journal of Computing*, 1(2), 131-156.

Ustimenko, V., & Romańczuk, U. (2013). *On extremal graph theory, explicit algebraic constructions of extremal graphs and corresponding Turing encryption machines Artificial Intelligence*, *Evolutionary Computing and Metaheuristics* (pp. 257-285). Heidelberg, Germany: Springer.

Vigila, S., & Muneeswaran, K. (2009). Implementation of text based cryptosystem using Elliptic Curve Cryptography. Paper presented at the First International Conference on Advanced Computing, 2009. ICAC 2009.

Wang, R. Z., & Hsu, S. F. (2011). Tagged Visual Cryptography. *IEEE Signal Processing Letters*, 18(11), 627-630

Yan, S. Y. (2008). *Cryptanalytic attacks on RSA*. New York, USA: Springer US.