ECE 297:11 Lecture 14

Survey of public key cryptosystems

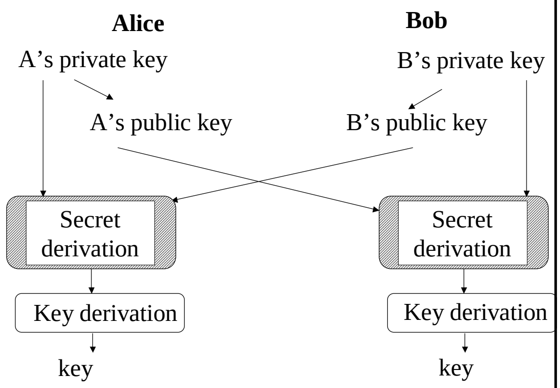
Bases of the public cryptosystems security

	Factorization	Discrete Logarithm	Elliptic Curve Discrete Logarithm
Given:	$\mathbf{N} = p \cdot q$	$\mathbf{y} = \mathbf{g}^x \bmod \mathbf{p} = \underbrace{g \cdot g \cdot g \cdot \ldots \cdot g}_{x\ \text{times}}$ constants **p**, **g**	$\mathbf{Q} = x \cdot \mathbf{P} = \underbrace{P+P+\ldots+P}_{x\ \text{times}}$ **P** - point of an elliptic curve
Unknown:	**p, q**	**x**	**x**

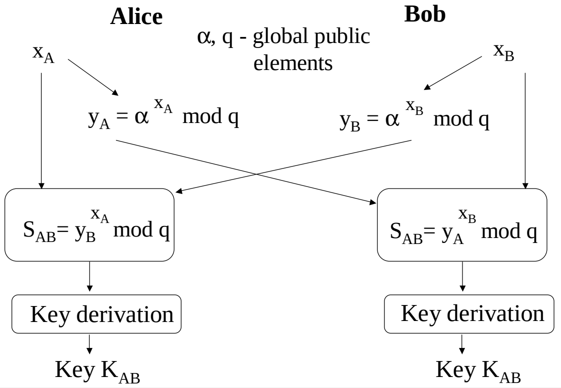
Most known public key cryptosystems

	Based on the difficulty of		
	Factorization	**Discrete logarithm**	**Elliptic curve discrete logarithm**
Signature	RSA	DSA, N-R	EC-DSA
Encryption	RSA	El-Gamal	EC-El-Gamal
Key agreement	RSA	Diffie-Hellman (DH)	EC-DH

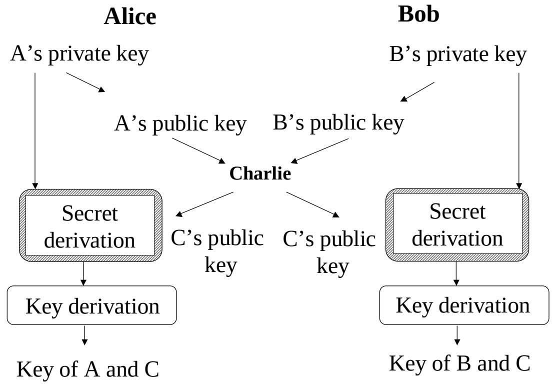
Key agreement



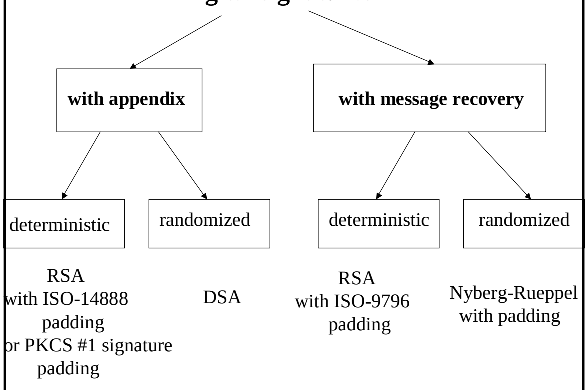
Diffie-Hellman key agreement scheme



Man-in-the-middle attack



Digital signatures



Genesis of DSS

1976 public key cryptography, Diffie-Hellman
1978 RSA (patent in 1983)
1982 NIST solicitation for a public key signature algorithm
1984 El Gamal algorithm (not pattented)
1989 Schnorr algorithm (patent in 1991 in U.S. and many other countries)
1990 the primary candidate considered by NIST is RSA
1991 NIST announces DSA
1994 DSS published as FIPS PUB 186

Digital Signature Algorithm

System parameters

May be shared by a group of users or belong to a single user; known to everybody

q - 160-bit prime
p - L-bit prime, such that q | p-1
where L = 512 + 64·k

g = $h^{(p-1)/q}$ mod p where 1 < h < p-1, such that g>1

From Fermat's theorem
g^q mod p = h^{p-1} mod p = 1
g - generator of the cyclic group of order q in Zp*

Digital Signature Algorithm

Public and private key

Private key

x - arbitrary 160 bit number 0 < x < q

Public key

y = g^x mod p 0 < y < p

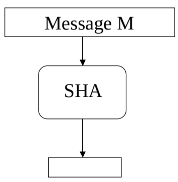
L - bit number

DSA: Signature generation

1. Choose random *message private key* 1< ***k*** < *q* (secret, different for each message)

2. Compute *message public key* ***r*** = (g^k mod *p*) mod *q*

3. Compute *hash value*



SHA(M)

4. Compute

$s = k^{-1}$ (SHA(M) + $x \cdot r$) mod q

SGN(M) = *r* || *s*

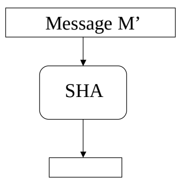
160 bit 160 bit 40 bytes

DSA: Signature verification

1. Compute *hash value*

r'	*s'*

[SGN(M)]'



SHA(M')

2. Compute

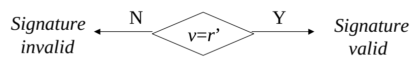
w = $(s')^{-1}$ mod *q*

3. Compute
u1 = SHA(M')·*w* mod *q*

4. Compute
u2 = *r*' ·*w* mod *q*

5. Compute
v = (($g^{u1} \cdot y^{u2}$) mod *p*) mod *q*

6. Compare



DSA vs. RSA

Functionality

DSS cannot be used for encryption

Advantages

export rules much less restrictive

certain countries do not allow encryption

Disadvantages

additional algorithm must be standardized and implemented for key exchange

DSS can be combined with the Diffie-Hellman key exchange scheme

El-Gamal Encryption

System parameters

May be shared by a group of users or belong to a single user; known to everybody

p - prime

g - generator of the group Zp*

El-Gamal Encryption

Public and private key

Private key

x - arbitrary number $1 \le x \le p-2$

Public key

$y = g^x \bmod p$ $0 < y < p$

El-Gamal: Encryption

1. Choose random *message private key* $1 \le \boldsymbol{k} \le p-2$, relatively prime with p-1 (secret, different for each message)
2. Compute *message public key* $\boldsymbol{r} = g^k \bmod p$
3. Compute

$$\boldsymbol{c} = y^k \cdot M \bmod p$$

$$C(M) = \boldsymbol{r} \,\|\, \boldsymbol{c}$$

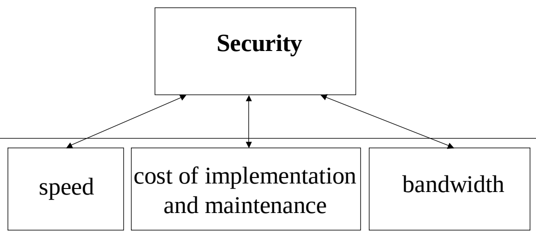
El-Gamal: Decryption

r	*c*

C(*M*)

$$M = c \cdot (r^x)^{-1} \bmod p$$

Choice of a public key cryptosystem



Parameters of implementation

Strategy of fair comparison

All algorithms have a **variable key length**

Best attacks specific for each cryptosystem

Security of various cryptosystems
depends to a different extant on the key length

Comparison of implementation characteristics (in particular speed) under the assumption that selected key sizes guarantee the same security level

Best known attacks

Basis of the cryptosystem security	**Factorization**	**Discrete Logarithm**	**Elliptic Curve Discrete Logarithm**
Best known attack	*General Number Field Sieve*	1. *General Number Field Sieve* 2. *Parallel collision search*	*2. Parallel collision search*
Complexity of the attack:	subexponential	1. subexponential 2. exponential	exponential

Best known attacks

Basis of the cryptosystem security	**Factorization**	**Discrete Logarithm**	**Elliptic Curve Discrete Logarithm**
Cryptosystem	RSA	DSA, DH	EC-DSA EC-DH
Security parameter	Modulus N	1. Length of the modulus p 2. Size q of the subgroup generated by g	Size q of the subgroup generated by P
Typical lengths of the security parameter (in bits)	768-2048	1. 768-2048 2. 160 (for DSA)	140-200

Theoretical computational security of the best known attacks

Basis of the cryptosystem security	**Complexity of the best known attack**
Factorization	**subexponential** $L_N[1/3, 1.92] = exp((1.92 + o(1))\cdot(\ln \boldsymbol{N})^{1/3})\cdot(\ln\ln \boldsymbol{N})^{2/3})$
Discrete Logarithm	**subexponential** $L_p[1/3, 1.92] = exp((1.92 + o(1))\cdot(\ln \boldsymbol{p})^{1/3})\cdot(\ln\ln \boldsymbol{p})^{2/3})$
Elliptic Curve Discrete Logarithm	**exponential** $(\pi\cdot\mathbf{q} / 2)^{1/2}/r$ r - number of processors working in parallel

Practical records

Basis of the cryptosystem security	**Factorization**	**Discrete Logarithm**	**Elliptic Curve Discrete Logarithm**
Number of bits of the security parameter	512	283?	108
Challenges regarding breaking the cryptosystem	**RSA Data Security Challenge, 1991-**	–	**Certicom challenge, 1997-**

Practical implementations of attacks

Discrete logarithm, DSA, DH

Year	**Number of bits of p**	**Number of decimal digits of p**	**Method**	**Estimated amount of computations**
1990	191	57	NFS-COS	
1996	248	74	NFS-DL	
1998	**283**	**85**	**NFS-COS**	31 MIPS-years
1998	430	129 (p of the special form)	SNFS	

Practical implementations of attacks

Elliptic curve discrete logarithm problem, ECC-DSA, DH

Year	Curve	Number of bits of q	Number of decimal digits of q	Method	Number of group operations
II.1998	ECC2-89	89	27	ρ-Pollard	1.8×10^{13}
I.1998	ECCp-89	89	27	ρ-Pollard	3.0×10^{13}
V.1998	ECC2K-95	95	29	ρ-Pollard	2.2×10^{13}
III.1998	ECCp-97	97	30	ρ-Pollard	2.0×10^{14}
IX.1999	ECC2-97	97	30	ρ-Pollard	1.0×10^{14}
IV. 2000	ECC2K -108	108	33	ρ-Pollard	2.0×10^{15}

Elliptic Curve Cryptosystems - ECC

Advantages

- first true alternative for RSA
- several times shorter keys
- fast and compact implementations, in particular in hardware
- a family of cryptosystems, instead of a single cryptosystem

Elliptic Curve Cryptosystems - ECC

Disdvantages

- complex mathematical description
- short period of research on the cryptanalysis

Elliptic Curve Cryptosystems vs. RSA

Certicom	**RSA Data Security Inc.**
ECC	**RSA** ECC
Security Builder	BSAFE
Efficient software and hardware implementations	Efficient software implementations
ECC - “cryptography of the XXI century”	ECC – cryptography for low-risk applications

Fact or myth?

RSA is much more secure because the factorization problem was studied much longer than elliptic curve discrete logarithm problem

Factorization problem studied intensively since **the end of 70’s**	Studies on factorization **before the era of computers and computer networks** is irrelevant
Elliptic curve discrete logarithm problem studied intensively since **the beginning of 90’s**	Studies on **attacks against discrete logarithms in GF(p)** conducted earlier. Many of these attacks apply to the elliptic curve discrete logarithms.

Progress in algorithms for solving the discrete logarithm problem

1997 *N. Smart*
1997 *T. Satoh, K. Araki*

Fast algorithm for a special class of curves

7.04.98 *R. Gallant, R. Lambert, S. Vanstone;* Certicom
8.04.98 *M. Wiener i R. Zuccherato;* Entrust

Algorithm speeding up computations √2m times for Koblitz curves over $GF(2^m)$

For a randomly selected curve, neither attack applies

Workshops on Elliptic Curve Cryptography, since 1997

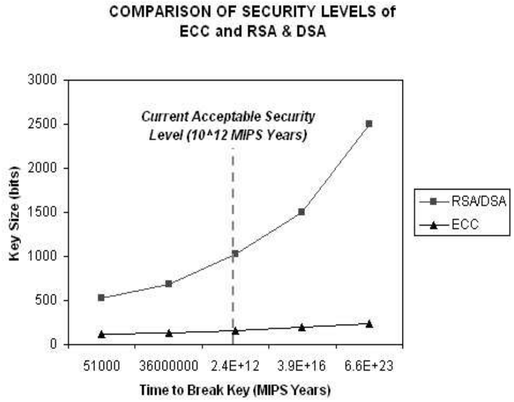
Sponsors: MasterCard, Mondex, etc.

Fact or myth?

Key length necessary to obtain the same level of security for RSA and Elliptic Curve Cryptosystems grows faster for RSA

True, if one takes into account only the **number of operations** necessary to conduct the attack

Untrue, if one takes into account much larger memory requirements for attacks against RSA



RAM requirements in the NFS factorization method

Number of bits of N	Memory in the first phase of the algorithm (clients)	Memory in the second phase of the algorithm (server)
428	64 MB	2 GB
512	160 MB	20 GB
1024	256 GB	~100 TB

Equivalent key sizes

according to Robert Silverman, RSA Inc., 1999

Assumption: ***The same amount of arithmetic operations***

RSA/DSA	ECC	Symmetric ciphers	Number of arithmetic operations
512	119	56	$1{,}7 \times 10^{19}$
768	144	69	$1{,}1 \times 10^{23}$
1024	**163**	**79**	$\mathbf{1{,}3 \times 10^{26}}$
2048	222	100	$1{,}5 \times 10^{35}$

Equivalent key sizes

according to Michael Wiener, Entrust Technologies

Basic assumption: *The same number of instructions in MIPS-years*

RSA/DSA	ECC		Number of instructions w MIPS-years
Software attack	Software attack	Hardware attack	
1024	**138**	**170**	**3×10^{11}**

Equivalent key sizes

according to Michael Wiener

Detailed assumptions (1)

Hardware attack based on ASICs:

- clock frequency 64 MHz
- 70 levels of pipelining
- cost $16

Equivalent key sizes
according to Michael Wiener

Detailed assumptions (2)

Number of PCs, 300 MHz, necessary to break RSA-1024

2^{30} PC-years

Number of ASICs necessary to break ECC-k

$2^{k/2-51}$ ASIC-years

Equivalent key sizes
according to Michael Wiener

Detailed assumptions (3)

Cost of access to a PC

$250

Cost of an ASIC

$16

1 PC-year ≈ 16 ASIC-years

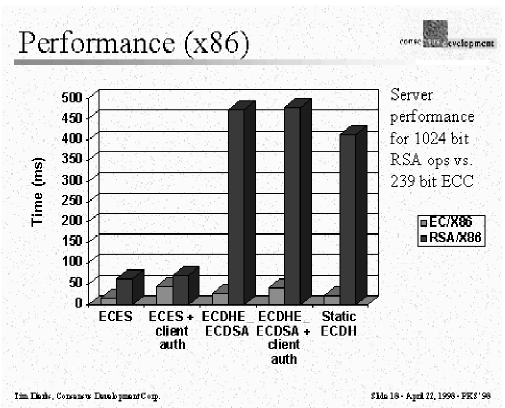
k=170

Digital Signature Timings
Pentium Pro, 200 MHz, Michael Wiener, Entrust

	RSA-1024 (e=3)	DSA-1024	ECDSA-170
Signature generation	43 ms	7 ms	**5 ms**
Signature verification	**0.6 ms**	27 ms	19 ms
Key generation	1100 ms	7 ms	**7 ms**

Digital Signature Timings
Pentium Pro, 180 MHz, Scott Contini, RSA DSI

	RSA-1024 (e=3)	DSA-1024	ECDSA-170
Signature generation	47 ms	28 ms	**6 ms**
Signature verification	**1 ms**	52 ms	30 ms



Binary code size

	RSA	DSA	EC-DSA
Generation of system parameters	**N/A**	small	very large
Key generation	medium	**very small**	**very small**
Core operations	**small**	small	medium

Which cryptosystem is the best? (1)

Secure electronic mail

- speed of operations is not critical, security and trust of customers are more important
- message encrypted using a symmetric key cryptosystem

A key for a symmetric key cryptosystem encrypted once for each receiver

All operations performed by a sender

A key for a symmetric key cryptosystem decrypted separately by each receiver

Load distributed among receivers

Advantage: RSA

Which cryptosystem is the best? (2)

Use in public key certificates

- each certificate and CRL are signed only once but verified hundreds of times

Advantage: RSA

Which cryptosystem is the best? (3)

Wireless communication

- large cost of transmission
- shorter keys in ECCs
- shorter signatures and certificates in ECCs and DSA
- shorter messages in the key agreement schemes based on ECCs

Advantage: ECC

Which cryptosystem is the best? (4)

Hardware implementation

- small area of integrated circuits implementing ECC, in particular ECCs over $GF(2^m)$
- faster decryption and key generation

Advantage: ECC

Which cryptosystem is the best? (5)

Smart cards

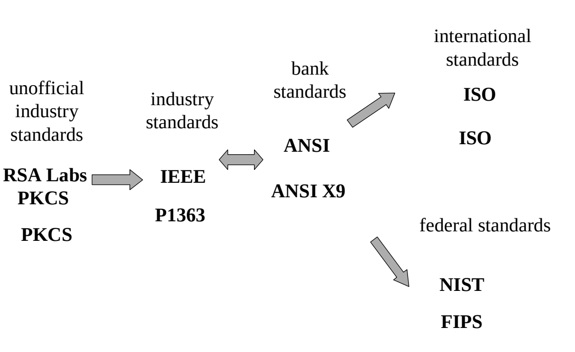
ECCs

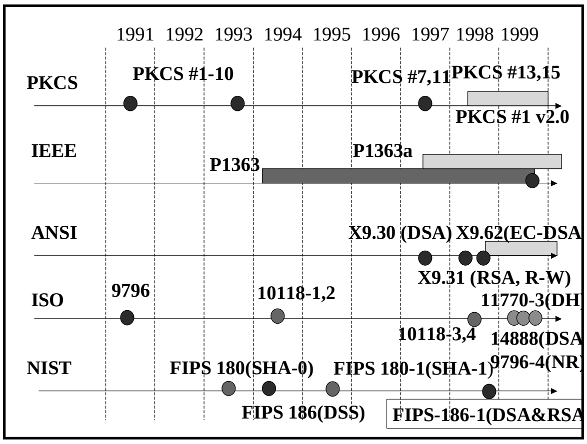
- smaller EEPROM requirements
- do not require an arithmetic cooprocessor (at least for a class of curves over $GF(2^m)$
- smaller requirements on the interface with a card reader
- allow to generate a key on the card

Advantage: ECC

Cryptographic standards

Public-key Cryptography Standards





PKCS
Public-Key Cryptography Standards

Informal Industry Standards

developed by RSA Laboratories

in cooperation with
Apple, Digital, Lotus, Microsoft, MIT, Northern Telecom, Novell, Sun

First, except PGP, formal specification of RSA and formats of messages.

Industry standards - PKCS

	factorization	discrete logarithm	Elliptic curve discrete logarithm
encryption	**PKCS #1** **RSA**		PKCS #13 new scheme
signature	**PKCS #1** **(RSA i R-W)**		PKCS #13 EC-DSA
key agreement		**PKCS #2** **DH**	PKCS #13 EC-DH1, 2 EC-MQV

IEEE P1363

Working group of IEEE including representatives of major cryptographic companies and university centers from USA, Canada and other countries

Part of the Microprocessors Standards Committee

Modern, open style

Quaterly meetings + multiple teleconferences + + discussion list + very informative web page with the draft versions of standards

IEEE P1363

Combined standard including the majority of modern public key cryptography

Several algorithms for implementation of the same function

Tool for constructing other, more specific standards

Specific applications or implementations may determine a profile (subset) of the standard

IEEE P1363

	factorization	discrete logarithm	Elliptic curve discrete logarithm
encryption	**RSA** with OAEP		
signature	**RSA** & **R-W** with ISO-14888 or ISO 9796	**DSA,** **NR** with ISO 9796	**EC-DSA, EC-NR** with ISO 9796
key agreement		**DH1 DH2** and **MQV**	**EC-DH1, EC-DH2** and **EC-MQV**

IEEE P1363a

	factorization	discrete logarithm	Elliptic curve discrete logarithm
encryption	RSA with OAEP	**new scheme**	**new scheme**
signature	RSA & R-W with ISO-14888 or ISO 9796	DSA, NR with ISO-9796	EC-DSA, EC-NR with ISO 9796
key agreement	**new scheme**	DH1 DH2 & MQV	EC-DH1 EC-DH2 & EC-MQV

ANSI X9
American National Standards Institute

Work in the subcommittee X9F
developing standards for **financial institutions**

Standards for the wholesale
(e.g., interbank)
and retail transactions
(np. bank machines, smart card readers)

ANSI represents U.S.A. in **ISO**

ANSI X9 Standards

	factorization	discrete logarithm	Elliptic curve discrete logarithm
encryption	**X9.44** **RSA**		
signature	**X9.31** (**RSA** & **R-W**)	**X9.30** **DSA**	**X9.62** **EC-DSA**
key agreement		**X9.42** **DH1, DH2, MQV**	**X9.63** **EC-DH**1, 2 **EC-MQV**

NIST FIPS
National Institute of Standards and Technology
Federal Information Processing Standards

American Federal Standards

Required in the government institutions

Original algorithms developed in cooperation with the National Security Agency (NSA)

NIST - FIPS

	factorization	discrete logarithm	Elliptic curve discrete logarithm
encryption			
signature	**FIPS 186-1** **RSA**	**FIPS 186** **DSA**	
key agreement			

American Standards

	RSA	DSA, DH	EC-DSA EC-DH
Federal		FIPS 186	
Banking	X9.31	X9.30 X9.42	X9.62 X9.63
Industry	IEEE P1363 PKCS-1	IEEE P1363 PKCS-2	IEEE P1363 PKCS-13

ISO
International Organization for Standardization

International standards

Common standards with **IEC** - International Electrotechnical Commission

ISO/IEC JTC1 **SC 27**

Joint Technical Committee 1, Subcommitte 27

Full members (21):

Australia, Belgium, Brazil, Canada, China, Denmark, Finland, France, Germany, Italy, Japan , Korea., Holland , Norway , Poland, Russia , Spain, Sweden, Switzerland , UK, USA

ISO: International Organization for Standardization

Long and laborious process of the standard development

Minimum 3 years

- Study period
- NP - New Proposal
- WD - Working Draft
- CD - Committee Draft
- DIS - Draft International Standard
- IS - International Standard

Review of the standard after 5 years = ratification, corrections or revocation

International standards ISO

	factorization	discrete logarithm	Elliptic curve discrete logarithm
encryption			
signature	ISO 9796-1 ISO 9796-2	ISO-14888-3 ISO 9796-4	ISO-14888-3 ISO 9796-4
key agreement		ISO-11770-3	ISO-11770-3

Secure key sizes

	factorization	Discrete logarithm	Elliptic curve discrete logarithm
PKCS			
IEEE P1363			
ANSI X9	≥ 1024	≥ 1024	≥ 160
NIST FIPS		≥ 1024	
ISO			

Padding schemes

	encryption	Signatures with appendix	Signatures with message recovery
PKCS	**OAEP** PKCS #1	**PKCS #1**	
IEEE P1363	**OAEP**	**ISO 14888**	**ISO 9796**
ANSI X9	**OAEP**	**ISO 14888**	**ISO 9796**
NIST FIPS			
ISO		**ISO 14888**	**ISO 9796**

Standard Internet Protocols

Secure e-mail

S/MIME v.2	RSA
v.3	RSA, DSA, DH

Secure WWW

SSL v. 3.0	RSA, DSA, DH, **proposed extension with ECCs**

Secure payment card protocols

SET	RSA, **proposed extension with ECCs**

Virtual Private Networks

IPSec	DH, **EC-DH**

Patents - only U.S. and Canada

RSA	DSA, DH	EC-DSA, EC-DH
Patent expired in **2000**	DH Patent expired in **1997**	No patents for cryptosystems themselves. Over **40 patent petitions** regarding implementation details, *Certicom Inc.*

Summary

- RSA in common use, ECC struggle to enter the market
- New standards will support all three types of cryptosystems
- ECC particularly advantages in environments with limited bandwidth and storage (e.g., cellular telephones, pagers, smart cards)
- If there is no breakthrough in cryptanalysis the market will be shared among two (or three) classes of cryptosystems