World Applied Sciences Journal 29 (Computer Sciences, Engineering and Its Applications): 150-154, 2014
ISSN 1818-4952
© IDOSI Publications, 2014
DOI: 10.5829/idosi.wasj.2014.29.csea.2246

Highly Secured ECC and RSA Based Crypto Processor

K.P. Thooyamani, R. Udayakumar and V. Khanaa

School of Computing Science, Bharath University, Chennai-73, India

Abstract: With rapid increase in communication and network applications, cryptography has become a crucial issue to ensure the security of transmitted data. In this paper, I propose a microcode-based architecture with a novel reconfigurable data path which can perform either prime field GF (p) operations for arbitrary prime numbers, irreducible polynomials and precision. Using these field arithmetic units, users are capable of programming cryptographic algorithms in microcode sequences for full compliance with a majority of public-key cryptographic algorithms such as RIVEST-SHAMIR-ADLEMAN (RSA) and elliptical curve cryptosystems. An algorithmic optimization flexibility, high hardware utilization and high performance.

Key words: Public Key Cryptography · Private key cryptography · Elliptical curve cryptography · Rivest-Shamir-Adleman

INTRODUCTION

It is the science and art of creating the secret codes. Here we will use some encryption and decryption techniques using the private and public keys to provide authentication. It consists of both the active and passive attacks.

Review: Does increased security provide comfort to paranoid people? Or does security provide some very basic protections that we are naive to believe that we don't need? During this time when the Internet provides essential communication between tens of millions of people and is being increasingly used as a tool for commerce, security becomes a tremendously important issue to deal with.

There are many aspects to security and many applications, ranging from secure commerce and payments to private communications and protecting passwords. One essential aspect for secure communications is that of cryptography, which the focus of this chapter is. But it is important to note that while cryptography is necessary for secure communications, it is not by itself sufficient. The reader is advised, then, that the topics covered in this chapter only describe the first of many steps necessary for better security in any number of situations.

Major Purpose:

- The first is to define some of the terms and concepts behind basic cryptographic methods and to offer a way to compare the myriad cryptographic schemes in use today.
- The second is to provide some real examples of cryptography in use today.

The Purpose of Cryptography: Cryptography is the science of writing in secret code and is an ancient art; the first documented use of cryptography in writing dates back to circa 1900 B.C. when an Egyptian scribe used non-standard hieroglyphs in an inscription. Some experts argue that cryptography appeared spontaneously sometime after writing was invented, with applications ranging from diplomatic missives to war-time battle plans. It is no surprise, then, that new forms of cryptography came soon after the widespread development of computer communications. In data and telecommunications, cryptography is necessary when communicating over any untrusted medium, which includes just about any network, particularly the Internet.

Within the context of any application-to-application communication, there are some specific security requirements, including:

Corresponding Author: R. Udayakumar, School of Computing Science, Bharath University, Chennai-73, India.

Authentication: The process of proving one's identity. (The primary forms of host-to-host authentication on the Internet today are name-based or address-based, both of which are notoriously weak.)

Privacy/confidentiality: Ensuring that no one can read the message except the intended receiver.

Integrity: Assuring the receiver that the received message has not been altered in any way from the original.

Non-repudiation: A mechanism to prove that the sender really sent this message.

Module Description: The cryptographic system is designed using the following three main modules.

- RSA Description Module
- ECC Description Module
- Parallel Processing

RSA Description Module:

RSA: The first and still most common, PKC implementation, named for the three MIT mathematicians who developed it-Ronald Rivest, Adi Shamir and Leonard Adleman. RSA today is used in hundreds of software products and can be used for key exchange, digital signatures, or encryption of small blocks of data. Unlike Diffie-Hellman, RSA can be used for key exchange as well as digital signatures and the encryption of small blocks of data. Today, RSA is primary used to encrypt the session key used for secret key encryption (message integrity) or the message's hash value (digital signature). RSA's mathematical hardness comes from the ease in calculating large numbers and the difficulty in finding the prime factors of those large numbers. Although employed with numbers using hundreds of digits, the math behind RSA is relatively straight-forward.

ECC Description Module :

Elliptic Curve Cryptography (ECC): A PKC algorithm based upon elliptic curves. ECC can offer levels of security with small keys comparable to RSA and other PKC methods. It was designed for devices with limited compute power and/or memory, such as smartcards and PDAs.

Elliptic curves in public key cryptography was independently proposed by Koblitz and Miller in 1985 [1] and since then, an enormous amount of work has been done on elliptic curve cryptography. The attractiveness of using elliptic curves arises from the fact that similar lever of security can be achieved with considerably shorter keys than in methods based on the difficulties of solving discrete logarithms over integers or integer factorizations.

Parallel Processing: In the public key cryptographic systems, RSA/ECC are the most significant and peculiar characteristics compared to that of the any other algorithmic methods. RSA is the popular cryptography but ECC become more and more attractive. So a coprocessor which can meet the requirements of both *GF@j* and *GF(2'7* fields will be of great use for the future of the cryptography. This type coprocessor is not only compatible with the existing RSA system hut also suitable for the promising ECC system. In order to send the information from source to destination, RSA has to perform some partial amount of bit streams and it has to be encrypted by RSA algorithm, in the receiver section it has to be decrypted by the same algorithm. In the similar way the rest of the information has to be encrypted by ECC algorithm, in the receiver section, it has to be decrypted by the same algorithm. It doesn't take the data simultaneously ie., some higher bit streams has to be processed by RSA, or some lower bit streams has to be processed but ECC, viceversa..So that, it's highly impossible to break the information by the third person. We can easily recover the data from the middle man attacks; the required data are more secured, confidential in nature.

Proposel System: The Explosive growth in data communications and internet services has made cryptography an important research topic. Cryptography is used for confidentiality, authentication, data integrity and non-repudiation, which can be divided into two families: secret-key cryptography and public-key cryptography. Secret-key cryptography [1-4], which usually has a relatively compact architecture and smaller key size than public-key cryptography, is often used to encrypt/ decrypt sensitive information or documents. Public- key cryptography [3-10] offers fundamental technology for key agreement, encryption/decryption (two keys) and digital signatures. For example, the IEEE Standard 1363 [5] specifies public-key cryptographic techniques, including cryptographic schemes and its underlying mathematic operations based on number-theoretic hard problems.

In contrast to secret-key cryptography, public-key cryptography usually has a lower throughput rate and more complicated computation. Note that the actual throughput rate and area cost depends on the specific algorithm and its implementation schemes plus the requirements of target applications.

Kinds of Tasks: Basically, there are two kinds' tasks for public-key cryptosystem: one is controlling part, which is in charge of the cryptography protocols flow; the other is computation part, which is used to accelerate the big integer modular multiplication or point multiplication over elliptic curve. So our cryptosystem consists of a MCU (main controlling unit) and a coprocessor. In our design, MCU is a general 32-bit RlSC Embedded CPU, which will deal with the arithmetic and the protocol. While mprocessor does modular multiplication and point multiplications, which is computation intensive. Communications between the two parts can be through the coprocessor interface of the MCU. In this paper, only the RSA/ECC coprocessor is concerned.

Implementation of PKC: Implementing public-key cryptosystems on a general-purpose processor (GPP) is flexible because a variety of cryptosystems can be used at runtime. A drawback of a GPP realization is that it generally results in a lower throughput rate and larger power consumption. Considerable effort has been directed toward a fast realization of cryptography algorithms consisting of very large integer operands (up to 4096 bits).

For real-time applications, a dedicated hardware implementation is required to speedup the computation of cryptosystems. In particular, an application-specific integrated circuit (ASIC) solution generally leads to a higher throughput rate at a lower cost, but it is inflexible; therefore, it is only applied to a limited subset of cryptosystems.

To mitigate the gap between GPP and ASIC realization, application-specific cryptographic processors have been proposed [11-22], each with its own instruction-set architecture, data path design and target applications.

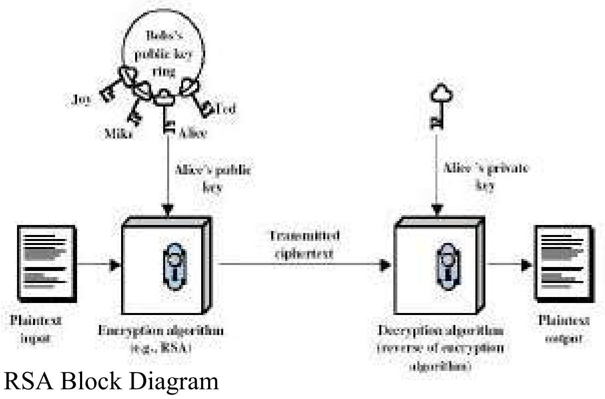
The RSA cryptosystem [6] and ECC [9, 10] are the two most popular public-key cryptosystems for cryptographic applications. One way of Zealizing RSA and ECC is to develop some related instructions and field arithmetic to carry out their underlying operations, particularly for those used in the modular exponentiation algorithm for RSA and the scalar multiplication algorithm for ECC. The field arithmetic includes operations defined in finite (Galois) fields, namely.

Advantages of RSA/ECC: It is easy principle and convenience to use. But with the improvement of computer computing power and methods of decomposing big integer, the secret key of RSA becomes longer and longer in order to maintain the security level. And the hardware implementation becomes more and more complex. Comparison with RSA system, the merits such as shorter secret key, smaller area of memory and highlevelsecurity make the elliptic curve ayptography (ECC) easier to be implemented through hardware. This characteristic will introduce ECC to the most attractive cryptosystem.

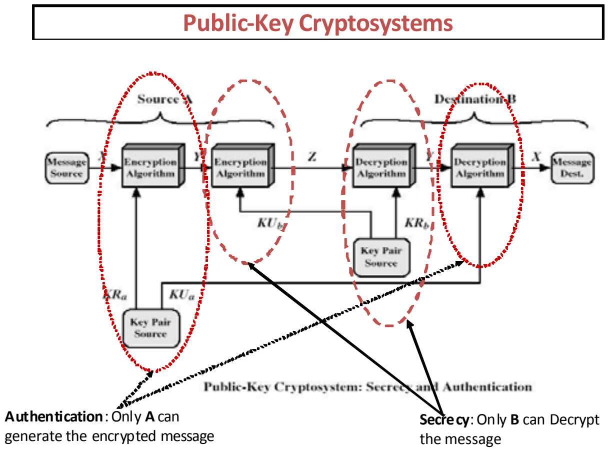
Algorithmically, the RSA public key system appears to be much stronger and scalable than the private key alternatives of DES, RC4 and RC2. The fact that the US government has licensed the RC4 and RC2 algorithms attests to their weakness, given current US legislation and protocols regarding the export of cryptographic software and systems. There is no such thing as a free lunch, however and for the stronger RSA security there is a processing cost that needs to be paid as state above, RSA is the popular cryptograph but ECC become more and more attractive. So a coprocessor which can meet the requirements of both GF@j and GF (2'7 fields will be of great use for the future of the cryptography. This type coprocessor is not only compatible with the existing RSA system hut also suitable for the promising ECC system.

In this paper, a new hardware reconfigurable public-key cryptography coprocessor is introduced, which can perform the complex arithmetic both of ECC and RSA, i.e., the coprocessor can carry out both point multiplication over elliptic maandmand multiplicationo fbig integers.

System Design of RSA/ECC: Basically, there are two kinds' tasks for public-key cryptosystem: one is controlling part, which is in charge of the cryptography protocols flow; the other is computation part, which is used to accelerate the big integer modular multiplication or point multiplication over elliptic curve. So our cryptosystem consists of a MCU (main controlling unit) and a coprocessor. In our design, MCU is a general 32-bit RlSC Embedded CPU, which will deal with the arithmetic and the protocol. While microprocessor does modular multiplication and point multiplications, which is computation intensive. Communications between the



RSA Block Diagram



ECC Block Diagram

two parts can be through the coprocessor interface of the MCU. In this paper, only the RSA/ECC coprocessor is concerned.

Some Comparisons: Some comparisons about system frequency and baud rate among different designs of the 512-bit RSA cryptosystem. Our RSA/ECC coprocessor bas area of 45K gates and about 100MHz system frequency and can achieve 190 Kbps encryption rate for 512-hit RSA and 50K bps encryption rate for 233-bit ECC. We can see that the proposed RSA/ECC has a comparative performance and chip complexity; furthermore, it can perform both RSA and ECC cryptographies. The scalability is also an attractive characteristic.

CONCLUSION

A high-performance cryptographic processor that supports both the prime field and binary extension field operations using a reconfigurable data path was presented. This data path successfully merged modular multiplication squaring and division/multiplication into single reconfigurable computing cells with limited multiplexers. The processor can handle field arithmetic for cryptography algorithms by using microcode sequences without modifying hardware. Cryptography chip is important to ensure information security in this paper, a new scalable and unified coprocessor for both RSA and ECC based on Montgomery algorithm is proposed. The coprocessor can perform the modular multiplication of RSA, point tnultiblication of ECC over *GF@)* and *GF (2")*. It can implement modular multiplication with the hit length fiom 32 to 512 and point multiplication of ECC with the bit length born 32 to 256 without any modification Experimental results show that the developed cryptographic processor exhibits obvious speed and performance advantages in comparison with related works and can accommodate a large number of cryptosystem applications.

REFERENCES

1. McIvor, C., M. McLoone and J.V. McCanny, 2004. “Modified Montgomery modular multiplication and RSA exponentiation techniques,” IEEE Proc. Comp. Digit. Tech., 151(6): 402-408.
2. Shieh, M.D., J.H. Chen, H.S. Wu and W.C. Lin, 2008. “A new modular exponentiation architecture for efficient design of RSA cryptosystem,” IEEE Trans. Very Large Scale Integr. (VLSI) Syst., 16(9): 1151-1161.
3. Lin, W.C., M.D. Shieh, J.H. Chen, C.H. Wu and H.S. Wu, 2007. “Acombined multiplication/division algorithm for cost-effective design of elliptic curve Cryptosystem” in Proc. IEEE Region 10 Conf. TENCON, pp: 1-4.
4. Chen, J.H., H.S. Wu, M.D. Shieh and W.C. Lin, 2007. “A new Montgomery modular multiplication algorithm and its VLSI design for RSA cryptosystem,” in Proc. IEEE Int. Symp. Circuits Syst., pp: 3780-3783.
5. Wu, C.H., C.M. Wu, M.D. Shieh and Y.T. Hwang, 2004. “High-speed, low-complexity systolic designs of novel iterative division algorithms”, IEEE Trans. Compute, 53(3): 375-380.

6.3 Chelton, W.N. and M. Benaissa, 2008 . “Fast elliptic curve cryptography on FPGA,” IEEE Trans. Very Large Scale Integr. (VLSI) Syst., 16(2): 198-205.

7. Großschadl, J. and E. Savas, 2004. “Instruction set extensions for fast arithmetic in finite fields,” in Proc. Cryptographic Hardw. Embedded Syst. (CHES), pp: 133-147.

8. Mentens, N., K. Sakiyama, L. Batina, B. Preneel and I. Verbauwhede, 2007. “A side-channel attack resistant programmable pkc coprocessor for embedded applications,” in Proc. IEEE Int.Symp.Syst.,Arch., Model.Simulation, pp: 194-200.
9. Smyth, N., M. McLoone and J.V. McCanny, 2006. “An adaptable and scalable asymmetric cryptographic processor,” in Proc. IEEE Int. Conf. Appl.-Specific Syst. Arch. Processors, pp: 341-346.
10. Leong, P.H.W. and I.K.H. Leung, 2002. “A microcoded elliptic curve processor using FPGA technology,” IEEE Trans. Very Large Scale Integr.(VLSI) Syst., 10(10): 550-559.
11. Satoh, A. and K. Takano, 2003. “A scalable dual-field elliptic curve cryptographic processor,” IEEE Trans. Comput., 52(4): 449-460.
12. Wang, Y., J. Leiwo and T. Srikanthan, 2005. “A unified architecture forcrypto-processing in embedded systems,” in Proc. IEEE Conf. Embedded Softw. Syst., pp: 1-7.
13. Zeng, X., C. Chen and Q. Zhang, 2004. reconfigurable public-key cryptography coprocessor,” in Proc. IEEE Asia Pacific Conf. Adv. Syst. Integr Circuits, pp: 172-175.
14. Sakiyama, K., L. Batina, B. Preneel and I. Verbauwhede, 2007. “Multicore curve-based cryptoprocessor with reconfigurable modular arithmetic logic units,” IEEE Trans. Comput., 56(9): 1269-1282.
15. Eslami, Y., A. Sheikholeslami, P.G. Gulak, S. Masui and K. Mukaida, 2006. “An area-efficient universal cryptography processor for smart cards,”IEEE Trans. Very Large Scale Integr. (VLSI) Syst., 14(1): 43-56.
16. Tencac, A.F. and K. Koc, 2003. “A scalable architecture for modular multiplication based on Montgomery’s algorithm,” IEEE Trans. Comput., 52(9): 1215-1221.
17. Wu, C.H., C.M. Wu, M.D. Shieh and Y.T. Hwang, 2004. “High-speed, low-complexity systolic designs of novel iterative division algorithms” IEEE Trans.Compute., 53(3): 375-380.
18. Savas, E., A.F. Tenca and C.K. Koc, 2000. “A scalable and unified multiplier architecture for finite fields” Proc. Cryptographic Hardware Embedded Syst. (CHES), pp: 277-292.
19. IEEE Standard Specifications for Public-Key Cryptography, IEEE Std 1363-2000, 2001.
20. Yan, Z. and D.V. Sarwate, 2003. “New systolic architectures for inversion and division ”IEEE Trans. Computers, 52(11): 1514-1519.