Cryptographic Technologies

Chapter 6

Crypto

- **Cryptology** — The art and science of making and breaking "secret codes"
- **Cryptography** — making "secret codes"
- **Cryptanalysis** — breaking "secret codes"
- **Crypto** — all of the above (and more)

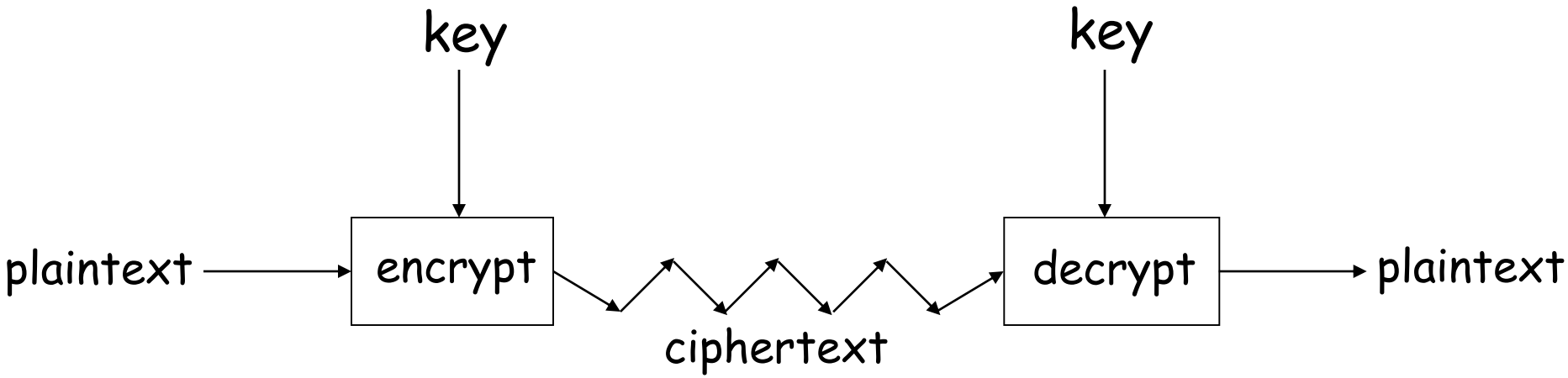
How to Speak Crypto

- A *cipher* or *cryptosystem* is used to *encrypt* the *plaintext*
- The result of encryption is *ciphertext*
- We *decrypt* ciphertext to recover plaintext
- A *key* is used to configure a cryptosystem
- A *symmetric key* cryptosystem uses the same key to encrypt as to decrypt
- A *public key* cryptosystem uses a *public key* to encrypt and a *private key* to decrypt (sign)

Crypto

- Basis assumption
  - The system is completely known to the attacker
  - Only the key is secret
- Also known as **Kerckhoffs Principle**
  - Crypto algorithms are not secret
- Why do we make this assumption?
  - Experience has shown that secret algorithms are weak when exposed
  - Secret algorithms never remain secret
  - Better to find weaknesses beforehand

Crypto as Black Box



A generic use of crypto

Simple Substitution

- Plaintext: fourscoreandsevenyearsago
- Key:

Plaintext	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
Ciphertext	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C

- Ciphertext:
  IRXUVFRUHDAGVHYHABHDUVDIR
- Shift by 3 is "Caesar's cipher"

Ceasar's Cipher Decryption

- Suppose we know a Ceasar's cipher is being used
- Ciphertext: VSRQJHEREVTXDUHSDQWU

Plaintext	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
Ciphertext	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C

- Plaintext: spongebobsquarepants

Not-so-Simple Substitution

- Shift by n for some n $\in$ {0,1,2,…,25}
- Then key is n
- Example: key = 7

Plaintext	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
Ciphertext	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G

Cryptanalysis I: Try Them All

- A simple substitution (shift by n) is used
- But the key is unknown
- Given ciphertext: CSYEVIXIVQMREXIH
- How to find the key?
- Only 26 possible keys — try them all!
- **Exhaustive key search**
- Solution: key = 4

Even-less-Simple Substitution

- Key is some permutation of letters
- Need not be a shift
- For example

Plaintext	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
Ciphertext	J	I	C	A	X	S	E	Y	V	D	K	W	B	Q	T	Z	R	H	F	M	P	N	U	L	G	O

- Then 26! > 2^{88} possible keys!

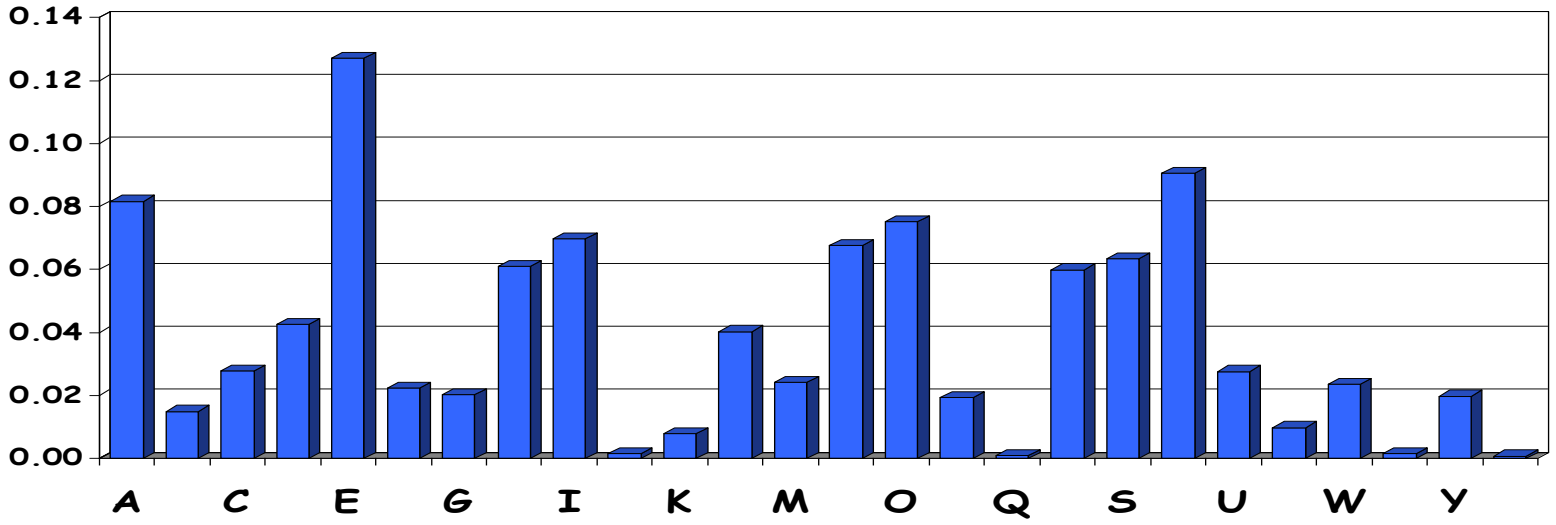
Cryptanalysis II: Be Clever

- We know that a simple substitution is used
- But not necessarily a shift by n
- Can we find the key given ciphertext:

PBFPVYFBQXZTYFPBFEQJHDXXQVAPTPQJKTOYQWIPBVWLXTOXBT
FXQWAXBVCXQWAXFQJVWLEQNTOZQGGQLFXQWAKVWLXQW
AEBIPBFXFQVXGTVJVWLBTPQWAEBFPBFHCVLXBQUFEVWLXGDP
EQVPQGVPPBFTIXPFHXZHVFAGFOTHFEFBQUFTDHZBQPOTHXTY
FTODXQHFTDPTOGHFQPBQWAQJJTODXQHFOQPWTBDHHIXQV
APBFZQHCFWPFHPBFIPBQWKFABVYYDZBOTHPBQPQJTQOTOGH
FQAPBFEQJHDXXQVAVXEBQPEFZBVFOJIWFFACFCCFHQWAUVW
FLQHGFXVAFXQHFUFHILTTAVWAFFAWTEVOITDHFHFQAITIXP
FHXAFQHEFZQWGFLVWPTOFFA

Cryptanalysis II

- Can't try all 2^{88} simple substitution keys
- Can we be more clever?
- English letter frequency counts…



Cryptanalysis II

- Ciphertext:

PBFPVYFBQXZTYFPBFEQJHDXXQVAPTPQJKTOYQWIPBVWLXTOXBTFXQWA
XBVCXQWAXFQJVWLEQNTOZQGGQLFXQWAKVWLXQWAEBIPBFXFQVX
GTVJVWLBTPQWAEBFPBFHCVLXBQUFEVWLXGDPEQVPQGVPPBFTIXPFHXZ
HVFAGFOTHFEFBQUFTDHZBQPOTHXTYFTODXQHFTDPTOGHFQPBQWAQ
JJTODXQHFOQPWTBDHHIXQVAPBFZQHCFWPFHPBFIPBQWKFABVYYDZB
OTHPBQPQJTQOTOGHFQAPBFEQJHDXXQVAVXEBQPEFZBVFOJIWFFACF
CCFHQWAUVWFLQHGFXVAFXQHFUFHILTTAVWAFFAWTEVOITDHFHFQ
AITIXPFHXAFQHEFZQWGFLVWPTOFFA

- Decrypt this message using info below

Ciphertext frequency counts:

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
21	26	6	10	12	51	10	25	10	9	3	10	0	1	15	28	42	0	0	27	4	24	22	28	6	8

Cryptanalysis: Terminology

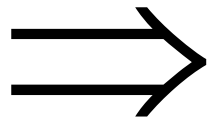
- Cryptosystem is **secure** if best know attack is to try all keys
- Cryptosystem is **insecure** if any shortcut attack is known
- By this definition, an insecure system might be harder to break than a secure system!

Double Transposition

- Plaintext: attackxatxdawn

	col 1	col 2	col 3
row 1	a	t	t
row 2	a	c	k
row 3	x	a	t
row 4	x	d	a
row 5	w	n	x

Permute rows and columns



	col 1	col 3	col 2
row 3	x	t	a
row 5	w	x	n
row 1	a	t	t
row 4	x	a	d
row 2	a	k	c

- Ciphertext: xtawxnattxadakc
- Key: matrix size and permutations (3,5,1,4,2) and (1,3,2)

One-time Pad Encryption

e=000 h=001 i=010 k=011 l=100 r=101 s=110 t=111

Encryption: Plaintext ⊕ Key = Ciphertext

	h	e	i	l	h	i	t	l	e	r
Plaintext:	001	000	010	100	001	010	111	100	000	101
Key:	111	101	110	101	111	100	000	101	110	000
Ciphertext:	110	101	100	001	110	110	111	001	110	101
	s	r	l	h	s	s	t	h	s	r

One-time Pad Decryption

e=000 h=001 i=010 k=011 l=100 r=101 s=110 t=111

Decryption: Ciphertext ⊕ Key = Plaintext

	s	r	l	h	s	s	t	h	s	r
Ciphertext:	110	101	100	001	110	110	111	001	110	101
Key:	111	101	110	101	111	100	000	101	110	000
Plaintext:	001	000	010	100	001	010	111	100	000	101
	h	e	i	l	h	i	t	l	e	r

Goals of Cryptography

- Four primary goals
- Many applications provide multiple cryptographic benefits simultaneously
- Confidentiality is most commonly addressed goal
  - The meaning of a message is concealed by encoding it
  - The sender encrypts the message using a cryptographic key
  - The recipient decrypts the message using a cryptographic key that may or may not be the same as the one used by the sender

Goals of Cryptography (continued)

- Integrity
  - Ensures that the message received is the same as the message that was sent
  - Uses hashing to create a unique message digest from the message that is sent along with the message
  - Recipient uses the same technique to create a second digest from the message to compare to the original one
  - This technique only protects against unintentional alteration of the message
  - A variation is used to create digital signatures to protect against malicious alteration

Goals of Cryptography (continued)

- Nonrepudiation
  - The sender of a message cannot later claim he/she did not send it
  - Available with asymmetric cryptosystems that can create digital signatures
- Authentication
  - A user or system can prove their identity to another who does not have personal knowledge of their identity
  - Accomplished using digital certificates
  - Kerberos is a common cryptographic authentication system

Cryptographic Algorithms

- Two types of cryptographic algorithms
  - Symmetric and asymmetric
- A cryptographic algorithm is used to encrypt a message
  - Change from plaintext to ciphertext
- And then decrypt the message
  - Change from ciphertext back to plaintext
- Early algorithms embodied “security through obscurity”
- Current algorithms are rigorously and openly examined

Cryptographic Algorithms (continued)

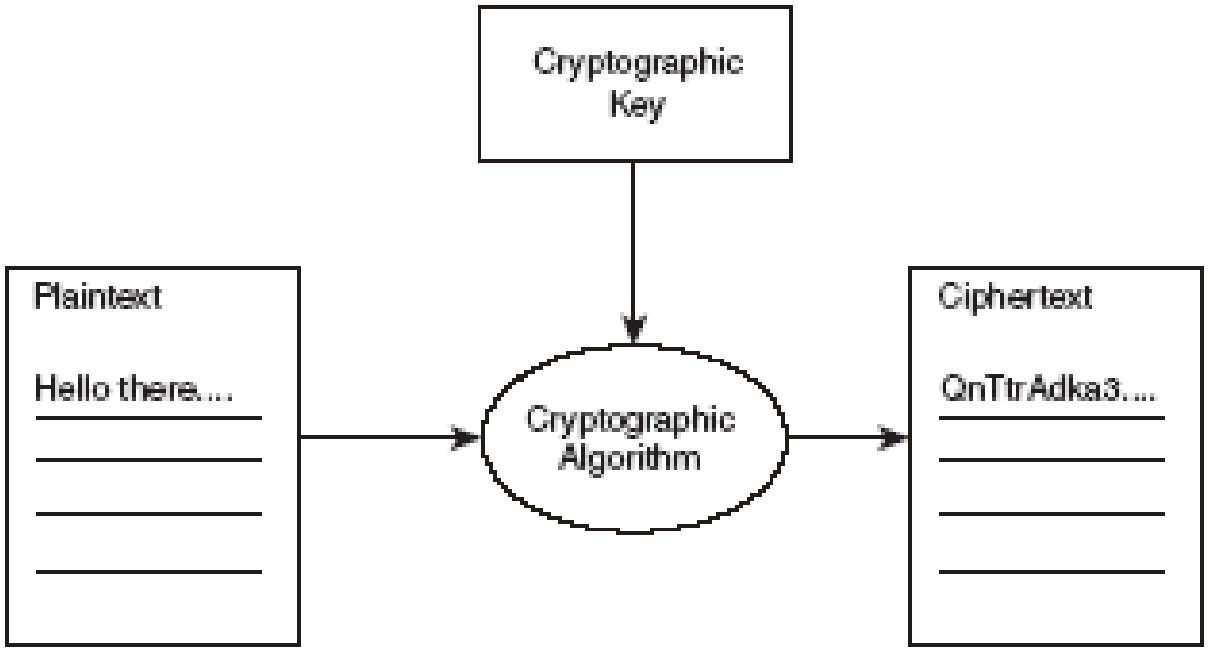


Figure 5.1
Basic encryption operation

Cryptographic Algorithms (continued)

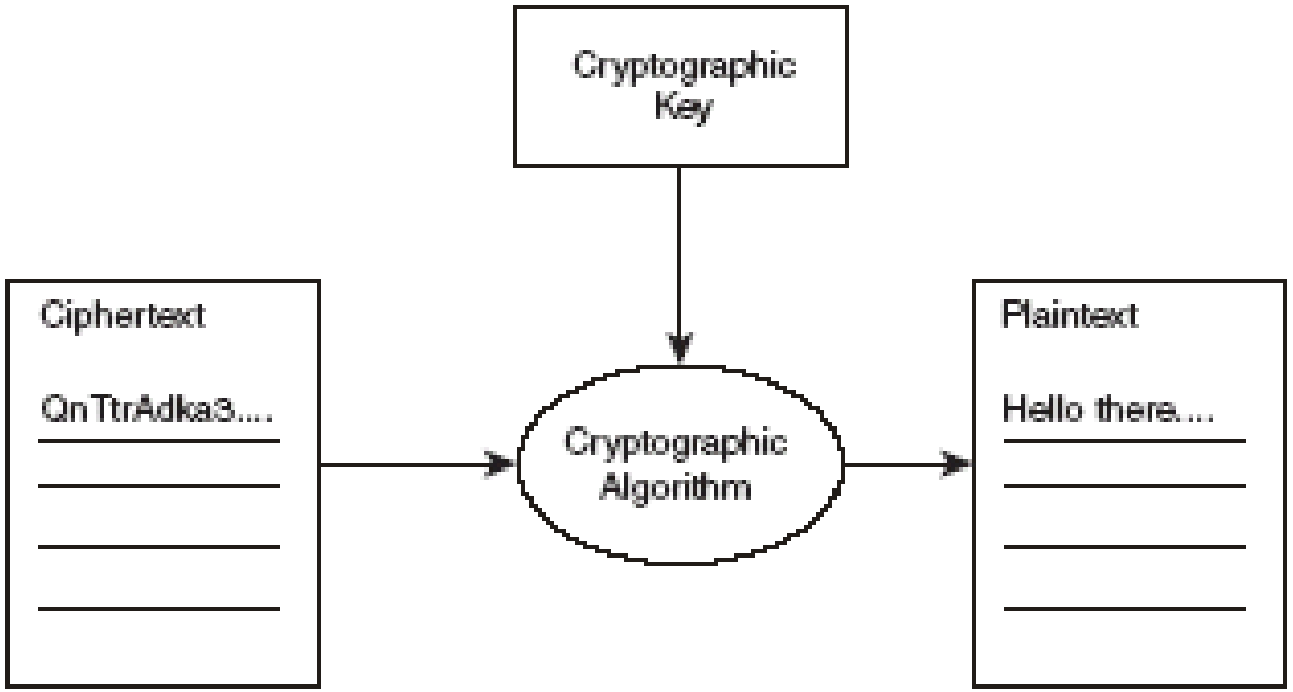


Figure 5.2

Basic decryption operation

Symmetric Algorithms

- Symmetry results from the sender and receiver using the same key
- Key is called ***shared secret key*** or ***secret key***
- Symmetric cryptosystems sometimes called ***secret key cryptosystems***
- Key length is a critical component of security

Key Length

- The longer the key, the greater the degree of protection
- A common attack against cryptosystems is the brute force attack
  - All possible keys are tried
  - Longer keys create an enormous number of possible combinations, frustrating brute force attacks
  - Formula used to compute the number of combinations is 2^n where n is the key length in bits

Key Length

TABLE 5.1 Possible Keys of a Given Length

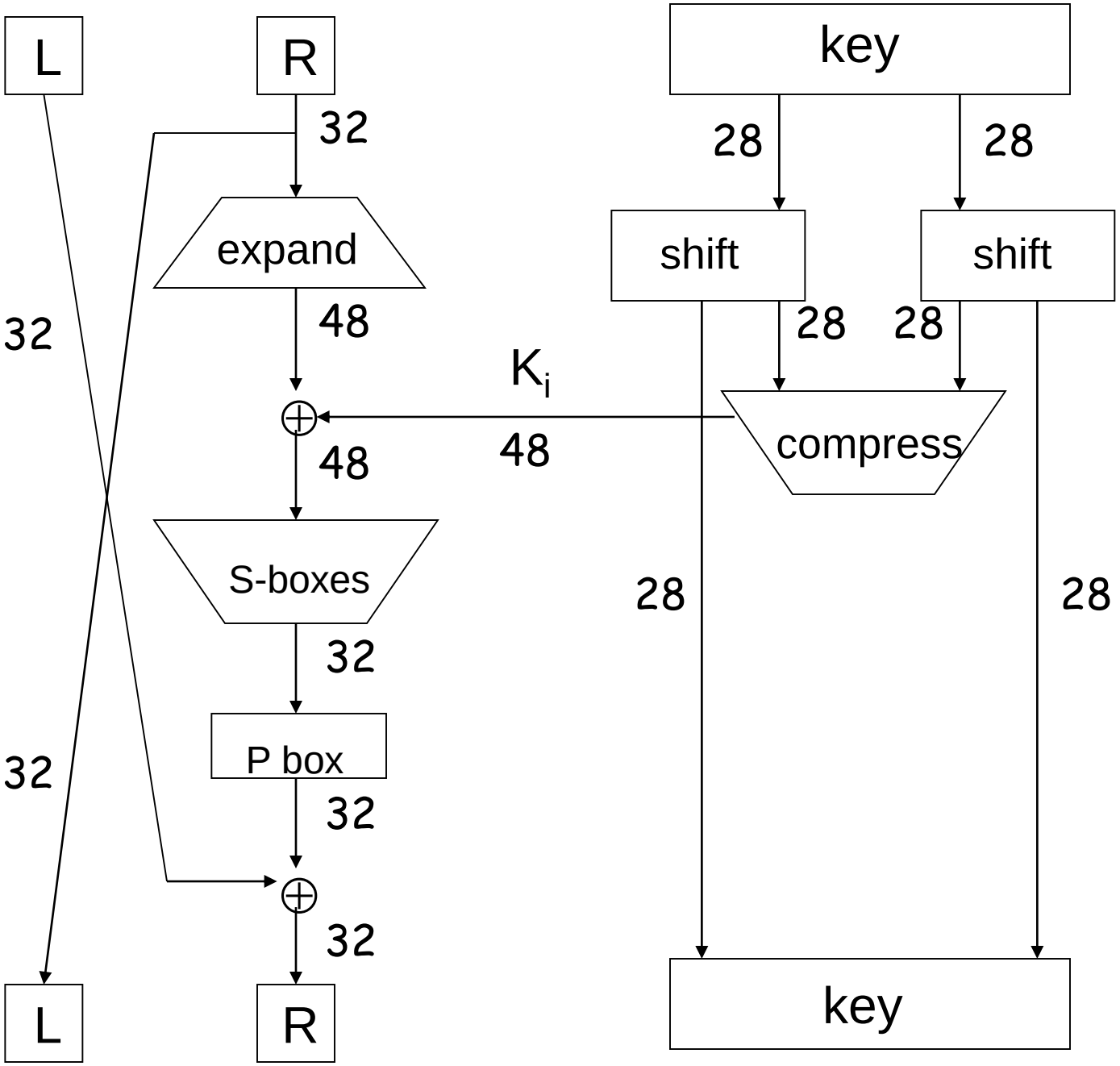
Key Length	Approximate Number of Possible Keys
56 bits	72,057,594,037,927,936
128 bits	3.40×10^{38}
256 bits	1.16×10^{77}
512 bits	1.34×10^{154}
1,024 bits	1.80×10^{308}
2,048 bits	3.23×10^{616}

Data Encryption Standard (DES)

- One of the most common symmetric cryptosystems
- Uses a 56-bit key with four modes of operation
  - Electronic codebook, ciphertext block chaining, output feedback, ciphertext feedback
- The DES algorithms are very flexible
- A fatal flaw
  - A 56-bit key is no longer considered strong enough to survive brute force attacks
- Current versions of DES use three separate iterations of DES encryption on each message
  - Triple DES (3DES)

DES Numerology

- DES is a Feistel cipher
  - 64 bit block length
  - 56 bit key length
  - 16 rounds
  - 48 bits of key used each round (subkey)
- Each round is simple (for a block cipher)
- Security depends primarily on "S-boxes"
  - Each S-boxes maps 6 bits to 4 bits

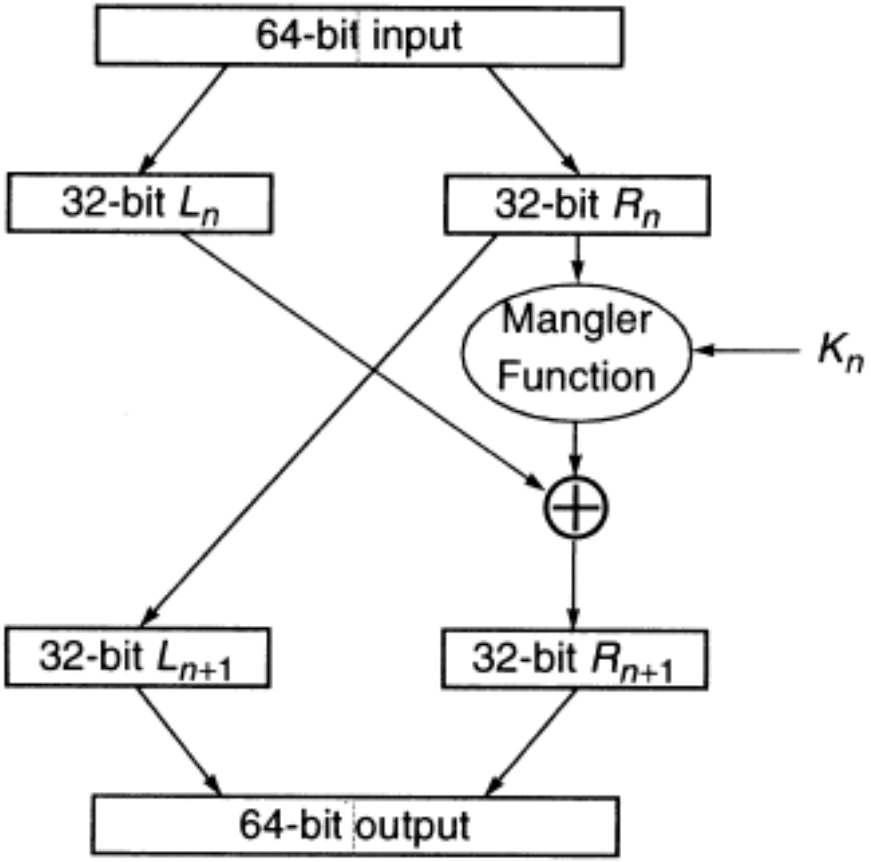


One Round of DES

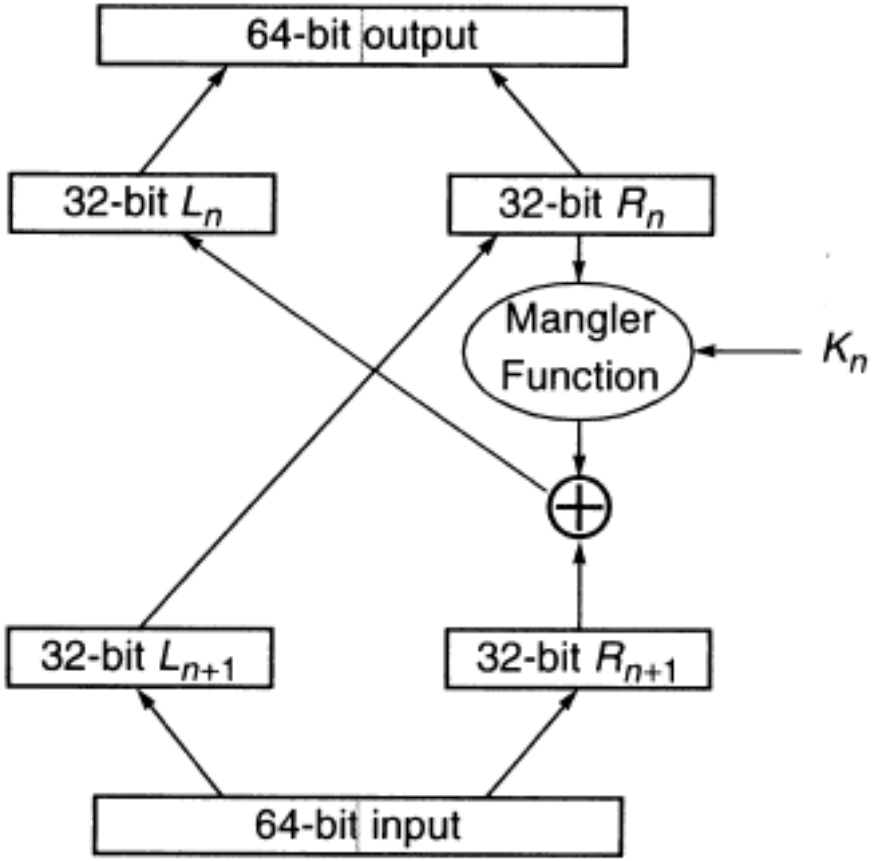
DES Mangler Function

- $L_n = R_{n-1}$
- $R_n = L_{n-1} \oplus (P(S\text{-}box(k_i \oplus E(R_{n-1}))))$

DES Decryption



Encryption



Decryption

DES Expansion Permutation

- Input 32 bits

```
 0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
```

- Output 48 bits

```
31  0  1  2  3  4  3  4  5  6  7  8
 7  8  9 10 11 12 11 12 13 14 15 16
15 16 17 18 19 20 19 20 21 22 23 24
23 24 25 26 27 28 27 28 29 30 31  0
```

DES S-box

- 8 "substitution boxes" or S-boxes
- Each S-box maps 6 bits to 4 bits
- S-box number 1

input bits (0,5)
↓ input bits (1,2,3,4)

	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
00	1110	0100	1101	0001	0010	1111	1011	1000	0011	1010	0110	1100	0101	1001	0000	0111
01	0000	1111	0111	0100	1110	0010	1101	0001	1010	0110	1100	1011	1001	0101	0011	1000
10	0100	0001	1110	1000	1101	0110	0010	1011	1111	1100	1001	0111	0011	1010	0101	0000
11	1111	1100	1000	0010	0100	1001	0001	0111	0101	1011	0011	1110	1010	0000	0110	1101

DES P-box

- Input 32 bits

```
 0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
```

- Output 32 bits

```
15  6 19 20 28 11 27 16  0 14 22 25  4 17 30  9
 1  7 23 13 31 26  2  8 18 12 29  5 21 10  3 24
```

DES Subkey

- 56 bit DES key, numbered 0,1,2,…,55
- Left half key bits, LK

```
49 42 35 28 21 14  7
 0 50 43 36 29 22 15
 8  1 51 44 37 30 23
16  9  2 52 45 38 31
```

- Right half key bits, RK

```
55 48 41 34 27 20 13
 6 54 47 40 33 26 19
12  5 53 46 39 32 25
18 11  4 24 17 10  3
```

DES Subkey

- For rounds `i=1,2,~,16`
  - Let LK = (LK circular shift left by r_i)
  - Let RK = (RK circular shift left by r_i)
  - Left half of subkey K_i is of LK bits

```
13 16 10 23  0  4  2 27 14  5 20  9
22 18 11  3 25  7 15  6 26 19 12  1
```

  - Right half of subkey K_i is RK bits

```
12 23  2  8 18 26  1 11 22 16  4 19
15 20 10 27  5 24 17 13 21  7  0  3
```

DES Subkey

- For rounds 1, 2, 9 and 16 the shift r_i is 1, and in all other rounds r_i is 2
- Bits 8,17,21,24 of LK omitted each round
- Bits 6,9,14,25 of RK omitted each round
- **Compression permutation** yields 48 bit subkey K_i from 56 bits of LK and RK
- **Key schedule** generates subkey

DES Last Word (Almost)

- An initial perm P before round 1
- Halves are swapped after last round
- A final permutation (inverse of P) is applied to (R_{16},L_{16}) to yield ciphertext
- None of these serve any security purpose

Data Encryption Standard (continued)

- 3DES provides an acceptably strong level of protection
- Variations of 3DES use either 2 or 3 keys
  - 3DES-EEE (encrypt-encrypt-encrypt) uses 3 keys
  - 3DES-EDE (encrypt-decrypt-encrypt) can use from 1 to 3 keys with different levels of protection

Advanced Encryption Standard (AES)

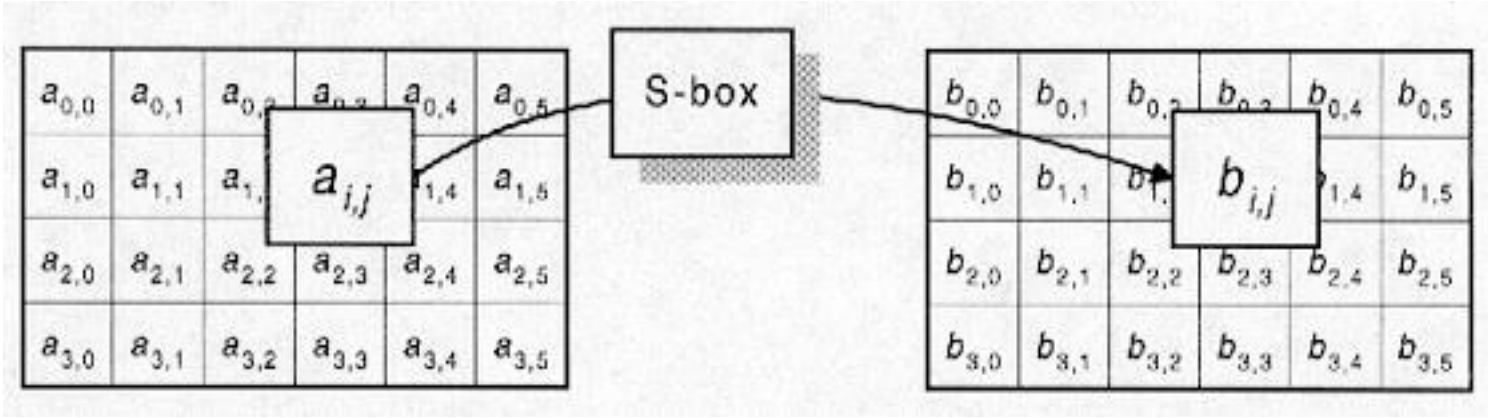
- Solicited in a competition sponsored by the National Institute of Standards (NIST)
- Candidate algorithms published their inner workings
- Winner was the Rijndael algorithm
- AES allows the user to select from 3 different key lengths
  - 128, 192, or 256 bits
  - The longer the key, the greater the security
- AES is gaining momentum, but the volume of applications that use DES makes conversion slow

AES Overview

- **Block size:** 128, 192 or 256 bits
- **Key length:** 128, 192 or 256 bits (independent of block size)
- 10 to 14 rounds (depends on key length)
- Each round uses 4 functions (in 3 "layers")
  - ByteSub (nonlinear layer)
  - ShiftRow (linear mixing layer)
  - MixColumn (nonlinear layer)
  - AddRoundKey (key addition layer)

AES ByteSub

- Assume 192 bit block, 4x6 bytes



- ByteSub is AES's "S-box"
- Can be viewed as nonlinear (but invertible) composition of two math operations

AES "S-box"

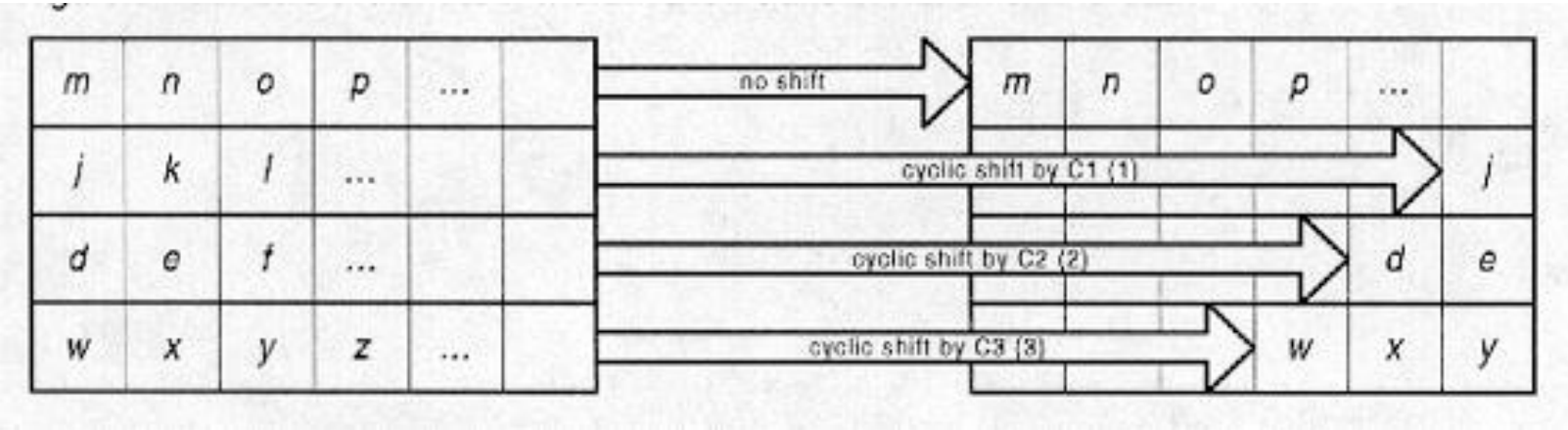
Last 4 bits of input

First 4 bits of input

	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0	63	7c	77	7b	f2	6b	6f	c5	30	01	67	2b	fe	d7	ab	76
1	ca	82	c9	7d	fa	59	47	f0	ad	d4	a2	af	9c	a4	72	c0
2	b7	fd	93	26	36	3f	f7	cc	34	a5	e5	f1	71	d8	31	15
3	04	c7	23	c3	18	96	05	9a	07	12	80	e2	eb	27	b2	75
4	09	83	2c	1a	1b	6e	5a	a0	52	3b	d6	b3	29	e3	2f	84
5	53	d1	00	ed	20	fc	b1	5b	6a	cb	be	39	4a	4c	58	cf
6	d0	ef	aa	fb	43	4d	33	85	45	f9	02	7f	50	3c	9f	a8
7	51	a3	40	8f	92	9d	38	f5	bc	b6	da	21	10	ff	f3	d2
8	cd	0c	13	ec	5f	97	44	17	c4	a7	7e	3d	64	5d	19	73
9	60	81	4f	dc	22	2a	90	88	46	ee	b8	14	de	5e	0b	db
a	e0	32	3a	0a	49	06	24	5c	c2	d3	ac	62	91	95	e4	79
b	e7	c8	37	6d	8d	d5	4e	a9	6c	56	f4	ea	65	7a	ae	08
c	ba	78	25	2e	1c	a6	b4	c6	e8	dd	74	1f	4b	bd	8b	8a
d	70	3e	b5	66	48	03	f6	0e	61	35	57	b9	86	c1	1d	9e
e	e1	f8	98	11	69	d9	8e	94	9b	1e	87	e9	ce	55	28	df
f	8c	a1	89	0d	bf	e6	42	68	41	99	2d	0f	b0	54	bb	16

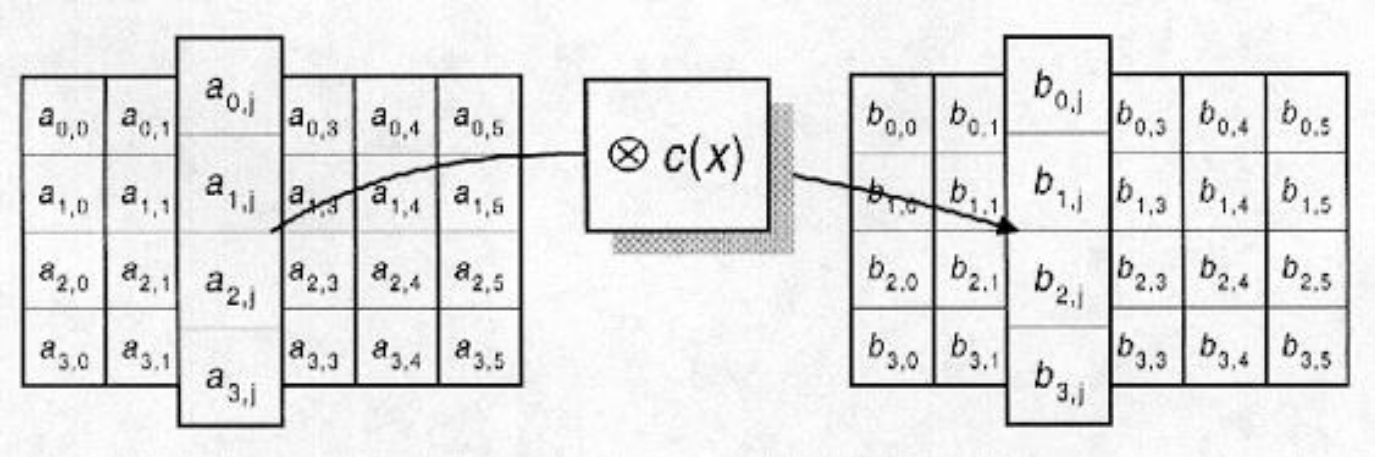
AES ShiftRow

- Cyclic shift rows



AES MixColumn

- Nonlinear, invertible operation applied to each column



- Implemented as a (big) lookup table

AES AddRoundKey

- XOR subkey with block

$$\begin{bmatrix} a_{00} & a_{01} & a_{02} & a_{03} & a_{04} & a_{05} \\ a_{10} & a_{11} & a_{12} & a_{13} & a_{14} & a_{15} \\ a_{20} & a_{21} & a_{22} & a_{23} & a_{24} & a_{25} \\ a_{30} & a_{31} & a_{32} & a_{33} & a_{34} & a_{35} \end{bmatrix} \oplus \begin{bmatrix} k_{00} & k_{01} & k_{02} & k_{03} & k_{04} & k_{05} \\ k_{10} & k_{11} & k_{12} & k_{13} & k_{14} & k_{15} \\ k_{20} & k_{21} & k_{22} & k_{23} & k_{24} & k_{25} \\ k_{30} & k_{31} & k_{32} & k_{33} & k_{34} & k_{35} \end{bmatrix} = \begin{bmatrix} b_{00} & b_{01} & b_{02} & b_{03} & b_{04} & b_{05} \\ b_{10} & b_{11} & b_{12} & b_{13} & b_{14} & b_{15} \\ b_{20} & b_{21} & b_{22} & b_{23} & b_{24} & b_{25} \\ b_{30} & b_{31} & b_{32} & b_{33} & b_{34} & b_{35} \end{bmatrix}$$

Block Subkey

- RoundKey (subkey) determined by **key schedule** algorithm

AES Decryption

- To decrypt, process must be invertible
- Inverse of MixAddRoundKey is easy, since $\oplus$ is its own inverse
- MixColumn is invertible (inverse is also implemented as a lookup table)
- Inverse of ShiftRow is easy (cyclic shift the other direction)
- ByteSub is invertible (inverse is also implemented as a lookup table)

Asymmetric Algorithms

- Differ from symmetric algorithms because sender and receiver use different keys
- Each user has a pair of keys
  - Public key and private key
  - Keys are mathematically related
  - Messages encrypted with public key can only be decrypted with private key
  - Public keys are freely distributed so that anyone can use them to encrypt a message
- Asymmetric algorithms are referred to as ***public key cryptosystems***

Asymmetric Algorithms Example

- Renee and Michael wish to communicate sensitive information
  - Renee and Michael share their public keys
  - When Renee sends a message to Michael, she encrypts it with Michael's public key
  - Only Michael can decrypt the message because decryption requires his private key, which he does not share with anyone

Asymmetric Algorithms (continued)

- Rivest, Shamir, Adelman algorithm (RSA)
  - One of the most well-known public key cryptosystems
  - Developed in the late 1970's
  - Relies on the fact that it is extremely difficult to factor large prime numbers
- Pretty Good Privacy (PGP)
  - A cross-platform solution
  - An implementation of several cryptographic algorithms (including RSA)
  - Supports management of a decentralized public key infrastructure

The Web of Trust

- Key exchange is a difficult problem
  - Before PGP, it was necessary to exchange keys offline
- PGP introduced the “web of trust” model
  - Allows users to rely on the judgment of others that a public key is authentic
- Four levels of trust
  - Implicit trust
  - Full trust
  - Marginal trust
  - Untrusted

Symmetric Versus Asymmetric Cryptosystems

- Choice between symmetric and asymmetric cryptosystems involves the number of keys that must be generated
  - Symmetric cryptosystems don't scale well
  - Asymmetric cryptosystems are slower than symmetric ones
  - Symmetric cryptosystems are excellent for securing the ends of a communication circuit such as a Virtual Private Network
  - Asymmetric cryptosystems are more practical when there are a large number of users

TABLE 5.2 Comparison of Symmetric and Asymmetric Cryptosystems

Symmetric Cryptosystems	Asymmetric Cryptosystems
Provide confidentiality among all participants who share the same secret key	Provide confidentiality between individual users of a cryptosystem
Provide integrity against modification by individuals who do not possess the secret key	Provide integrity against modification by anyone other than the sender of the message
Provide for authentication between two individuals when they are the only ones who possess the secret key	Provide for authentication of any individual user of the cryptosystem
Do not provide for nonrepudiation	Provide for nonrepudiation
Require shorter keys than asymmetric algorithms to achieve the same level of security	Require longer keys than symmetric algorithms to achieve the same level of security
Operate faster than asymmetric algorithms	Operate slower than symmetric algorithms
Are not easily scalable	Scale well to environments with large numbers of users
Do not facilitate the use of digital certificates	Lend themselves well to digital certificate hierarchies
Make the exchange of cryptographic keys difficult (often requiring offline exchange)	Allow for the exchange of public keys over otherwise insecure transmission media

Digital Signatures

- Add integrity and nonrepudiation functionality to cryptosystems
- Nonrepudiation can only be enforced with asymmetric algorithms
- Signature creation
  - A unique message digest is created by applying a hash function to the message
  - Variations of the Secure Hash (SHA) and MD Algorithms are commonly used
  - Sender encrypts the message digest with his/her private key

Digital Signatures (continued)

- Signature verification
  - Recipient decrypts the message and extracts the plaintext message and digital signature
  - Recipient applies same hash function to the message as that used by the sender to create a new message digest
  - Recipient decrypts the digital signature using the sender's public key to extract the sender's message digest
  - The recipient compares the two message digests
  - If the message digests match, signature is authentic
  - Non-matching signatures can be malicious but also can be due to transmission errors, etc.

Digital Certificates

- Digital certificates allow a third party to vouch for a digital signature
- The third party does the work to verify the identity of the sender
- Certification Authorities
  - The third parties that verify and certify the identity of a sender
  - Two of the most common CAs are VeriSign and Thawte

Digital Certificates (continued)

- Certificate generation
  - Sender selects and pays a CA
  - Sender submits required information for CA to verify their identity
  - CA issues a digital certificate following the X.509 standard
  - CA signs the digital certificate
- Certificate verification
  - A digital certificate can be used to securely transmit the sender's public key to any entity that trusts the CA and accepts the certificate

Summary

- Goals of cryptography are confidentiality, integrity, nonrepudiation, and authentication
- General steps in cryptography are to
  - Create a plaintext message
  - Use a cryptographic key and algorithm to produce a ciphertext message
  - Apply the same or a related key and algorithm to the ciphertext message
  - Recreate the original plaintext message
- There are two types of cryptographic algorithms
  - Symmetric (uses a shared secret key)
  - Asymmetric (uses a public and private key pair)

Summary

- Digital signatures are used to add integrity and nonrepudiation functionality to cryptosystems
- Digital signatures are created using hash functions applied to the message to create a message digest that is then encrypted
- Digital certificates allow a third party Certificate Authority to verify the identity of a sender who may not be well known to the recipient
- A digital certificate is a copy of a user's public key that has been digitally signed by a Certificate Authority.