Cryptography

- Introduction to Cryptography
- Substitution Ciphers
- Transposition Ciphers
- One-Time Pads
- Two Fundamental Cryptographic Principles

Need for Security

Adversary	Goal
Student	To have fun snooping on people's e-mail
Cracker	To test out someone's security system; steal data
Sales rep	To claim to represent all of Europe, not just Andorra
Businessman	To discover a competitor's strategic marketing plan
Ex-employee	To get revenge for being fired
Accountant	To embezzle money from a company
Stockbroker	To deny a promise made to a customer by e-mail
Con man	To steal credit card numbers for sale
Spy	To learn an enemy's military or industrial secrets
Terrorist	To steal germ warfare secrets

Some people who cause security problems and why.

Need for Security

Secrecy:

aka confidentiality, keeping information from unauthorized users.

Authentication:

Determine participants before disclosing sensitive information

Nonrepudiation:

aka signature verification

Integrity control:

Assurance that received data is from genuine senders, data is original and has not been altered in transit

Link Encryption (1)

Where, in the protocol stack does security belong?

Encryption on a layer by layer basis

Physical layer:

Foiling wiretapping by enclosing transmission lines in sealed tubes containing an inert gas.

Data link layer:

Packets encrypted as they leave one machine and decrypted upon arrival at the receiving machine.

Breaks down when packets have to traverse multiple routers

Does not allow specific sessions to be encrypted and not others between same two end points.

Link Encryption (2)

Network layer:

Firewalls may be installed to keep good packets and prevent unauthorized packets from traversing. IP security functions in this manner

Transport layer:

Entire connections may be encrypted end-to-end

For maximum security, end-to-end security is required

Application layer:

Only place where user authentication and nonrepudiation may be handled

Most Prevalent Fraud

Lax security procedures

Incompetent employees

Implementation bugs

Social engineering (spoofing or tricking users to disclose private information)

Human errors and omissions

Cryptography does not solve or prevent any of the above

Ciphers and codes

Cipher-

Character-for-character or bit-for-bit transformation without regard to the linguistic structure of the message

Code-

Replace one word or group of words with another word, word group or symbol.

Example – Most successful code ever devised –

Used by U.S. military during WWII. Navajo Indians talking to each other using specific Navajo words for military terms. Navajo language is highly tonal, exceedingly complex, and has no written form.

The Japanese were never able to break the code yet the U.S. broke the Japanese code.

Early Cryptography

Military

Diplomatic corps

Diarists

Lovers

Before computers –

biggest constraint was the ability of the “code clerk” to perform the necessary transformations.

Difficult to switch cryptography methods

Capture of code clerk by the enemy

Cryptology

Cryptanalysis –

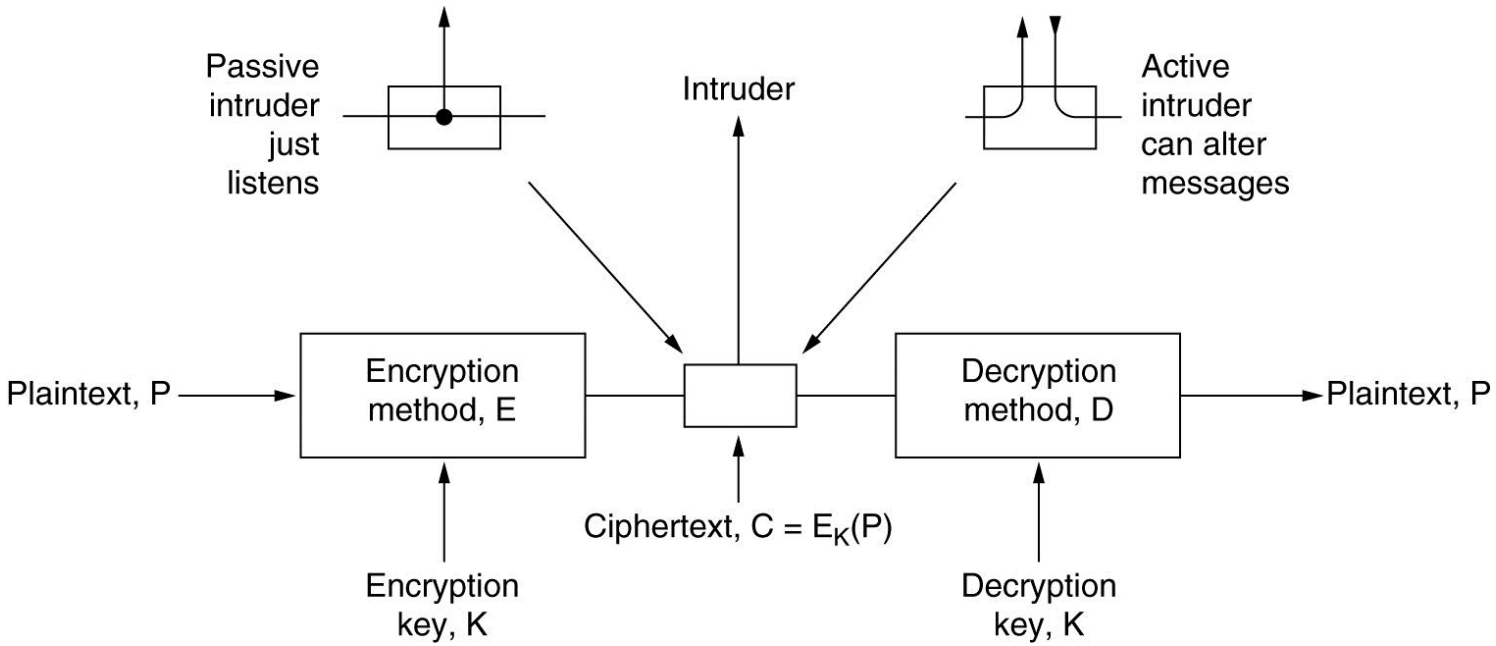
Art of breaking ciphers (aka hacking)

Cryptography –

Art of devising ciphers

Cryptanalysis + Cryptography = Cryptology

An Introduction to Cryptography



The encryption model (for a symmetric-key cipher).

Kerckhoff's Principle

All algorithms must be public; only the keys are secret

Enables a quick and easy change to ciphers

Does not require retraining of "code clerks"

Cryptographer get free consulting from a large number of academic cryptologists eager to break the system

Algorithms that withstand the time, are viewed as solid

The longer the key, the more secure the cipher

Cryptanalysis Variations

Ciphertext-only:

Cryptanalyst has a quantity of cipher text and now plain text

Known plaintext:

Cryptanalyst has a quantity of cipher text and plain-text

Chosen plaintext:

Cryptanalyst has the ability to encrypt plaintext of his own choosing

Categories of Encryption (1)

Substitution Ciphers:

Each letter or group of letters is replaced by another letter or group of letters. e.g Caesar Cipher, each plain text letter substituted with the letter two letters later in the alphabet:

```
attack
DWWDFN
```

Easily broken. But when substitution table is a random sequence of mutually exclusive letters, (i.e. 26! possible different ‘keys’) then much more difficult.

Weakness is frequency of letter use.

Letter-for-letter substitution know as *monoaphabetic substitution cipher*

Categories of Encryption (2)

Transposition Ciphers:

Substitution ciphers preserve the plaintext symbol order, but disguises them

Transposition Ciphers reorder the plaintext symbols but does not disguise them

Cryptanalyst must know he is working with a transposition cipher

Transposition Ciphers

M	E	G	A	B	U	C	K
7	4	5	1	2	8	3	6
p	l	e	a	s	e	t	r
a	n	s	f	e	r	o	n
e	m	i	l	l	i	o	n
d	o	l	l	a	r	s	t
o	m	y	s	w	i	s	s
b	a	n	k	a	c	c	o
u	n	t	s	i	x	t	w
o	t	w	o	a	b	c	d

Plaintext

pleasetransferonemilliondollarsto
myswissbankaccountsixtwotwo

Ciphertext

AFLLSKSOSELAWAIATOOSSCTCLNMOMANT
ESILYNTWRNNTSOWDPAEDOBUOERIRICXB

A transposition cipher.

One-Time Pads

Message 1:	1001001	0100000	1101100	1101111	1110110	1100101	0100000	1111001	1101111	1110101	0101110
Pad 1:	1010010	1001011	1110010	1010101	1010010	1100011	0001011	0101010	1010111	1100110	0101011
Ciphertext:	0011011	1101011	0011110	0111010	0100100	0000110	0101011	1010011	0111000	0010011	0000101
Pad 2:	1011110	0000111	1101000	1010011	1010111	0100110	1000111	0111010	1001110	1110110	1110110
Plaintext 2:	1000101	1101100	1110110	1101001	1110011	0100000	1101100	1101001	1110110	1100101	1110011

The use of a one-time pad for encryption and the possibility of getting any possible plaintext from the ciphertext by the use of some other pad.

Two Fundamental Cryptographic Principles

Redundancy:

- Messages (plaintext) must contain some redundancy
  - Foils random generated plaintext

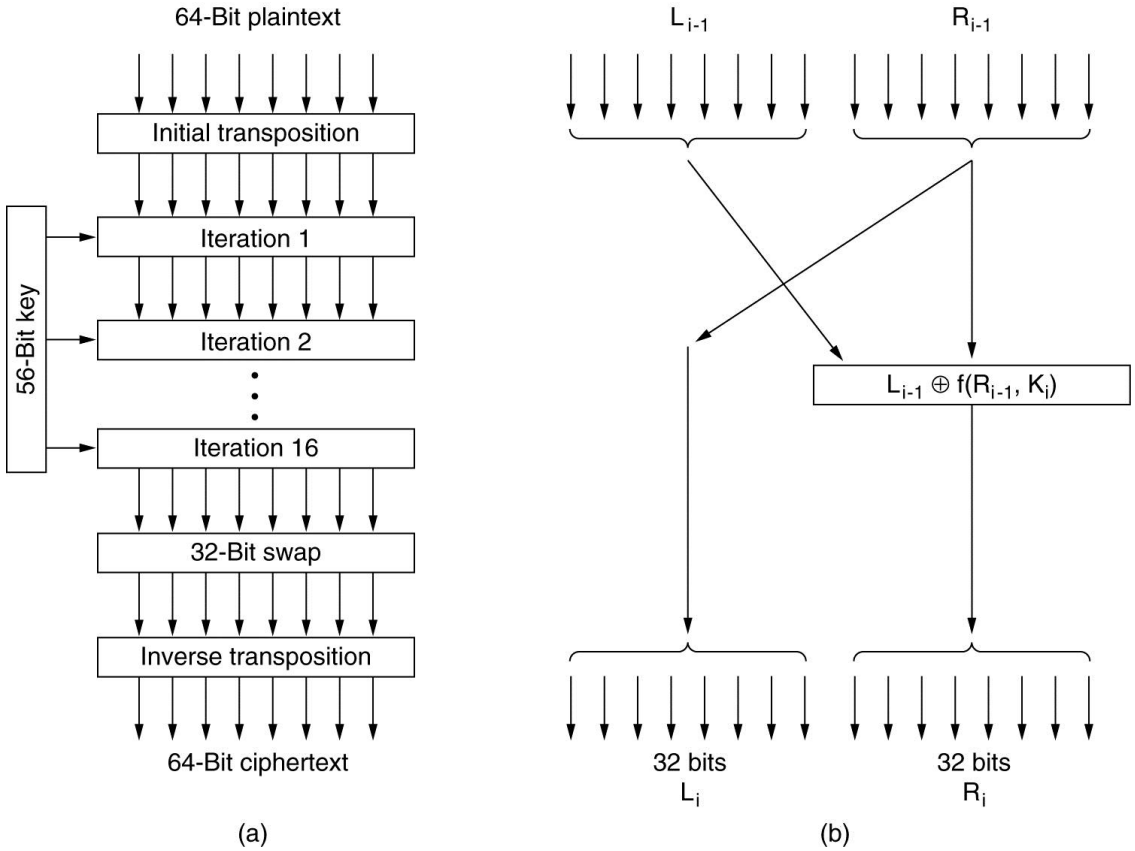
Freshness:

- Messages (plaintext) must not be aged from when it was transmitted.
  - Foils replay attacks

Symmetric-Key Algorithms

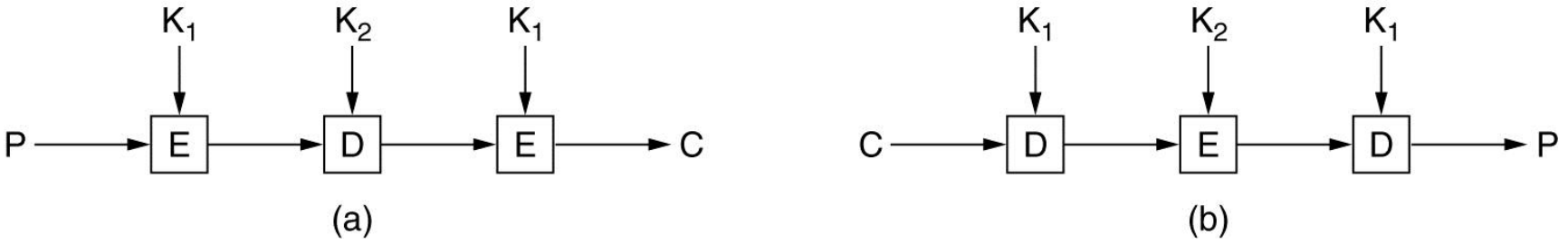
- DES – The Data Encryption Standard
- AES – The Advanced Encryption Standard
- Cipher Modes
- Other Ciphers
- Cryptanalysis

Data Encryption Standard



The data encryption standard. (a) General outline.
(b) Detail of one iteration. The circled + means exclusive OR.

Triple DES



(a) Triple encryption using DES. (b) Decryption.

AES – The Advanced Encryption Standard

Rules for AES proposals

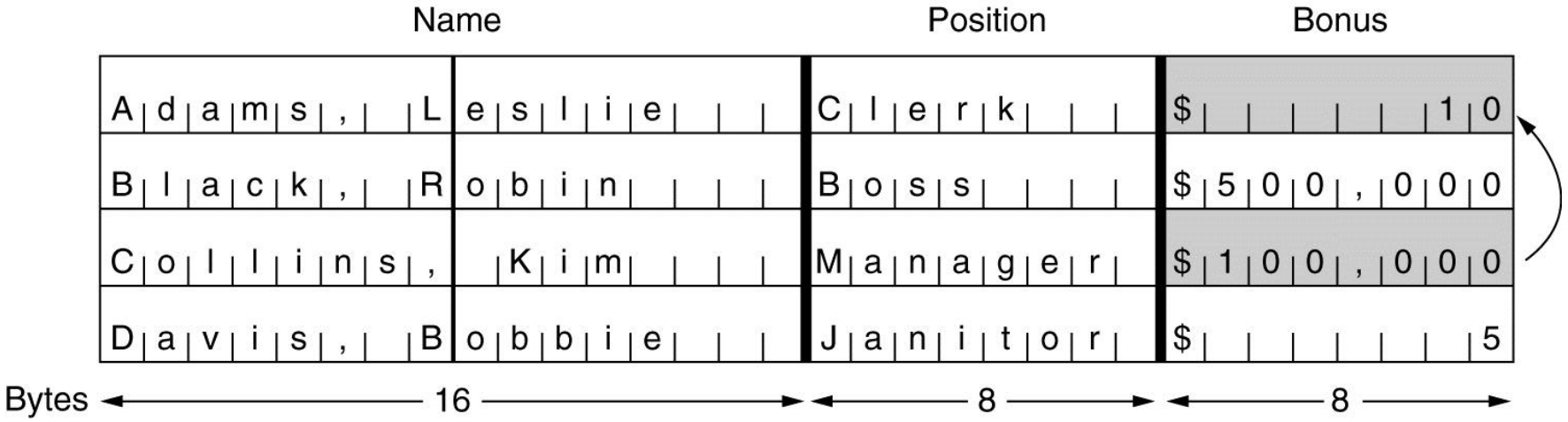
1. The algorithm must be a symmetric block cipher.
2. The full design must be public.
3. Key lengths of 128, 192, and 256 bits supported.
4. Both software and hardware implementations required
5. The algorithm must be public or licensed on nondiscriminatory terms.

AES (2)

An outline of Rijndael.

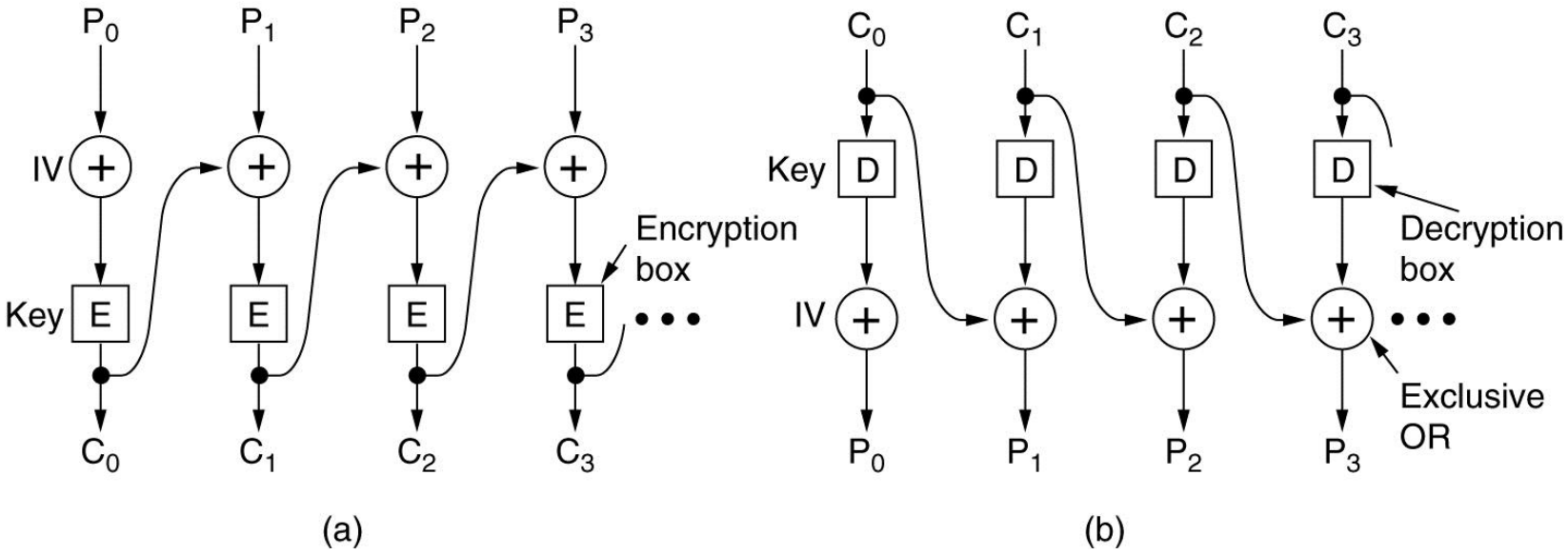
```
#define LENGTH 16                               /* # bytes in data block or key */
#define NROWS 4                                 /* number of rows in state */
#define NCOLS 4                                 /* number of columns in state */
#define ROUNDS 10                               /* number of iterations */
typedef unsigned char byte;                     /* unsigned 8-bit integer */

rijndael(byte plaintext[LENGTH], byte ciphertext[LENGTH], byte key[LENGTH])
{
  int r;                                        /* loop index */
  byte state[NROWS][NCOLS];                     /* current state */
  struct {byte k[NROWS][NCOLS];} rk[ROUNDS + 1];      /* round keys */

  expand_key(key, rk);                          /* construct the round keys */
  copy_plaintext_to_state(state, plaintext);    /* init current state */
  xor_roundkey_into_state(state, rk[0]);        /* XOR key into state */

  for (r = 1; r <= ROUNDS; r++) {
        substitute(state);                      /* apply S-box to each byte */
        rotate_rows(state);                     /* rotate row i by i bytes */
        if (r < ROUNDS) mix_columns(state);     /* mix function */
        xor_roundkey_into_state(state, rk[r]);  /* XOR key into state */
  }
  copy_state_to_ciphertext(ciphertext, state);  /* return result */
}
```

Electronic Code Book Mode



The plaintext of a file encrypted as 16 DES blocks.

Cipher Block Chaining Mode



Cipher block chaining. (a) Encryption. (b) Decryption.

Cryptanalysis (1)

Cipher	Author	Key length	Comments
Blowfish	Bruce Schneier	1–448 bits	Old and slow
DES	IBM	56 bits	Too weak to use now
IDEA	Massey and Xuejia	128 bits	Good, but patented
RC4	Ronald Rivest	1–2048 bits	Caution: some keys are weak
RC5	Ronald Rivest	128–256 bits	Good, but patented
Rijndael	Daemen and Rijmen	128–256 bits	Best choice
Serpent	Anderson, Biham, Knudsen	128–256 bits	Very strong
Triple DES	IBM	168 bits	Second best choice
Twofish	Bruce Schneier	128–256 bits	Very strong; widely used

Some common symmetric-key cryptographic algorithms.

Cryptanalysis (2)

Differential Cryptanalysis:

May be used to attack any bloc cipher

Begin with 2 plaintext blocks, differing in only a small number of bits

Watch carefully what happens on each internal iteration of encryption. Look for bit patterns that are more common than others. May lead to *probabilistic attacks*.

Cryptanalysis (3)

Linear Cryptanalysis:

Can break DES with only 2^{43rd} known plaintexts.

XOR certain bits in plaintext with ciphertext. Done repeatedly, half the bits should be 1 and half 0. Ciphers, however, introduce biases in one direction or another. This bias then used to reduce work factor.

Cryptanalysis (4)

Electrical Power Consumption Analysis Cryptanalysis:

Typically, computers use ~3 volts to represent a 1 bit, and 0 volts for a 0 bit.

Attacker replaces *n*-GHz clock with a slow (e.g. 100-Hz) clock and precisely monitors power consumed by each machine instruction

From this data, deducing the key has become relatively easy!

This cryptanalysis defeated only by carefully encoding the algorithms with machine level instructions.

Cryptanalysis (5)

Timing analysis cryptanalysis:

Cryptographic algorithms are full of '`if`', '`then`' and '`else`' statements. By slowing the clock, and determining amounts of time for each step, the round keys may be deduced.

Once all round keys are known, the original key is easily computed

Public-Key Algorithms (1)

- RSA
- Other Public-Key Algorithms

Public-Key Algorithms (2)

Solves problem of symmetric key dissemination

Encryption key is significantly different from decryption key, so that one cannot be derived from the other

Public-Key Algorithms (3)

Requirements:

1. D(E(P)) = P

2. Exceedingly difficult to deduce D from E

3. E cannot be broken by a chosen plaintext attack

E ➔ Keyed encryption algorithm

D ➔ Keyed decryption algorithm

P ➔ Plaintext

RSA
based on number theory principles

- Choose two large primes, p and q (typically 1024 bits)
- Computer $n = p \times q$ and $z = (p-1) \times (q-1)$
- Choose a number relatively prime to z and call it d
- Find e such that $e \times d = 1 \bmod z$

RSA

Plaintext (P)			Ciphertext (C)		After decryption	
Symbolic	Numeric	P^3	P^3 (mod 33)	C^7	C^7 (mod 33)	Symbolic
S	19	6859	28	13492928512	19	S
U	21	9261	21	1801088541	21	U
Z	26	17576	20	1280000000	26	Z
A	01	1	1	1	01	A
N	14	2744	5	78125	14	N
N	14	2744	5	78125	14	N
E	05	125	26	8031810176	05	E

Sender's computation

Receiver's computation

$p = 3$
$q = 11$
$n = 33$
$z = 20$
$d = 7$

An example of the RSA algorithm.