Advances in Economics, Business and Management Research, volume 215
Proceedings of the 2022 7th International Conference on Social Sciences and Economic Development (ICSSED 2022)

Research on the Security of Elliptic Curve Cryptography

Jiaxu Bao

Queen Mary University of London, London, UK, E1 4NS
jiaxubao@gmail.com

ABSTRACT
Elliptic curve cryptography has the characteristics of high-security strength and low computational complexity. Elliptic curve cryptography relies on point multiplication, which is the most time-consuming part of the encryption and decryption process. The Elliptic Curve Cryptosystem is currently the most famous and potential public key cryptosystem. It is proposed based on the computational difficulty of discrete logarithms on the elliptic curve, and its security research is an important research area in academia. This paper analyzes the security of elliptic cryptographic curves from the performance comparison of ECC and RSA. Moreover, this paper implements RSA and ECC using random private keys, and the sample data input is 64-bit, 8-bit, and 256-bit. Experiments are done on MATLAB R2008a on an Intel Pentium dual-core processor. The findings reveal that RSA is efficient at encryption, but sluggish at decryption, whereas ECC is slow at encryption but efficient at decryption. Overall, ECC outperforms RSA in terms of efficiency and security. ECC surpasses RSA in terms of operational security and efficiency, according to this research.

Keywords: *Elliptic Curve Cryptography, Security of Elliptic Curve Cryptography, RSA, ECC*

1. INTRODUCTION

According to Afreen and Mehrotra [1], the changing global landscape shows an elegant fusion of communications and computing, where computers using wired communications are rapidly being replaced by small handheld embedded computers using wireless communications in almost all fields; this increases data security and privacy requirements. Elliptic curve cryptography is one of the most promising public key cryptosystems. Due to the advantages of elliptic curve cryptography over other public key cryptosystems in terms of security, implementation efficiency and implementation cost, it has been widely used and has been adopted as the standard of public key cryptography algorithm by many countries and international standard organizations [2]; thus, its security problem has naturally attracted extensive attention from scholars. Although discussion on the security of public key cryptography mechanisms is very lively, there are limited research achievements in terms of the security of ECC mechanisms. Because of the particularity and application of cryptography, it is necessary to discuss the security of cryptography.

In academics, security analysis of two practical and popular asymmetric algorithms RSA and ECC is carried out. RSA is considered to be the first generation of public key cryptography and has been very popular since its inception, while ECC has also grown in popularity recently. The integer factorization problem (IFP) is used to secure the RSA cryptosystem, whereas the elliptic curve discrete logarithm problem (ECDLP) is used to secure the ECC cryptosystem [3]. The major advantage of ECC over RSA is because solving ECDLP with the best-known technique takes completely exponential time, whereas solving IFP with RSA takes sub-exponential time. This implies that ECC may employ many fewer parameters than RSA while maintaining the same level of security. This paper analyzes the security of elliptic cryptographic curves from the performance comparison of ECC (Elliptic Curve Cryptography) and RSA (Rivest Shamir Adleman).

2. LITERATURE REVIEW

Several authors have performed RSA and ECC's security analyses using different measurement parameters. Gula et al. (2004) compares elliptic curve point multiplication on RSA and ECC based on two 8-bit processor computer systems, and it discovered that ECC-160-point multiplication was more efficient than RSA-1024 private key operation on both systems [2]. In the work of Bosch et al. [3], the scholars evaluate the risk of key usage in terms of key length for RSA and ECC, they report that until 2014, using 1024-bit RSA provided some small risk, while using 160-bit ECC on prime fields was probably safe longer usage time. Moreover, some scholars also concluded that RSA is faster, but ECC is better than RSA in terms of security [4]. Jansma et al. [5] compared the usage of digital signatures in RSA and ECC, and suggested that RSA might be a good choice for applications that need to verify messages rather than generate signatures. Another researcher suggests that RSA is currently stronger than ECC, although they also suggest that ECC will outperform RSA in the future [6]. Mahto et al. [7] proposed the use of ECC over insecure channels to enhance the security of 64-bit one-time password (OTP) data communications.

Moreover, [10] compared elliptic curve dot multiplication operations on ECC and RSA on two 8-bit processor computer systems, and they found that ECC-160 dot multiplication was more efficient than RSA-1024 private key operations on both systems. [11] proposed key length-based risk assessment for RSA and ECC keys, and they agree that until 2014, using 1024-bit RSA provided some small risk, while 160-bit ECC on prime fields Can be used safely for longer periods of time. [12] concluded that RSA is faster, but ECC is better than RSA in terms of security. [13] compared the usage of digital signatures in RSA and ECC and suggested that RSA might be an application where it is necessary to verify messages rather than generate signatures. [14] shows that currently, RSA is more powerful than ECC, however, in the near future, ECC may outperform RSA. [15] demonstrate that ECC is superior to RSA in terms of operational efficiency and security.

3. PERFORMANCE COMPARATION OF ECC WITH RSA

3.1. Steps of Elliptic Curve Algorithm

3.1.1. Encryption Algorithm

(1) encodes the message to generate a number m, and selects a point $P(x,y)$ on the ellipse domain;

(2) selects a random number k, and calculates the point $P_1 = (x_1, y_1) = kC$;

(3) calculates $P_2 = (x_2, y_2) = kPb$ according to $B's$ public key Pb;

(4) calculates the ciphertext $C = mx + y$;

(5) sends $C' = \{kG, P + kPb, C\}$ to B.

3.1.2. Decryption algorithm

(1) uses his own private key nb to calculate: $P + kPb - nb(kG) = P$

(2) calculates $m = (C - y)/x$, and obtains the plaintext m.

3.2. ECC and RSA Performance Comparation

Most products that use public key cryptography for encryption and digital signatures are based on the RSA algorithm, but with the progress and perfection of the method of factoring large integers, the number of digits in the password has been increasing to ensure its security. It is generally believed that passwords with a word length of more than 1024 bits are safe. This is a heavy burden for applications using RSA.Compared with cryptographic systems such as RSA, ECC has the characteristics of high speed, small space occupation, less computation, and increased security.

3.2.1. Better security and small amount of calculation

The security of ECC is influenced by the difficulty of determining k from kP and P, which involves the logarithm issue of elliptic curves. Currently, Pollard's rho method is one of the fastest ways to solve the logarithm of an elliptic curve. The table below compares this method with the general number field sieve method for factoring large integers. As can be seen from the below table, the keys used by ECC are much shorter than those used in RSA. When the keys are the same, ECC and RSA require similar computations. Therefore, compared with the same security RSA, ECC needs less computation than RSA because the key used by ECC is shorter. If it uses Pollard's rho method to find the logarithm of an elliptic curve, the table 1 can be achieved.

Table 1. using Pollard's rho method to find the logarithm of an elliptic curve

Key Size	MIPS year
150	3.8×10^{10}
205	7.1×10^{18}
234	1.6×10^{28}

Table 2. Integer factorization using general number field sieves

Key Size	MIPS year
512	3×10^{4}
768	2×10^{8}
1024	3×10^{11}
1280	1×10^{14}
1536	3×10^{16}
2048	3×10^{20}

In addition, the exponential integration method can be used to attack the discrete logarithm problem on finite field, and its computational complexity is: $O[exp\sqrt[3]{(logp)(loglogp)^2}]$, where p is the modulus (prime). But this approach doesn't work well for discrete logarithm problems on elliptic curves. The current method for attacking discrete logarithm issues on elliptic curves is a suitable method for attacking discrete logarithm problems on cyclic group, and its computational complexity is: $O[exp(log\sqrt{p_{max}})]$. In which, p_{max} is the largest prime factor of the order of Abel group formed by the elliptic curve. Therefore, elliptic curve cryptosystems are more secure than public key systems over finite fields.

3.2.2. Fast processing speed and take up little space

Although in RSA, its processing speed of public key can be improved by selecting a smaller public key, and speed of encryption and signature verification can be improved, making it comparable to ECC in terms of speed for signature verification. But in the processing speed of private keys, such as signature and decryption, ECC is much faster than RSA. Therefore, in the same situation, ECC has better encryption performance compared with RSA.

Moreover, compared with RSA, ECC has a smaller password length and system parameters, but its security strength is satisfactory. That is, it occupies much less storage space than RSA, which is very important for the application of encryption algorithms in IC card. In summary, the elliptic curve encryption system has obvious advantages compared with the RSA encryption algorithm.

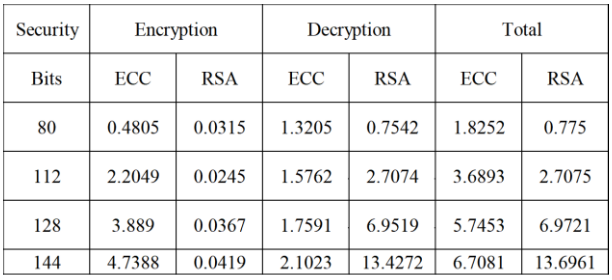
4. ANALYSIS/DISCUSSION

The point multiplication operation of ECC cryptosystem is the most time-consuming part of entire encryption and decryption process, which needs to further optimization during operation. Moreover, a general-purpose processor can only process 64-bit data at most with a conventional instruction, which is inefficient for ECC calculation. In FPGA/ASIC, theoretically, each clock cycle can process data of any word length, which is relatively low compared to general-purpose processors. The processor has higher computing efficiency and speed. In addition, FPGA/ASIC can be customized and optimized according to the needs of application, and parallelization technology, such as pipeline and ping-pong operation, can be used to further accelerate the calculation process and obtain higher performance. Therefore, FPGA/ASIC is a better choice to realize ECC computing in occasions with high computing performance requirements or embedded real-time applications.

4.1. The SECURITY ANALYSIS of ECC and RSA

This paper implements RSA and ECC using random private keys according to the recommendations of [8] [9], and the sample data input is 64-bit, 8-bit, and 256-bit. Experiments are done on MATLAB R2008a on an Intel Pentium dual-core processor (533 MHz, 1.60 GHz, 1 MB L2 cache) and 2GB DDR2 RAM under the Ms-Windows platform. The efficiency of ECC over RSA is shown in the table below. The findings reveal that RSA is fast at encryption but sluggish at decryption, whereas ECC is slow at encryption but fast at decryption. Overall, ECC outperforms RSA in terms of performance and security. In terms of operational security and efficiency, this research shows that ECC surpasses RSA.

Table 3. 8 BITS DECRYPTION, ENCRYPTION, AND TOTAL TIME



Security	Encryption		Decryption		Total	
Bits	ECC	RSA	ECC	RSA	ECC	RSA
80	0.4805	0.0315	1.3205	0.7542	1.8252	0.775
112	2.2049	0.0245	1.5762	2.7074	3.6893	2.7075
128	3.889	0.0367	1.7591	6.9519	5.7453	6.9721
144	4.7388	0.0419	2.1023	13.4272	6.7081	13.6961

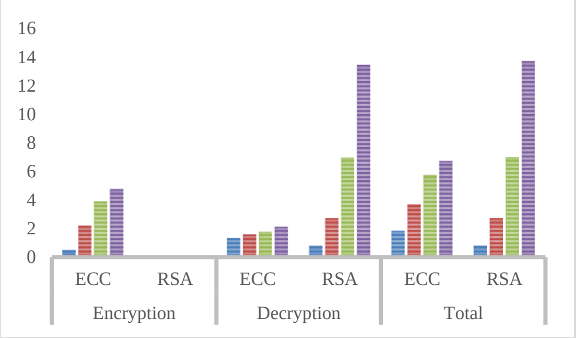
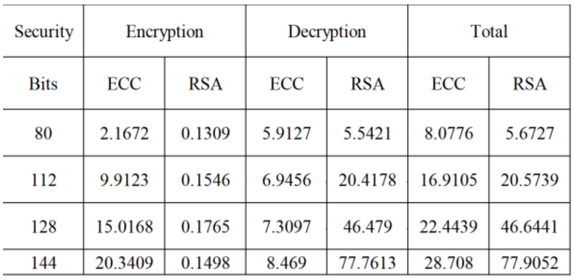


Figure 1. 8 BITS DECRYPTION, ENCRYPTION, AND TOTAL TIME

Table 4. 64 BITS DECRYPTION, ENCRYPTION, AND TOTAL TIME



Security	Encryption		Decryption		Total	
Bits	ECC	RSA	ECC	RSA	ECC	RSA
80	2.1672	0.1309	5.9127	5.5421	8.0776	5.6727
112	9.9123	0.1546	6.9456	20.4178	16.9105	20.5739
128	15.0168	0.1765	7.3097	46.479	22.4439	46.6441
144	20.3409	0.1498	8.469	77.7613	28.708	77.9052

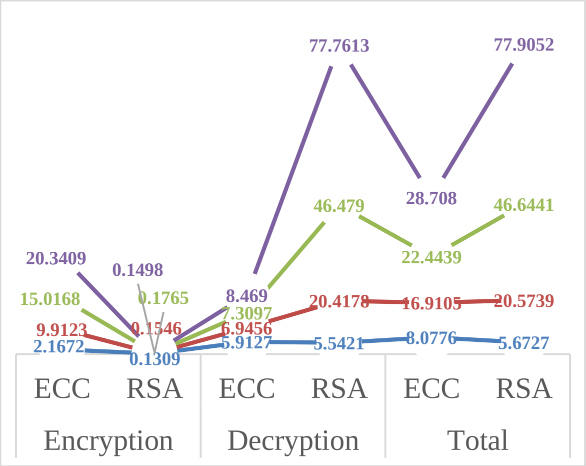


Figure 2. 64 BITS DECRYPTION, ENCRYPTION, AND TOTAL TIME

Table 5. 256 BITS DECRYPTION, ENCRYPTION, AND TOTAL TIME

Security	Encryption		Decryption		Total	
Bits	ECC	RSA	ECC	RSA	ECC	RSA
80	7.91	0.54	22.87	19.32	30.81	19.86
112	39.65	0.57	26.35	102.02	66.02	102.62
128	58.21	0.59	27.41	209.61	85.85	210.16
144	77.51	0.58	32.14	311.05	109.67	311.62

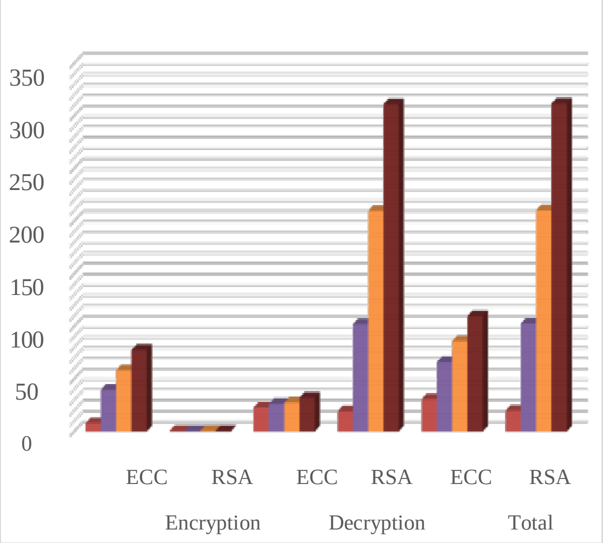


Figure 3. 256 BITS DECRYPTION, ENCRYPTION, AND TOTAL TIME

5. CONCLUSION

This paper analyzes RSA and ECC's security strength for 3 samples of input data based on NIST-recommended 64-bit, 8-bit, 256-bit random keys. This paper implements RSA and ECC using random private keys, and the sample data input is 64-bit, 8-bit, and 256- bit. Experiments are done on MATLAB R2008a on an Intel Pentium dual-core processor. The results show that RSA is efficient at encryption but slow at decryption, and ECC is slow at encryption but efficient at decryption. Overall ECC is more efficient and secure than RSA. This work indicates that ECC outperforms RSA in terms of operational security and efficiency. The result of the paper is also limited by the random keys and the processor, and it is not accurate enough. Future work can focus on overcoming the limitations of existing algorithms.

ACKNOWLEDGMENT

I want to thank my parents for supporting indirectly or directly in this research paper.

REFERENCES

[1] Afreen, R., & Mehrotra, S. C. (2011). A review on elliptic curve cryptography for embedded systems. arXiv preprint arXiv:1107.3631.

[2] Gura, N., Patel, A., Wander, A., Eberle, H., & Shantz, S. C. (2004, August). Comparing elliptic curve cryptography and RSA on 8-bit CPUs. In International workshop on cryptographic hardware and embedded systems (pp. 119- 132). Springer, Berlin, Heidelberg.

[3] Bos, J., Kaihara, M., Kleinjung, T., Lenstra, A. K., & Montgomery, P. L. (2009). On the Security of 1024-bit RSA and 160-bit Elliptic Curve Cryptography (No. REP_WORK).

[4] Kute, V. B., Paradhi, P. R., & Bamnote, G. R. (2009). A software comparison of rsa and ecc. Int. J. Comput. Sci. Appl, 2(1), 43-59.

[5] Jansma, N., & Arrendondo, B. (2004). Performance comparison of elliptic curve and rsa digital signatures. nicj. net/files.

[6] Alese, B. K., Philemon, E. D., & Falaki, S. O. (2012). Comparative analysis of public-key encryption schemes. International Journal of Engineering and Technology, 2(9), 1552- 1568.

[7] Mahto, D., & Yadav, D. K. (2015, February). Enhancing security of one-time password using elliptic curve cryptography with biometrics for e-commerce applications. In Proceedings of the 2015 Third International Conference on Computer, Communication, Control and Information Technology (C3IT) (pp. 1-6). IEEE.

[8] Mahto, D., & Yadav, D. K. (2015, February). Enhancing security of one-time password using elliptic curve cryptography with biometrics for e-commerce applications. In Proceedings of the

2015 Third International Conference on Computer, Communication, Control and Information Technology (C3IT) (pp. 1-6). IEEE.

[9] Nagar, A., Mohapatra, D. P., & Chaki, N. (Eds.). (2015). Proceedings of 3rd International Conference on Advanced Computing, Networking and Informatics: ICACNI 2015, Volume 1 (Vol. 43). Springer.

[10] Shim, K. A. (2015). A survey of public-key cryptographic primitives in wireless sensor networks. IEEE Communications Surveys & Tutorials, 18(1), 577-601.

[11] Deng, L., Huang, H., & Qu, Y. (2017). Identity Based Proxy Signature from RSA without Pairings. Int. J. Netw. Secur., 19(2), 229-235.

[12] Diffie, W., & Hellman, M. E. (1976). " New Directions in Cryptography" IEEE Transactions on Information Theory, v. IT-22, n. 6.

[13] Dong, X. (2015). A multi-secret sharing scheme based on the CRT and RSA. International Journal of Electronics and Information Engineering, 2(1), 47-51.

[14] Gura, N., Patel, A., Wander, A., Eberle, H., & Shantz, S. C. (2004, August). Comparing elliptic curve cryptography and RSA on 8-bit CPUs. In International workshop on cryptographic hardware and embedded systems (pp. 119- 132). Springer Berlin, Heidelberg.

[15] Han, L., Xie, Q., & Liu, W. (2017). An Improved Biometric Based Authentication Scheme with User Anonymity Using Elliptic Curve Cryptosystem. Int. J. Netw. Secur., 19(3), 469-478.