82

Elliptic Curve Cryptography: Delivering High-Performance Security for E-Commerce and Communications

Paul Lambert

Elliptic curve cryptography (ECC) provides the highest strength per key bit of any known public-key security technology. The relative strength advantage of ECC means that it can offer the same level of cryptographic security as other algorithms using a much smaller key. ECC's shorter key lengths result in smaller system parameters, smaller public-key certificates, and, when implemented properly, faster performance with lower power requirements and smaller hardware processors. As a result, ECC is able to meet the security and performance demands of virtually any application.

With the increased amount of sensitive information being transmitted wirelessly and over the Internet, information security has become a critical component to many applications. Cryptography in turn has become a fundamental part of the solution for secure applications and devices. Across a variety of platforms, cryptographic technology provides security to a wide range of applications such as electronic commerce, access control, and secure wireless communications. The ongoing challenge for manufacturers, systems integrators, and service providers is to incorporate efficient, cost-effective security into the mobile, high-performance devices and applications that the market demands. While other cryptographic algorithms cannot effectively meet this challenge, ECC's strength and performance advantages make it an ideal solution to secure Internet commerce, smart card, and wireless applications, as will be demonstrated further on in this chapter.

AU7495—Chapter82—25/1/2007—20:53—PARTHIBAN—14742—XML MODEL CRC12a – pp. 1059–1066.

82.1 Understanding the Strong, Compact Security of ECC

All public-key cryptosystems are based on a hard one-way mathematical problem. ECC is able to deliver strong security at smaller key sizes than other public-key cryptographic systems because of the difficulty of the hard problem upon which it is based. ECC is one of three different types of cryptographic systems that are considered to provide adequate security, defined in standards and deployed in today's applications. Rather than explaining the complete mathematical operation of each of these three systems, this chapter will serve to introduce and compare each system.

First, what is meant by a hard or difficult mathematical problem? A mathematical problem is difficult if the fastest known algorithm to solve the problem takes a long time relative to the input size. To analyze how long an algorithm takes, computer scientists introduced the notion of *polynomial time* algorithms and *exponential time* algorithms. Roughly speaking, a polynomial time algorithm runs quickly relative to the size of its input, and an exponential time algorithm runs slowly relative to the size of its input. Therefore, easy problems have polynomial time algorithms, and difficult problems have exponential time algorithms.

The phrase *relative to the input size* is fundamental in the definition of polynomial and exponential time algorithms. All problems are straightforward to solve if the input size is very small, but cryptographers are interested in how much harder a problem gets as the size of the input grows. Thus, when looking for a mathematical problem on which to base a public-key cryptographic system, cryptographers seek one that cannot be solved in less than exponential time because the fastest known algorithm takes exponential time. Generally, the longer it takes to compute the best algorithm for a problem, the more secure is a public-key cryptosystem based on that problem.

What follows is a discussion of the three different types of cryptographic systems along with an explanation of the hard mathematical problems on which they are based.

82.1.1 RSA and the Integer Factorization Problem

The best-known cryptosystem based on the integer factorization problem, *RSA*, is named after its inventors, Ron Rivest, Adi Shamir, and Len Adleman. Another example is the Rabin–Williams system. The core concept of the integer factorization problem is that an integer p (a whole number) is a *prime number* if it is divisible only by 1 and p itself. When an integer n is the product of two large primes, to determine what these two factors are we need to find the prime numbers p and q such that: $p \times q = n$. The integer factorization problem, then, is to determine the prime factors of a large number.

82.1.2 DSA and the Discrete Logarithm Problem

The Diffie–Hellman key agreement scheme, the grandfather of all public-key cryptography schemes, is based on the discrete log problem. Taher Elgamal first proposed the first public-key cryptographic system that included digital signatures based on this problem. Elgamal proposed two distinct systems: one for encryption and one for digital signatures. In 1991, Claus Schnorr developed a more efficient variant of Elgamal's digital signature system. The U.S. Government's Digital Signature Algorithm (DSA), the best-known of a large number of systems with security based on the discrete logarithm problem, is based on Elgamal's work. The *discrete logarithm problem* modulo prime p is defined in terms of modular arithmetic. This problem starts with a prime number p. Then, given an integer g (between 0 and $p-1$) and a multiplicand y (the result of exponentiating g), the following relationship exists between g and y for some x: $y = g^x \pmod{p}$. The discrete logarithm problem is to determine the integer x for a given pair g and y: Find x so that $g^x = y \pmod{p}$. Like the integer factorization problem, no efficient algorithm is known to solve the discrete logarithm problem.

82.1.3 ECC and the Elliptic Curve Discrete Logarithm Problem

The Security of ECC rests on the difficulty of the elliptic curve discrete logarithm problem. As with the integer factorization problem and the discrete logarithm problem, no efficient algorithm is known to solve the elliptic curve discrete logarithm problem. In fact one of the advantages of ECC is that the elliptic curve discrete logarithm problem is believed to be more difficult than either the integer factorization problem or the generalized discrete logarithm problem. For this reason, ECC is the strongest public-key cryptographic system known today.

In 1985, mathematicians Neil Koblitz and Victor Miller independently proposed the *elliptic curve cryptosystem*, with security resting on the discrete logarithm problem *over the points on an elliptic curve*. Before explaining the hard problem, a brief introduction to elliptic curves is needed.

An *elliptic curve* defined modulo a prime p, is the set of solutions (x,y) to the equation: $y^2 = x^3 + ax + b \pmod{p}$ for the two numbers a and b. This means that y^2 has the remainder $x^3 + ax + b$ when divided by p. If (x,y) satisfies the above equation, then $p = (x,y)$ is a *point* on the elliptic curve.

An elliptic curve can also be defined over the finite field consisting of 2^m (even numbers) elements. This field, referred to as F_2^m, increases the efficiency of ECC operation in some environments. One can define the addition of two points on the elliptic curve. If P and Q are both points on the curve, then $P + Q$ is always another point on the curve. The elliptic curve discrete logarithm problem starts with selecting a field (a set of elements) and an elliptic curve. (Selecting an elliptic curve consists of selecting values for a and b in the equation $y^2 = x^3 + ax + b$.) Then xP represents the point P added to itself x times.

Suppose Q is a multiple of P, so that $Q = xP$ for some x. The elliptic curve discrete logarithm problem is to determine x with any given P and Q.

82.2 A Comparison of Cryptographic Systems

Of the three problems, the integer factorization problem and the discrete logarithm problem both can be solved by general algorithms that run in *subexponential time*, meaning that the problem is still considered hard but not as hard as those problems that admit only fully exponential time algorithms. On the other hand, the best general algorithm for the elliptic curve discrete logarithm problem is fully exponential time. This means that the elliptic curve discrete logarithm problem is currently considered more difficult than either the integer factorization problem or the discrete logarithm problem.

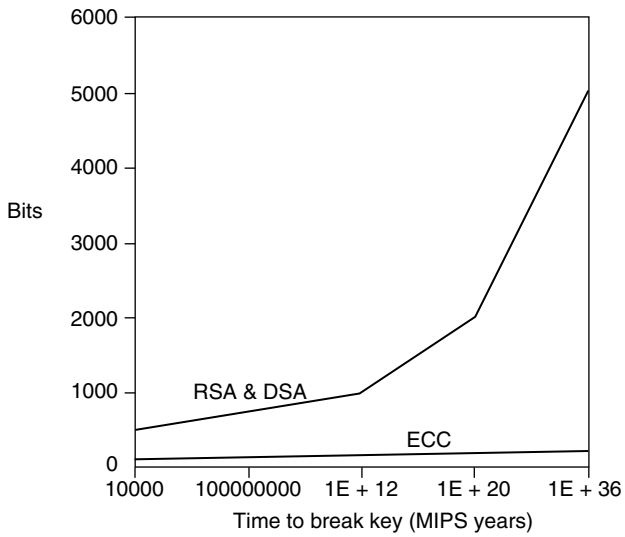

EXHIBIT 82.1 Comparison of security levels.

In Exhibit 82.1, the graph compares the time required to break ECC with the time required to break RSA or DSA for various key sizes using the best-known algorithm. The values are computed in *MIPS years*. A MIPS year represents the computing time of 1 year on a machine capable of performing 1 million instructions per second. As a benchmark, it is generally accepted that 10^{12} MIPS years represents reasonable security at this time, as this would require most of the computing power on the planet to work for a considerable amount of time. To achieve reasonable security, RSA and DSA need to use a 1024-bit key, while a 160-bit key is sufficient for ECC. The graph in Exhibit 82.1 shows that the gap between the systems grows as the key size increases. For example, note how the ratio increases with the 300-bit ECC key compared with the 2000-bit RSA and DSA keys. With this background in ECC's high security relative to small key size, we can explore how ECC benefits today's leading-edge applications.

82.3 Securing Electronic Transactions on the Internet

One prominent application that requires strong security is electronic payment on the Internet. When making Internet-based credit card purchases, users want to know that their credit card information is protected, while the merchant wants assurance that the person making the purchase cannot later refute the transaction. Combined with these authentication needs, a secure electronic payment system must operate fast enough to handle consumer's needs conveniently. It must be capable of handling a high volume of transactions reliably and, simultaneously, be accessible from multiple locations, and be easy to use. ECC can meet all these needs. For example, consider the role ECC plays in securing a recently launched experimental pilot for Internet commerce. The pilot is based on the Secure Electronics Transaction (SET) specification developed to address the requirements of the participants in these Internet transactions.

The SET specification is administered by an organization known as Secure Electronic Transaction LLC (SETCo) formed by Visa and MasterCard. The initial specification provided a complex security protocol using RSA for the public-key components. Because the release of the SET 1.0 specification, implementations of the protocol have been increasing worldwide along with the growing consumer confidence in electronic commerce. Vendors and financial institutions have proposed a number of enhancements to the protocol to further its appeal.

In an ongoing effort to explore ways to improve the SET specification, an experimental pilot program was launched in July 1998 that ran until September 1998. A consortium of players joined together to implemented some exciting leading-edge technologies for use with the SET protocol including ECC, chip cards, and PCI cryptographic hardware. During the pilot, up to 200 selected participants received a smart card, which was a Zions Bank MasterCard with an embedded microprocessor, along with a SET software wallet and a Litronics card reader. These participants shopped at the U.S. Department of Treasury's Bureau of Engraving and Printing Website and were assured that their transactions were protected.

82.3.1 Pilot Operation

1. Cardholder has certificate request and receipt.
2. Cardholder visits Web site at www.bep.treas.gov, selects goods, and initiates payment.
3. Certificates and digital certificates are exchanged.
4. Purchase order and digital signatures are sent via the Internet to the MasterCard payment gateway. Both parties are authenticated; data is decrypted and reformatted.
5. The data is sent via leased lines to Global Payment Systems (GPS) in Atlanta.
6. GPS sends reformed credit card information and purchase data over MasterCard's private BankNet leased line network to Zions Bank.
7. Zions debits cardholder account and issues payment to the Bureau's account via its acquiring bank, Mellon Bank.

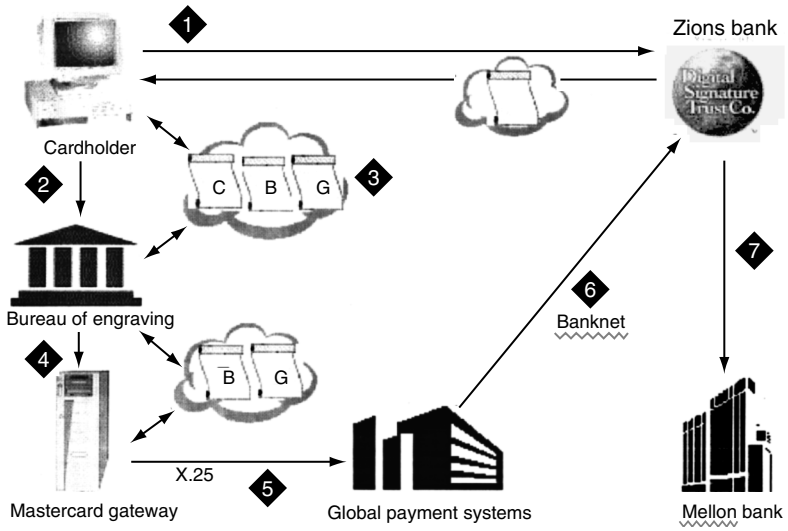

EXHIBIT 82.2 Experimental SET™ pilot.

As represented by Exhibit 82.2, upon receiving the card and reader, the cardholder applies online for a digital certificate with the ECC smart-card-enabled GlobeSet Wallet through Digital Signature Trust Company (DST). DST issues certificates on behalf of Zions Bank using GlobeSet's ECC-enabled CA. The public key is securely sent to DST where a certificate is created and sent back to the cardholder via the Internet. The certificate is stored on the smart card for future use.

82.3.1.1 Procedure

The shopper visits the Bureau's Website at www.bep.treas.gov and selects an item to purchase with his or her Zions Bank MasterCard. The ECC-enabled GlobeSet POS (point of sale) submits a SET wake-up message to the wallet, and the cardholder initiates a transaction by inserting his or her card in to the Litronics reader. All sensitive communication between the two parties is encrypted for privacy and the data is digitally signed for integrity and nonrepudiation according to the SET specification. The purchase order and accompanying information are sent via the Internet through the merchant to the ECC-enabled GlobeSet payment gateway at MasterCard, also employing certificates, signatures, and encryption. The gateway decrypts the data, authenticates both parties, and reformats the data. The data is sent over MasterCard's private BankNet leased-line network to receive payment authorization from Zions Bank, which debits the cardholder's MasterCard account and issues payment to the Bureau through its acquiring bank, Mellon Bank. Cardholders receive their merchandise via the U.S. Postal Service in the usual manner. Implemented end-to-end within an algorithm coexistent system, ECC is an enabling technology adding performance and cost advantages to SET as demonstrated in this pilot.

82.3.1.2 Improving Performance

A comprehensive benchmarking process comparing the performance of ECC and RSA was completed at GlobeSet and audited by a team from SETCo. Improved performance is especially desirable for banks and vendors because cryptographic processing is frequently a bottleneck that can be cleared only with increased hardware costs. In preliminary software-only benchmark tests, ECC demonstrated a positive and significant performance advantage, with overall cryptographic processing overhead reduced by 73 percent. ECC is around 40 times faster than RSA on the payment gateway, which is the SET component more prone to bottlenecks. Signing alone is more than 100 times faster with ECC on this component.

82.3.1.3 Increasing Cardholder Security

Smart cards offer a higher level of security than software-only-based digital wallets because a user's private key and certificate can be stored on the card. As a cryptographic hardware token, smart cards provide stronger user authentication and nonrepudiation than software. Their use translates into lower risk and less fraud for banks, merchants, and consumers.

82.3.1.4 Reducing the Cost of Smart Card Deployment

Smart cards (Exhibit 82.3) are small, portable, tamper-resistant devices providing users with convenient storage and processing capability. As a result, smart cards have been proposed for use in a wide variety of applications such as electronic commerce, identification, and healthcare. For many of these proposed applications, cryptographic security is essential. This requirement is complicated by the fact that smart cards need to be inexpensive in order to be practical for widespread use. The problem is not how to implement cryptography on a smart card but how to do so efficiently and cost-effectively. The smart card is amenable to cryptographic implementations for several reasons. The card contains many security features that enable the protection of sensitive cryptographic data, providing a secure environment for processing. The protection of the private key is critical; to provide cryptographic services, this key must never be revealed. The smart card protects the private key and many consider the smart card to be an ideal cryptographic token; however, implementing public-key cryptography in a smart card application poses numerous challenges. Smart cards present a combination of implementation constraints that other platforms do not: Constrained memory and limited computing power are two of them. The majority of the smart cards on the market today have between 128 and 1024 bytes of RAM, 1 and 16 kb of EEPROM, and 6 and 16 kb of ROM with the traditional 8-bit CPU typically clocked at a mere 3.57 MHz. Any addition to memory or processing capacity increases that cost of each card because both are extremely cost sensitive. Smart cards are also slow transmitters, so to achieve acceptable application speeds data elements must be small (to limit the amount of data passed between the card and the terminal). While cryptographic services that are efficient in memory usage and processing power are needed to contain costs, reductions in transmission times are also needed to enhance usability.

82.3.2 Use of EEC in Smart Cards

Elliptic curve cryptography is ideally suited for implementations in smart cards for a number of reasons:

- *Less memory and shorter transmission times*—The strength (difficulty) of the elliptic curve discrete logarithm problem algorithm means that strong security is achievable with proportionately smaller key and certificate sizes. The smaller key size in turn means that less memory is required to

EXHIBIT 82.3 The smart card.

store keys and certificates and that less data must be passed between the card and the application, so transmission times are shorter.

- *Scalability*—As smart card applications require stronger and stronger security (with longer keys), ECC can continue to provide the security with proportionately fewer additional system resources. This means that with ECC smart cards are capable of providing higher levels of security without increasing their costs.
- *No coprocessor*—The reduced processing times of ECC also make it ideal for the smart card platform. Other public-key systems involve so much computation that a dedicated hardware device, known as a *crypto coprocessor*, is required. The crypto coprocessors not only take up precious space on the card, but they also increase the cost of the chip by about 20 to 30 percent, which translates to an increase of about $3 to $5 on the cost of each card. With ECC, the algorithm can be implemented in available ROM, so no additional hardware is required to perform strong, fast security functions.
- *On-card key generation*—As mentioned earlier, the private key in a public key pair must be kept secret. To truly prevent a transaction from being refuted, the private key must be completely inaccessible to all parties except the entity to which it belongs. In applications using the other types of public key systems currently in use, cards are personalized (keys are either loaded or injected into the cards) in a secure environment to meet this requirement. Because of the complexity of the computation required, generating keys on the card is inefficient and typically impractical.

With ECC, the time needed to generate a key pair is so short that even a device with a very limited computing power of a smart card can generate the key pair, provided a good random number generator is available. This means that the card personalization process can be streamlined for applications in which nonrepudiation is important.

82.4 Extending the Desktop to Wireless Devices

Wireless consumers want access to many applications that previously have only been available from the desktop or wired world. In response to the growing demand for new wireless data services, Version 1.0 of the Wireless Application Protocol (WAP) provides secure Internet access and other advanced services to digital cellular phones and a variety of other digital wireless devices. The new specification enables manufactures, network operators, content providers, and applications developers to offer compatible products and secure services that work across different types of digital devices and networks. Wireless devices are not unlike smart cards in that they also introduce many security implementation challenges. The devices themselves must be small enough to have the portability that users demand. More importantly, the bandwidth must be substantially reduced. The WAP Forum, the organization that developed the WAP specification, has responded to these market and technology challenges by incorporating ECC into the WAP security layer (Wireless Transport Layer Security, or WTLS) specification. With ECC, the same type of sensitive Web-based electronic commerce applications (such as banking and stock trades) that are currently confined to the fixed, wired world can run securely on resource-constrained wireless devices. Strong and efficient security that requires minimal bandwidth, power consumption, and code space is uniquely achievable with ECC. ECC meets the stringent security requirements of the market by incorporating elliptic curve-based Diffie–Hellman key management and the elliptic curve digital signature algorithm (ECDSA) into a complete public-based security system.

Exhibit 82.4 and Exhibit 82.5 compare the signature size and encrypted message size for each of the three cryptosystems discussed earlier. The reduced digital signature and encrypted message sizes result in huge savings of bandwidth, a critical resource in the wireless environment.

EXHIBIT 82.4 Signature Size for a 2000-Bit Message

System Type	Signature Size (bits)	Key Size (bits)
RSA	1024	1024
DSA	320	1024
ECDSA	320	160

EXHIBIT 82.5 Size of Encrypted 100-Bit Message

System Type	Encrypted Message (bits)	Key Size (bits)
RSA	1024	1024
EIGamal	2048	1024
ECES	321	160

82.5 Conclusions

Three types of public-key cryptographic systems are available to developers and implementers today: integer factorization system, discrete logarithm systems, and elliptic curve discrete logarithm systems. Each of these systems can provide confidentially, authentication, data integrity, and nonrepudiation. Of the three public-key systems, ECC offers significant advantages that are all derived (directly or indirectly) from to its superior strength per bit. These efficiencies are especially advantageous in thin-client applications in which computational power, bandwidth, or storage space is limited. The advantages and resulting benefits of ECC for a wide range of applications are well recognized by many in the industry. ECC is being incorporated by a growing number of international standards organizations into general cryptographic standards such as IEEE and ANSI and is being considered for integration into vertical market standards for telecommunications, electronic commerce, and the Internet. Meanwhile, an increasing number of computing and communications manufacturers are building ECC technology into their products to secure a variety of applications for corporate enterprise, the financial community, government agencies, and end users alike. ECC technology has earned its reputation as a truly enabling technology by making many of these products and applications possible by providing viable security.