E-비즈니스 환경과 암호화 기법

강 석 호* · 이 우 기**

《目 次》

요약

본 연구에서는 일반적인 암호화개념을 E비즈니스 환경과 접목하여 적용 및 분류하였다. 암호화(Encryption) 이슈는 크게 보아 비밀키 암호시스템과 공개키 암호시스템, 그리고 공개키-비밀키의 조합방식으로 분류된다. 공개키 암호화시스템은 보안성이 뛰어나지만 계산속도와 효율성 측면에서 난점이 있고, 비밀키 암호시스템은 송·수신자가 동일한 키에 의해 암호화 또는 복호화를 하는 것이며 이는 공개키 알고리즘에 비해 알고리즘이 간단하므로 속도가 빠르고, 소프트웨어로 구현 시 파일의 크기가 작으며, 하드웨어로 구현하는 경우 회로가 간단해지는 경제적인 이유로 널리 이용되고 있으나, 키의 관리에 약점이 있다. 공개키-비밀키의 조합방식은 암호화와 복호화 과정에서 서로 같은 키를 사용하고, 그 키를 암호화하여 키의 전송 및 비밀 보관이 불필요하게 하여 양자 간의 장점을 딴 이중암호화 방식이다. 공개키만의 적용은 대상이 한정적이며 계산에 막대한 노력이 수반되나 보안성이 아주 뛰어난 B2B 등에 적합하고, 비밀키만을 적용하는 데에는 C2C가 정도일 것이며, 공개키-비밀키의 조합은 전자상거래에 가장 보편적인 B2C, G2C, 및 G2B 등에 부합된다고 보았다.

주제어: 암호화(Encryption), 복호화(Decryption), AES(Advanced Encryption Standard), E비즈니스 보안

* 서울대학교 산업공학과
** 인하대학교 산업공학전공

Ⅰ. 서 론

비밀키 암호 시스템(private key encryption system)은 암호화(encryption)하고 복호화(decryption) 하는데 필요한 Key 값이 같은 알고리즘으로 대표적으로는 DES(Data Encryption Standard)와 AES(Advanced Encryption Standard)가 있다. DES는 미국 상무부 표준국(NBS:현재는 미국 국립 표준·기술 연구소(NIST)로 개편)이 1977년 IBM 사의 제안을 바탕으로 제정한 데이터 암호화 표준 규격으로, 연방 정부의 연방 정보 처리 표준 46(FIPS publication 46)으로 채택된 것이다[8, 9]. 그리고 AES는 DES의 표준 기한이 만료되는 1998년을 기점으로 NIST에서는 향후 정부와 상업계에서 사용할 수 있는 강한 비밀키 암호화 알고리즘인 미국 차세대 암호 표준(AES)을 공모한 것으로 보안, 비용, 알고리즘 및 수행특성의 3 부분을 평가해서 채택했다. 수많은 암호 알고리즘이 공모되었고, 5개의 우수항목이 선정되었었으며, NIST는 Rijndael을 최종 선택하여 AES로 지정하였다[1, 2].

공개키 암호 시스템(public key encryption system)은 미국 스탠퍼드 대학의 헬만(M.H. Hellman) 등이 개발한 암호 시스템으로, 데이터의 암호화(encryption)에는 공개키(public key)가 사용되고 복호화(decryption)에는 비밀키(private key)가 사용되는 암호 시스템으로 비밀키 암호 시스템의 정보 교환 당사자간의 암호 전달 문제를 해결한 것으로, 키 값의 안전을 꾀할 수 있고, 디지털 서명 등에 용이하게 쓰일 수 있다. 대표적으로 RSA 공개키 암호방식(RSA public key crytosystem)이 있으며 RSA는 1978년에 MIT 공과대학의 Rivest, Shamir, Adelman 등 3인이 공동 개발한 RSA법(RSA scheme)이라는 암호화 알고리즘을 사용하는 공개 키 암호 방식으로 암호화 조작은 용이하고 복호화에는 방대한 조작이 필요하지만 어떤 복호화 키가 주어지면 용이하게 역변환이 가능하게 되는 일방향성 돌파구(trap door) 함수의 개념. 즉, 큰 수의 소인수 분해에는 많은 시간이 소요되지만 소인수 분해의 결과를 알면 원래의 수는 곱셈에 의해 간단히 구해지는 사실에 바탕을 두고 있다[7].

본 연구를 간략히 일별하면 거래내역이 모두 전자부호(electronic code)로 처리되는 전자상거래에서는 이러한 암호화 문제가 핵심적인 과제임을 인식하고 상호관계와 해결대안 모색에 주안점을 두고 있다는 것이다. 논문의 구성은 다음과 같다. 우선 제2장에서는 연구배경으로서 비밀키와 공개키방식 등을 살펴보고, 제3장에서 제 암호화 방식들의 비교를 한 후, 제4장에서 전자상거래 관점에서 암호화방식의 분류하고 양자 간의 관계를 설정하여 비교하면서 결론을 맺는다.

Ⅱ. 키 기반 보안

2.1 비밀키와 공개키

1) 비밀키(private key cryptography)

암호화하고 복호화 하는데 필요한 Key 값이 똑같은 알고리즘으로 공개키 알고리즘보다는 빠르다. 주로 NSA(National Security Agency)에 의해 개발되었으며 기밀로 취급되고 있으며 80-bit의 key를 사용하는 것이 대부분이며, 다음과 같은 것들이 있다[10, 12]. ROT13(Ceasar의 암호에서 유래한 방법으로 각각의 알파벳을 그 알파벳으로부터 13번째 알파벳으로 대체하는 방식으로, key를 사용하지 않으며, 전혀 안전하지 않음), crypt(독일의 Enigma encryption machine을 모델로 하여 만든 UNIX의 암호화방식으로 전혀 안전하지 않음), DES(Data Encrypt Standard), RC2, RC4 및 RC5(Ronald Rivest에 의해 개발되었으며, 가변길이의 key와 data block크기 등으로 암호화 및 반복이 가능하다), IDEA(International Data Encryption Algorithm), skipjack (NSA에 의해 개발되었으며 기밀로 취급), SEAL (빠른 소프트웨어 구현을 목표로 설계된 스트림암호이며 160비트 키를 사용한다. 소프트웨어적인 대용량의 암/복호화에 적당한 알고리즘이나 인터넷 응용에서는 별로 사용되지 않는다) 등이 있다. BLOWFISH의 경우 DES와 같은 전형적인 Feistel 구조이며, 키가 자주 바뀌는 응용에서는 Key setup time이 길어 불리하며, CAST128는 Feistel 구조이며, 키 길이는 40비트부터 128비트까지 8비트 단위로 지원하며 키 길이가 40비트 이상 80비트 이하일 때는 12라운드를, 88비트 이상 128비트 이하일 때는 16라운드를 사용하도록 권고하고 있다. SAFER의 경우 64비트 블럭길이와 64비트 키길이를 지원하는 알고리즘으로 Key schedule 알고리즘을 확장하여 128비트 키도 사용 가능하다. Knudsen이 보다 강화된 Key scheduling을 제안하여 현재는 이 버전이 주로 사용되며 64비트 키를 사용할 때는 8라운드를, 128비트 키를 사용할 때는 10라운드를 사용하도록 권고되고 있다. 이상과 같은 DES기반 암호화방식보다 진보된 AES에 기반한 비밀키 방식으로는 다음과 같은 것들이 있다[1, 2, 13].

- Rijndael: Daemen과 Rijmen에 의해 개발되었고 2000년 10월 AES 알고리즘으로 최종 선정되었다. SPN (Substitution-Permutation Network) 구조의 가변 블럭길이를 지원하는 블록암호이다. 지원 블럭길이는 128, 192, 256비트이며, 각 블럭길이에

대해 128, 192 혹은 256비트의 키를 사용할 수 있으며 라운드 수는 키 길이에 의해 결정되며, 128비트 블럭을 사용하는 경우 128, 192, 256비트 키에 대해 각각 10, 12, 14라운드를 사용하도록 권고되고 있다.

- RC6: Rivest와 RSA사가 공동 개발한 AES 2차 후보 알고리즘으로 가변길이의 블럭길이, 키 길이 및 라운드 수를 갖는 Parameterized block cipher이다. 128비트 블럭길이에 대한 라운드 수는 20라운드이나 학계에서는 Security margin이 비교적 작아 라운드 수를 좀 더 증가시킬 것을 권고하고 있다.
  256비트 블럭길이에 대해서는 거의 분석이 이루어지지 않아 당분간은 128비트 블럭암호로만 사용될 것으로 보인다.
- TWOFISH: Schneier 등이 개발한 AES 2차 후보 알고리즘으로 Key-dependent S-box 16라운드의 Feistel 구조이다. 응용에 따라서 유연성 있게 Key scheduling을 할 수 있도록 한 것이 특징이다.
- MARS: IBM에서 제안한 AES 2차 후보 알고리즘으로 충분히 긴 가변길이의 키를 지원한다. 대부분의 기존 블럭암호의 구조와는 약간 다른 새로운 구조로 설계되었으나, 구조가 복잡하여 분석이 어렵다는 지적을 받고 있다.
- SERPENT: Biham 등이 제안한 AES 2차 후보 알고리즘으로 256비트까지의 키길이를 지원하는 SPN 구조의 128비트 블록암호이다. Bit-slice implementation을 염두에 두고 설계된 암호로 하드웨어 구현은 매우 용이하나 소프트웨어 성능은 상당히 떨어지는 편이다.
- CAST256: CAST128을 확장하여 AES 후보 알고리즘으로 제안하였다. 48라운드의 직렬 연산 구조로 인해 하드웨어나 소프트웨어적으로 효율성은 상당히 떨어지는 편이다.

2) 공개키(asymmetric key cryptography)

공개키(public key)와 개인키(private key) 혹은 비밀키가 존재하여 이중 어느 것 하나로 암호화(Encrypt)를 하면 다른 하나로 복호화(Decrypt)해야만 볼 수 있는 알고리즘이다. 이는 대칭키의 키 관리 문제로 대두된 것으로 공개키는 남에게 알려주고 개인키는 자신이 소장함으로 개인키(private key)의 안전을 꾀할 수 있다[5, 7, 10].

- Diffie-Hellman: key를 교환하기 위한 체계이다. 실제로 encryption과 decryption을 사용하는 방법이 아니라, 공유하고 있는 private key를 교환하기 위한 수단이다.

- RSA: MIT의 Ronald Rivest와 Adi Shamir 그리고, USC의 Leonard Adleman에 의해 개발된 잘 알려진 public key를 이용한 암호체계이다. 정보를 암호화시키거나 전자서명을 하는데 모두 사용할 수 있다. 구현에 따라 어떤 길이의 key도 사용될 수 있으며, 일반적으로 긴 길이의 key가 더 안전하다.
- ElGamal: 누승법과 단위 계산에 의한 알고리즘이다. RSA와 비슷한 용도로 사용될 수 있으며 일반적으로 길이가 긴 key가 더 안전하다.
- DSA(Digital Signature Algorithm): NSA에 의해 계발되어 NIST에 의해 FIPS (Federal Information Processing Standard)로 채택되었다. DSA의 key의 길이에 제한은 없으나 NIST에 의해 512에서 1024bit까지의 key만이 채택되었다. 이는 encryption으로의 사용을 위한 구현이 가능하지만, 전자서명을 위해서만 쓰일 수 있으며 DSS로도 불린다.

3) 비밀키와 공개키의 응용

암호화 시스템은 크게 보아 비밀키 암호시스템과 공개키 암호시스템, 그리고 공개키-비밀키의 조합방식으로 분류된다. 이때 공개키 암호화시스템은 보안성이 뛰어나지만 계산속도와 효율성 측면에서 난점이 있고, 비밀키 암호시스템은 송·수신자가 동일한 키에 의해 암호화 또는 복호화를 하는 것이며 이는 공개키 알고리즘에 비해 알고리즘이 매우 간단하므로 속도가 월등히 빠르고, 소프트웨어로 구현 시 파일의 크기가 작으며, 하드웨어로 구현하는 경우 회로가 간단해지는 경제적인 이유로 널리 이용되고 있으나 키의 관리에 결정적인 약점이 있다. 공개키-비밀키의 조합방식은 암호화와 복호화 과정에서 서로 같은 키를 사용하고, 그 키를 암호화하여 키의 전송 및 비밀 보관 등을 필요 없게 만든 시스템으로 양자간의 장점을 딴 이중암호화 방식이다.

비밀키 및 공개키 시스템의 대표적인 것으로 전자서명(digital signature)을 들 수 있다 [3, 13]. 이는 누군가의 private key로 암호화된, message digest이다. 이 암호화 과정을 서명(signing)이라고 하며, 전자 서명은 두 가지 주요한 기능을 지닌다. 첫 번째로, 파일이 변조되었는지 알 수 있으며, 둘째로 그 메시지에 사용한 사람이 누구인지 알 수 있다. 전자서명은 앞에서 언급했던 Public key encryption을 이용한다. 예를 들어 어떤 사람이 자신의 secret key를 이용하여 문서를 encrypt하였다면, 그 문서는 그 사람의 public key를 가지고 있는 사람만이 볼 수 있으며, 또한 그 문서를 그 secret key의 주인이 작성하였음을 확신할 수 있다. 반대로 어떤 사람이 어떤 문서를 public key로 decrypt의 역함수를 이용하

여 encrypt한 문서는 secret key를 가진 사람만이 볼 수 있기 때문에, 그 key의 주인만이 그 문서를 읽을 수 있다.

Ⅲ. AES 알고리즘들의 비교

3.1 비밀키 암호화 알고리즘들의 비교

DES와 AES의 암호화 알고리즘의 키 값과 블록사이즈, 속도 등을 비교하면 다음 Table 1과 같은데, 이 결과는 [1, 2, 8]을 정리한 것이다. 여기에서 BlockSize가 128bits인 것들은 Cast256, Mars, RC6, Rijndael, Twofishs 가 있는데 이중에서 RC6를 제외한 나머지는 MaxKeySise가 256이며 RC6만 제일 큰 2048bits를 가진다. 그리고 속도는 Blowfish, Cast128, RC5, Rijndael, Twofishs 등이 빠르지만, Blowfish나 Cast128 그리고 RC5의 경우엔 BlockSize가 너무 작아서 실용성이 떨어진다. 키 값의 사이즈와 속도를 비교해 볼 때 최근에 나온 RC6와 Rijndael과 Twofishs 등의 알고리즘이 효용성이 크다고 알려져 있다[1].

Table 1. DES와 AES의 암호화 알고리즘의 비교

ID	Name	Patented	MaxKeySize	BlockSize	Speed
05	Blowfish	No	448bits	64bits	2.46mb/sec
07	Cast128	No	128bits	64bits	2.60mb/sec
15	Cast256	Yes	256bits	128bits	1.68mb/sec
8	Gost	No	256bits	64bits	1.63mb/sec
12	IDEA	Yes	128bits	64bits	0.75mb/sec
13	Mars	Yes	128bits	128bits	1.38mb/sec
11	Misty1	Yes	128bits	64bits	1.01mb/sec
01	RC2	No	1024bits	64bits	0.47mb/sec
03	RC5	Yes	2048bits	64bits	2.67mb/sec
04	RC6	Yes	2048bits	128bits	1.66mb/sec
09	Rijndael	No	256bits	128bits	2.12mb/sec
06	Twofishs	No	256bits	128bits	2.12mb/sec

3.2 수행도 비교

Table 2. 플랫폼별 암호화 및 복호화 수행도 비교

	32-bit (C)	32-bit (Java)	64-bit (C and assembler)	8-bit (C and assembler)	32-bit smartcard (ARM)	Digital Signal Processors
MARS	II	II	II	II	II	II
RC6	I	I	II	II	I	II
Rijndael	II	II	I	I	I	I
Serpent	III	III	III	III	III	III
Twofish	II	III	I	II	III	I

Table 3. 플랫폼별 키 스케줄링 수행도 비교

	32-bit (C)	32-bit (Java)	64-bit (C and assembler)	8-bit (C and assembler)	Digital Signal Processors
MARS	II	II	III	II	II
RC6	II	II	II	III	II
Rijndael	I	I	I	I	I
Serpent	III	II	II	III	I
Twofish	III	III	III	II	III

Table 4. 전체적 수행도

	Encryption/Decryption	Key Setup
MARS	II	II
RC6	I	II
Rijndael	I	I
Serpent	III	II
Twofish	II	III

전체적으로 RC6와 Rijindael이 암호화, 복호화에서는 성능이 비슷하며 Rijndael의 경우 키 스케줄링이 좀 더 좋다. C로 32비트 암호화, 복호화를 구현하면 RC6가 빠름을 알 수 있고, 32bit의 경우 암호화 복호화는 RC6가 더 빠르다는 것은 입증된 바이다[2].

이를 통합해보면 암호화와 복호화 속도 및 키 스케줄링을 관점에서 RC6와 Rijndael이 적합

한 대안이 될 수 있다고 본다. 그러므로 여기서는 32bits의 블록 사이즈를 사용하는 것이므로 RC6 알고리즘을 채택하는데 무리가 없음을 알 수 있다.

Ⅳ. 암호화와 전자상거래

4.1 암호화 방식의 분류

공개키 암호 시스템은 키 값을 그냥 전달시키지 않고 공개키로 암호화 한 후 전달된 내용을 비밀키로 복호화 하면 되므로 키 관리 측면에서 우수하게 평가되고 있으며, 비밀키 보다 복잡한 알고리즘을 사용하기 때문에 일반적인 보안성도 높지만, 한편으로 암호화(encryption)와 복호화(decryption) 속도가 느림을 알 수 있다. 이 암호 시스템은 EC거래와는 달리 보안수준과 키관리에 주안점을 둘 경우 높은 보안수준을 요구하면서 그렇게 많지 않은 상대를 처리할 수 있는 G2G나 G2B에 적용시킬만한 수준이다.

보안성은 공개키 비밀키 모두 높은 수준이므로 등급 차이를 별로 주지 않았으며, 키 관리 문제는 비밀키의 노출 문제로 비밀키만 낮은 등급을 주었고, 속도는 전자상거래에 있어서 중요한 요소이기 때문에 차이를 두었다.

Table 5. E-business 분야별 암호화 효율성

암호화 방법	보안성	키관리	속도	종합 효율성	적용가능분야
Public Key	I	I	III	I	G2G, G2B
Public-Private Key	II	I	III	II	G2C, B2B, B2C
Private Key	II	III	I	II	C2C

비밀키 암호 시스템은 키 값을 상대방에게 전달하여야 하는 문제에 봉착하기 때문에 키 관리 측면에선 낮은 등급을 받았고, 보안성은 공개키 알고리즘보다는 느슨함으로 인해서 중간 정도의 등급을 받았지만 암호화와 복호화의 문제에선 공개키보다는 간단한 알고리즘을 이용하였기에 실제 구현 시 프로그램의 가벼움과 암호화 복호화가 빠르기 때문에 높은 등급을 받았다. 이 암호 시스템은 EC에 적용시키기엔 키 관리의 문제가 너무 커서 C2C에나 적합한 수준이다.

마지막으로 공개키-비밀키 알고리즘은 비밀키를 이용해 암호화 한 후 그 키 값을 공개키로

암호화하여 전달하는 방식으로 자체는 비밀키의 암호화를 하므로 보안성은 비밀키과 같고, 키 관리 문제는 비밀키에 적용시킨 키를 공개키로 암호화하므로 공개키와 같으며, 속도는 비밀키 암호 시스템의 속도에 키 값만 공개키로 암호화 복호화를 하면 되므로 비밀키와 비슷한 수준이다. 이 암호 시스템은 EC중에서 상당한 보안수준을 요구하면서 많은 상대를 처리 할 수 있는 G2B, G2C, B2B, B2C 등 거의 모두에 적용시킬만한 수준이다. SET은 비자카드와 마스터카드가 마이크로소프트, IBM, 넷스케이프, SAIC, GTE, 및 VeriSign 등으로부터의 기술적 지원을 바탕으로 인터넷을 이용한 보안이 보장되는 카드거래를 위해 도입한 전자상거래 표준작업중인 사항이다. SET 프로토콜은 DES와 RSA를 복합적으로 사용하고 또 전자서명과 해시(Hash)기술을 혼합한 1024비트 체계이다. SET는 개인키 및 공개키를 쌍으로 비대칭키로 데이터를 암호화해 전송한다. 예를 들어 구매자가 카드번호와 주문서를 입력하면 카드회사는 판매업체 서버로 자사의 공개키를 알려주고 고객에게도 공개키와 개인키를 부여한다. 판매업체에서는 고객 정보를 개인키로 풀고 고객 공개키를 이용해 인증하는 과정을 거친다.

4.2 암호화 방식과 E-business의 관계

전자상거래의 형태를 80-20룰을 기준으로 나누어 분석한 연구에 따르면, 중저가의 대량의 상품이 현재 전자상거래의 주류를 이룬다는 것이다[11, 6]. 즉, B2C가 전자상거래의 최대 규모를 형성하고 있다는 것이며, 한편 이것은 보안문제를 소비자들이 아직 확신할 수 없기 때문에 최악의 경우 손해를 감수할 수 있는 금액의 표준화된 상품(예, 아마존을 통한 서적의 구매)에 치중하고 있다는 것이다 [3]. 따라서 암호화 방식에 있어서 현재로서는 활용이 미미하지만 B2C 영역의 해결이 최대의 과제이며 이에 대한 효과적인 대안되는 공개키-비밀키의 조합 접근법이 현재 E비즈니스에서의 최대의 과제 중 하나임을 알 수 있다. 현재 타원곡선암호(Elliptic Curve Cryptosystem) 역시 주목할 만한 기술로서 타원곡선이라고 불리는 수식을 기반으로 암호화 및 복호화를 수행하는 방식이다. 특히 키 크기가 작아서 스마트카드(IC카드)와 같은 기기에 적용되기에 매우 적합하므로 가능성이 매우 큰 방식이다.

V. 결 론

암호화(Encryption) 문제는 전자상거래(E-business)를 포함한 인터넷응용에서 현재 가장 핵심적인 요소의 하나로 인식되고 있다. 왜냐하면 보안이 보장되지 않으면 거래자체가 시작되지 않기 때문이다. 암호화 시스템은 크게 보아 비밀키 암호시스템과 공개키 암호시스템, 그리

고 공개키-비밀키의 조합방식으로 분류된다. 이때 공개키 암호화시스템은 보안성이 뛰어나지만 계산속도와 효율성 측면에서 난점이 있고, 비밀키 암호시스템은 송·수신자가 동일한 키에 의해 암호화 또는 복호화를 하는 것이며 이는 공개키 알고리즘에 비해 알고리즘이 매우 간단하므로 속도가 월등히 빠르고, 소프트웨어로 구현 시 파일의 크기가 작으며, 하드웨어로 구현하는 경우 회로가 간단해지는 경제적인 이유로 널리 이용되고 있으나 키의 관리에 결정적인 약점이 있다. 공개키-비밀키의 조합방식은 암호화와 복호화 과정에서 서로 같은 키를 사용하고, 그 키를 암호화하여 키의 전송 및 비밀 보관 등을 필요 없게 만든 시스템으로 양자간의 장점을 딴 이중암호화 방식이다.

본 연구에서 처음으로 분류를 시도한 암호화와 전자 상거래와의 관련성은 다음과 같다. 우선 공개키만의 적용은 대상이 한정적이며 계산에 막대한 노력이 수반되나 보안성이 아주 뛰어난 BB 혹은 GG가 적합하고, 비밀키 만을 적용하는 데에는 C2C가 바람직하다고 판단하였고, 공개키-비밀키의 조합은 전자상거래에 가장 보편적인 B2C, G2C, 및 G2B 등에 적합하다고 보았다.

본 연구의 기대효과는 이러하다. 우선 현재 E-business의 시장성장의 병목의 하나로서 보안문제가 거론 되고 있는 바, 본 연구의 접근법에 의해 E비즈니스관련 보안문제해결을 위한 방향을 제시한 의미가 있으며, 특히 E-business에서 가장 큰 시장규모를 차지하게 될 기업-일반고객(B2C) 등의 관계에 적합한 보안모델에 초점을 두었다는 점이다. 이러한 암호화 알고리즘을 통해서 갈수록 보안성이 강조되고 EC의 활성화를 위한 장애중의 하나인 사용자의 보안에 대한 확신을 줄 경우 G2B(Government to Business), G2C(Government to Customer), B2B(Business to Business), B2C(Business to Customer) 등을 비롯하여 향후에는 거의 전체적 전자상거래 분야로의 보안문제의 중요성이 점차 증대될 것으로 보인다.

Acknowledgement: 본 연구는 1998년도 한국경영정보학회 학술대회에서 저자의 발표사항에 질의내역을 추가하여 확장한 것이다.

참 고 문 헌

1. AES http://www.nist.gov/aes
2. A. Ambainis, M. Jakobsson, H. Lipmaa(2004). Cryptographic Randomized Response Techniques. Public Key Cryptography, pp. 425-438.

3. M. Bichler, A. Segev(2001). Methodologies for the Design of Negotiation Protocols on E-markets. Computer Networks 37(2). pp. 137-152.
4. J. Borst, B. Preneel, J. Vandewalle(1999). Linear Cryptanalysis of RC5 and RC6. Proc. Fast Software Encryption, pp. 16-30.
5. R. P. Brent(2000). Recent Progress and Prospects for Integer Factorisation Algorithms. Proc. COCOON, pp. 3-22.
6. E.C. Cheng, G. Loizou(2000). A Publish/Subscribe Framework: Push Technology in E-Commerce. BNCOD, pp. 153-170.
7. W. Fischer, J. P. Seifert(2002). Note on Fast Computation of Secret RSA Exponents. Proc. ACISP, pp. 136-143.
8. P. Gaudry, E. Schost (2004). Construction of Secure Random Curves of Genus 2 over Prime Fields. EUROCRYPT, pp. 239-256.
9. M.E. Locasto, A.D. Keromytis(2004). Camouflage FS: Increasing the Effective Key Length in Cryptographic Filesystems on the Cheap. ACNS, pp. 1-15.
10. N. Nedjah, L.M. Mourelle, M.P. Cardoso(2006). A Compact Piplined Hardware Implementation of the AES-128 Cipher. ITNG, pp. 216-221.
11. B. Schneier, Applied Cryptography, 2nd edition, John Wiley & Sons, 1996.
12. Z. Tian, L.Y. Liu, J. Li, J. Chung, V. Guttemukkala(1999). Business-to-Business E-Commerce with Open Buying on the Internet. WECWIS, pp. 56-62.
13. C. Wagner, E. Turban(2002). Are Intelligent E-Commerce Agents Partners or Predators?. Communications of the ACM 45(5) pp. 84-90.
14. T.I. Wang, K. H. Tsai, M. Lee(2004). A Two-Layer Cryptographic Scheme for an e-Service Framework Based on Mobile Agents. EEE, pp. 98-105.