MCA : MCA23203

CRYPTOGRAPHY & NETWORK SECURITY

UNIT - 1

Unit-1: Introduction to Cryptography and Block Ciphers

Introduction to Cryptography- Conventional Encryption: Conventional encryption model- classical encryption techniques- substitution ciphers and transposition ciphers cryptanalysis- steganography- stream and block ciphers- block ciphers principles- data encryption standard (DES)- DES Encryption and Decryption – DES example – Strength of DES – AES Structure and Transformation functions.

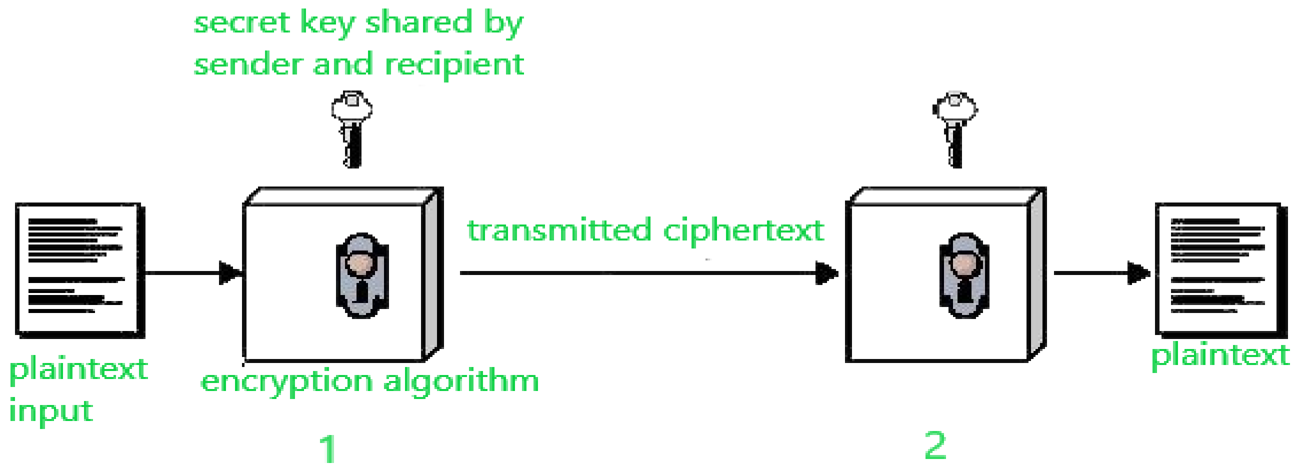
Introduction to Cryptography

What is Cryptography

- Cryptography
  - In a narrow sense
    - Mangling information into apparent unintelligibility
    - Allowing a secret method of un-mangling
  - In a broader sense
    - Mathematical techniques related to information security
    - About secure communication in the presence of adversaries
- Cryptanalysis
  - The study of methods for obtaining the meaning of encrypted information without accessing the secret information
- Cryptology
  - Cryptography + cryptanalysis

Conventional Encryption Model

- **Conventional encryption** is a cryptographic system that uses the same key used by the sender to encrypt the message and by the receiver to decrypt the message.
- It was the only type of encryption in use prior to the development of public-key encryption.



Conventional Encryption Model

- Suppose A wants to send a message to B, that message is called plaintext.
- Now, to avoid hackers reading plaintext, the plaintext is encrypted using an algorithm and a secret key (at 1).
- This encrypted plaintext is called ciphertext. Using the same secret key and encryption algorithm run in reverse(at 2),
- B can get plaintext of A, and thus the message is read and security is maintained.
- **Conventional encryption has mainly 5 ingredients :**

1. **Plain text**
2. **Encryption algorithm**
3. **Secret key**
4. Ciphertext
5. **Decryption algorithm**

Conventional Encryption Techniques

Symmetric Encryption

- conventional / secret-key / single-key
- Sender and recipient share a common key
- All classical encryption algorithms are secret-key-based
- Was the only type prior to the invention of public-key in 1970's
- By far most widely used

Conventional Encryption Principles

- An encryption scheme has five ingredients:
  - Plaintext
  - Encryption algorithm
  - Secret key
  - Ciphertext
  - Decryption algorithm
- Security depends on the secrecy of the key, not the secrecy of the algorithm

Some Basic Terminology

- Cipher
  - Algorithm for transforming plaintext to ciphertext
- Encipher (encrypt)
  - Converting plaintext to ciphertext
- Decipher (decrypt)
  - Recovering ciphertext from plaintext
- Cryptography
  - Study of encryption principles/methods
- Cryptanalysis (codebreaking)
  - Study of principles/ methods of deciphering ciphertext without knowing key
- Cryptology
  - Field of both cryptography and cryptanalysis

Conventional Encryption Principles

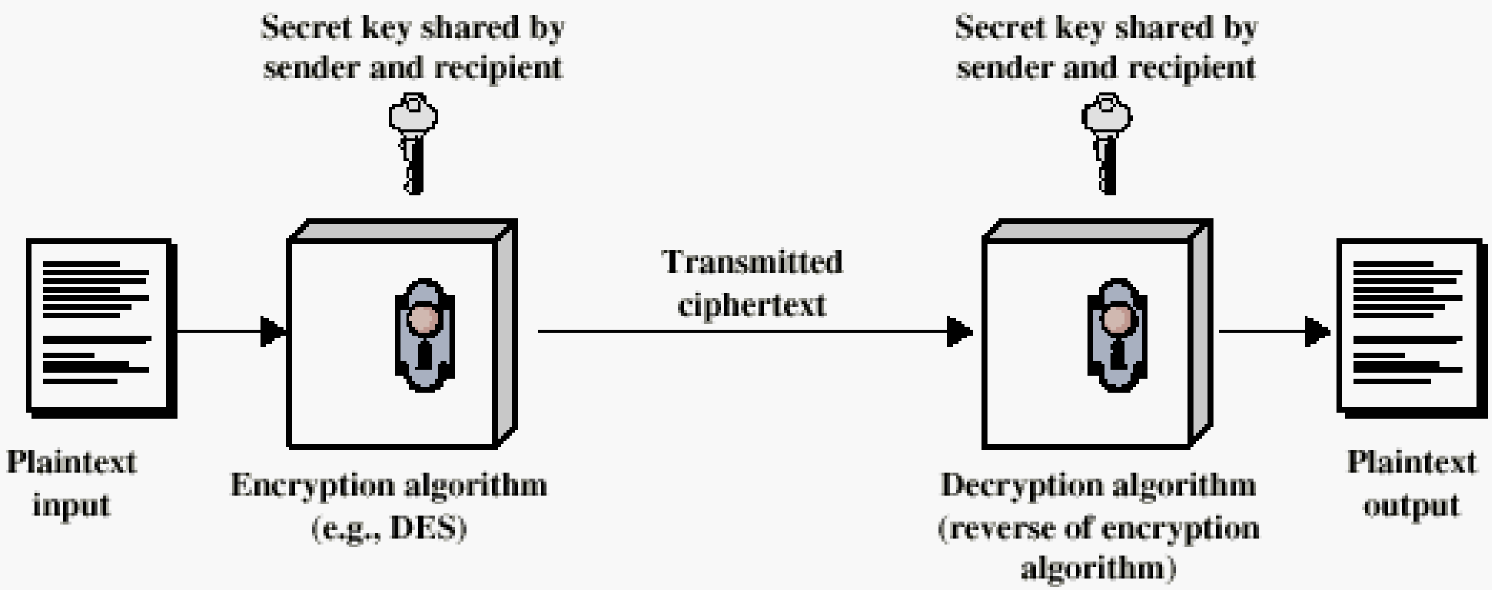


Figure 2.1 Simplified Model of Conventional Encryption

Requirements

- Two requirements for secure use of symmetric encryption:
  - A strong encryption algorithm
  - A secret key known only to sender and receiver
- Mathematically we have:

  $Y = \mathrm{E}_K(X)$

  $X = \mathrm{D}_K(Y)$
- Assume encryption algorithm is known
- Implies a secure channel to distribute key

Cryptography

- Characterize cryptographic systems by:
  - Type of encryption operations used
    - Substitution / transposition / product
      - Some examples will be discussed later
  - Number of keys used
    - Single-key or secret / two-key or public
  - Way in which plaintext is processed
    - Block / stream

Cryptanalysis

- Objective
  - Recover key not just message
- General approaches:
  - Cryptanalytic attack
  - Brute-force attack

Cryptanalytic Attacks

- **Ciphertext only**
  - Only know algorithm & ciphertext, is statistical, know or can identify plaintext
- **Known plaintext**
  - Know/suspect plaintext & ciphertext
- **Chosen plaintext**
  - Select plaintext and obtain ciphertext
- **Chosen ciphertext**
  - Select ciphertext and obtain plaintext
- **Chosen text**
  - Select plaintext or ciphertext to en/decrypt

More Definitions

- **Unconditional security**
  - No matter how much computer power or time is available, the cipher cannot be broken since the ciphertext provides insufficient information to uniquely determine the corresponding plaintext
- **Computational security**
  - Given limited computing resources (e.g. time needed for calculations is greater than the age of universe), the cipher cannot be broken

Brute Force Search (Exhaustive Key Search)

- Always possible to simply try every key
- Most basic attack, proportional to key size
- Assume either know / recognize plaintext

Average time (1/2)

Key Size (bits)	Number of Alternative Keys	Time required at 1 decryption/µs	Time required at 10^6 decryptions/µs
32	$2^{32} = 4.3 \times 10^9$		
56	$2^{56} = 7.2 \times 10^{16}$		
128	$2^{128} = 3.4 \times 10^{38}$	2^{127} µs $= 5.4 \times 10^{24}$ years	5.4×10^{18} years
168	$2^{168} = 3.7 \times 10^{50}$	2^{167} µs $= 5.9 \times 10^{36}$ years	5.9×10^{30} years
26 letters (permutation)	$26! = 4 \times 10^{26}$	2×10^{26} µs $= 6.4 \times 10^{12}$ years	6.4×10^6 years

Classical Substitution Ciphers

- Letters of plaintext are replaced by other letters or numbers or symbols
  - A popular TV show?
- If plaintext is viewed as a sequence of bits, then substitution involves replacing plaintext bit patterns with ciphertext bit patterns

Caesar Cipher

- Earliest known substitution cipher
  - By Julius Caesar
- First attested use in military affairs
- Replaces each letter by the 3rd letter following
- Example:

```
meet me after the toga party
PHHW PH DIWHU WKH WRJD SDUWB
```

Caesar Cipher

- Define transformation (mapping scheme) as:

  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
  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

- Assign each letter a number

  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
  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

- Mathematical form of Caesar cipher:

  $c = \mathrm{E}(p) = (p + k) \bmod (26)$

  $p = \mathrm{D}(c) = (c - k) \bmod (26)$

One mapping scheme → one key/cipher

How many possible keys/ciphers are there?

Cryptanalysis of Caesar Cipher

- Only have 26 possible ciphers
  - Map A to A, B,…, or Z
- Could simply try each in turn
- A **brute force search**
  - Given ciphertext, just try all shifts of letters
  - Need to recognize when have plaintext
    - e.g. break ciphertext “GCUA VQ DTGCM”
    - One student works on one cipher each

Monoalphabetic Cipher

- Rather than just shifting the alphabet
- Could shuffle (jumble) the letters arbitrarily
- Each plaintext letter maps to a different random ciphertext letter
- Hence key is 26 letters long

```
Plain: 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
Cipher:D K V Q F I B J W P E S C X H T M Y A
       U O L R G Z N


Plaintext:  ifwewishtoreplaceletters
Ciphertext: WIRFRWAJUHYFTSDVFSFUUFYA
```

- How many possible keys in total?

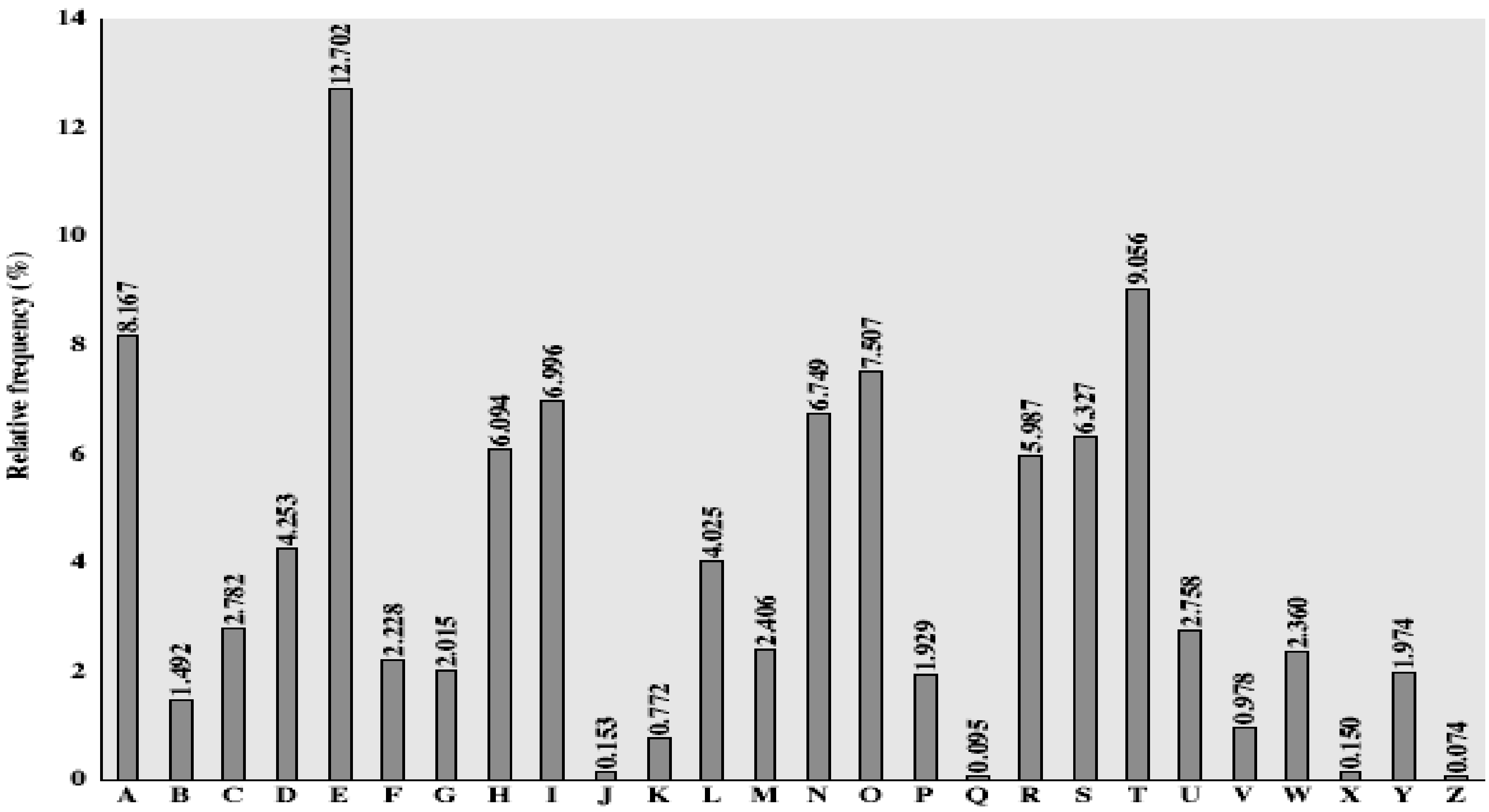
Monoalphabetic Cipher Security

- A total of 26! = 4^1026 keys
- With so many keys, one might think it is secure
- But would be **!!!WRONG!!!**
- Problem is language characteristics

Language Redundancy and Cryptanalysis

- Human languages are **redundant**
  - e.g. "th lrd s m shphrd shll nt wnt"
- Letters are not equally commonly used
- In English
  - E is by far the most common letter
  - Followed by T, R, N, I, O, A, S
  - Other letters like Z, J, K, Q, X are fairly rare
  - Which set of characters are most commonly used in Chinese?
- Have tables of single, double & triple letter frequencies for various languages

English Letter Frequencies



Use in Cryptanalysis

- Key concept
  - Monoalphabetic substitution ciphers do not change relative letter frequencies
  - Discovered by Arabian scientists in 9th century
  - Calculate letter frequencies for ciphertext
  - Compare counts/plots against known values
- Caesar cipher looks for common peaks/troughs
  - Peaks at: A-E-I triple, NO pair, RST triple
  - Troughs at: JK, X-Z
- Monoalphabetic must identify each letter
  - Tables of common double/triple letters help

Example Cryptanalysis

- Given ciphertext:

  UZQSOVUOHXMOPVGPOZPEVSGZWSZOPFPESXUDBMETSXAIZ
  VUEPHZHMDZSHZOWSFPAPPDTSVPQUZWYMXUZUHSX
  EPYEPOPDZSZUFPOMBZWPFUPZHMDJUDTMOHMQ

- Count relative letter frequencies (see text)
  - Guess which two individual letters are for e & t (with the highest frequencies)?
  - Proceed with trial and error finally get:

Many other substitution methods…

- Playfair Cipher
- Polyalphabetic Ciphers
- Vigenère Cipher
- Aids
- Kasiski Method
- Autokey Cipher
- One-Time Pad

Transposition Ciphers

- Now consider classical **transposition** or **permutation** ciphers
- These hide the message
  - By rearranging the letter order
  - Without altering the actual letters used
- Does the cipher text have the same frequency distribution as the original text?
- Is it subject to the language frequency cryptanalysis?

Rail Fence Cipher

- Write message letters out diagonally over a number of rows
- Then read off cipher row by row
  - E.g. write message out as:

```
m e m a t r h t g p r y
 e t e f e t e o a a t
```

- Giving ciphertext

```
MEMATRHTGPRY ETEFETEOAAT
```

Row Transposition Ciphers

- A more complex transposition
- Write letters of message out in rows over a specified number of columns
- Then reorder the columns according to some key before reading off the rows

Key:	3	4	2	1	5	6	7
Plaintext:	a	t	t	a	c	k	p
	o	s	t	p	o	n	e
	d	u	n	t	i	l	t
	w	o	a	m	x	y	z

Read starting from the 3rd column, then the 4th one, and so on.

Ciphertext: TTNAAPTMTSUOAODWCOIXKNLYPETZ

Method (algorithm) is not a secret, but key is the key!

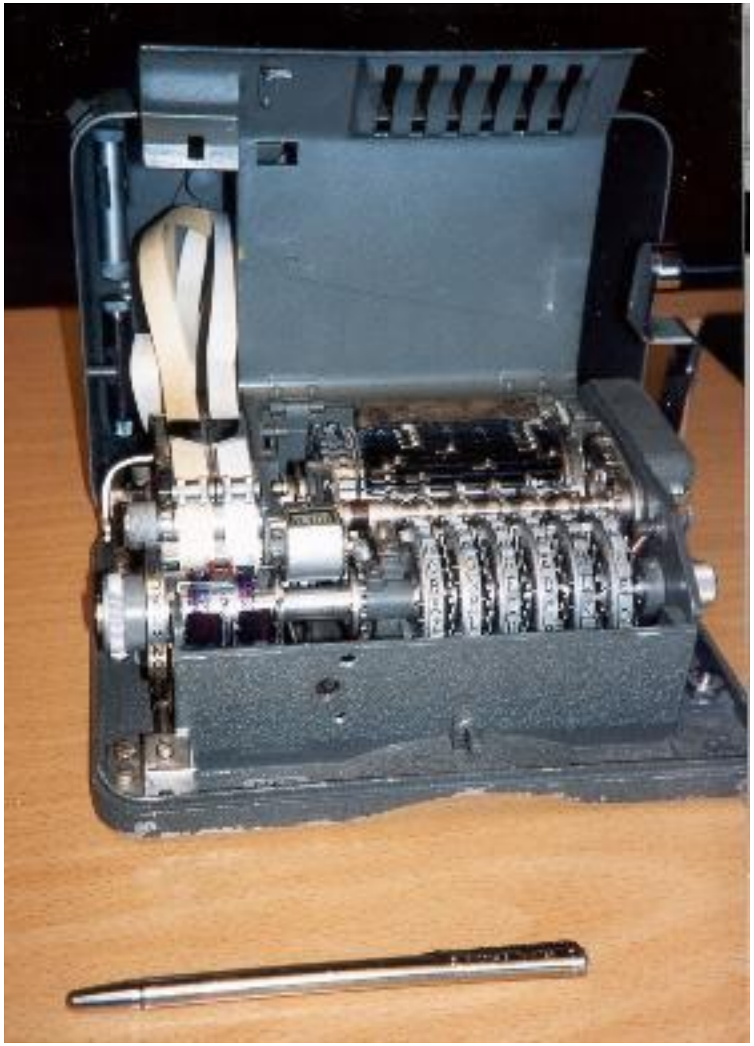
Product Ciphers (Hybrid Scheme)

- Ciphers using substitutions or transpositions are not secure because of language characteristics
- Hence consider using several ciphers in succession to make harder, but:
  - Two substitutions make a more complex substitution
  - Two transpositions make a more complex transposition
  - But ***a substitution followed by a transposition*** makes a new much harder cipher
- This is the bridge from classical to modern ciphers!

Rotor Machines

- Before modern ciphers, rotor machines were most common complex ciphers in use
- Widely used in WW2
  - German Enigma, Allied Hagelin, Japanese Purple
- Implemented a very complex, varying ***substitution*** cipher
- Used a series of cylinders, each giving one substitution, which rotated and changed after each letter was encrypted
- With 3 cylinders have 26^3=17576 alphabets

Hagelin Rotor Machine



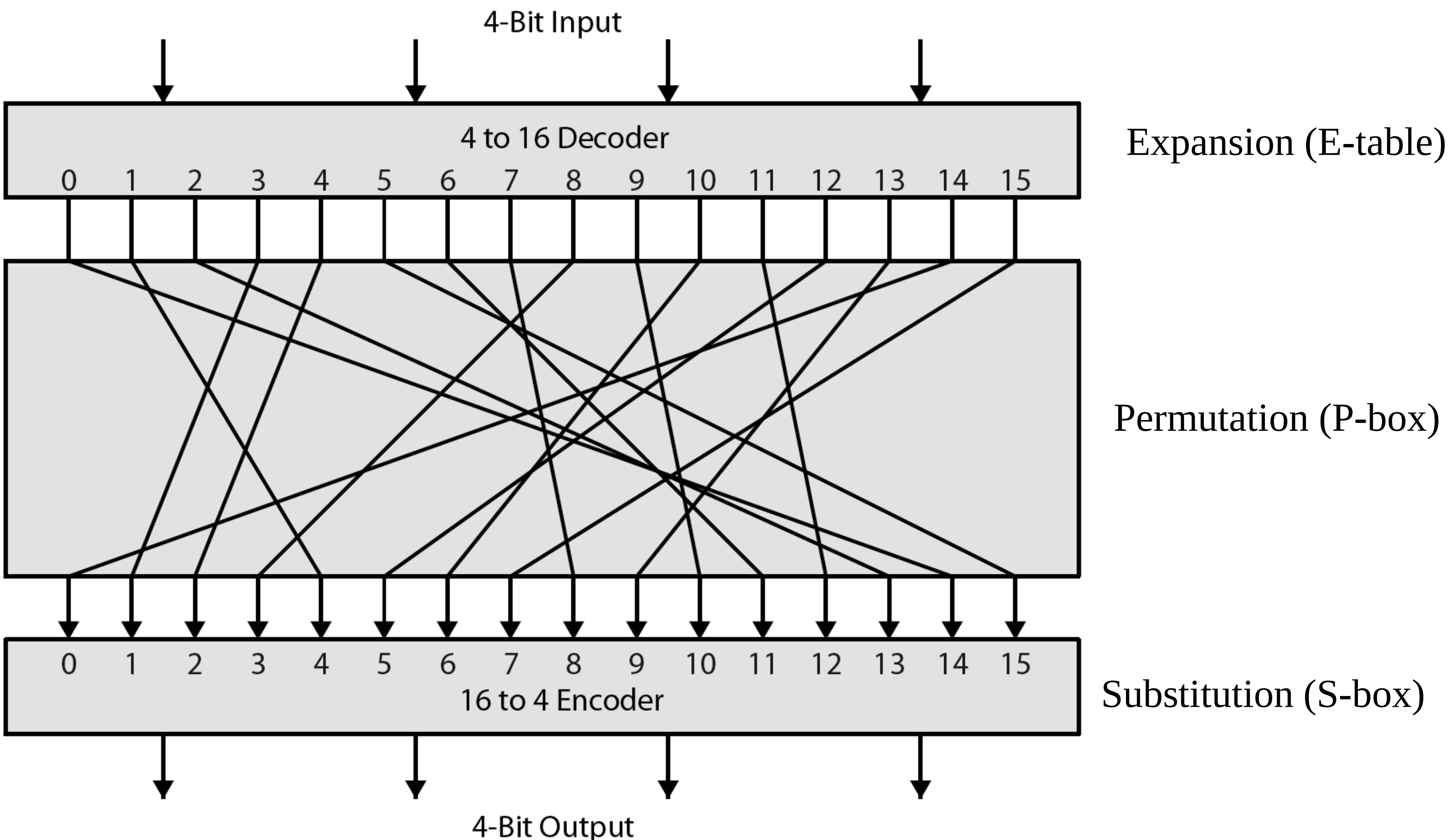
Steganography (藏头/尾诗)

- An alternative to encryption
- Hides existence of message
  - Using only a subset of letters/words in a longer message marked in some way
  - Using invisible ink
  - Hiding in LSB in graphic image or sound file
- Has drawbacks
  –

Block vs. Stream Ciphers

- Block ciphers
  - Process messages in blocks, each of which is then en/decrypted
  - Like a substitution on very big characters
    - 64-bits or more
      - Need a table of 2^64 entries for a 64-bit block
    - Instead, create from smaller building blocks
      - Using the idea of a product cipher
  - Many current ciphers are block ciphers
    - A wide range of applications
- Stream ciphers
  - Process messages a bit or byte at a time when en/decrypting

Ideal Block Cipher



Conventional Encryption Algorithms

- Data Encryption Standard (DES)
  - The most widely used encryption scheme
  - The algorithm is reffered to as the Data Encryption Algorithm (DEA)
  - DES is a block cipher
  - The plaintext is processed in 64-bit blocks
  - The key is 56-bits in length (original version)

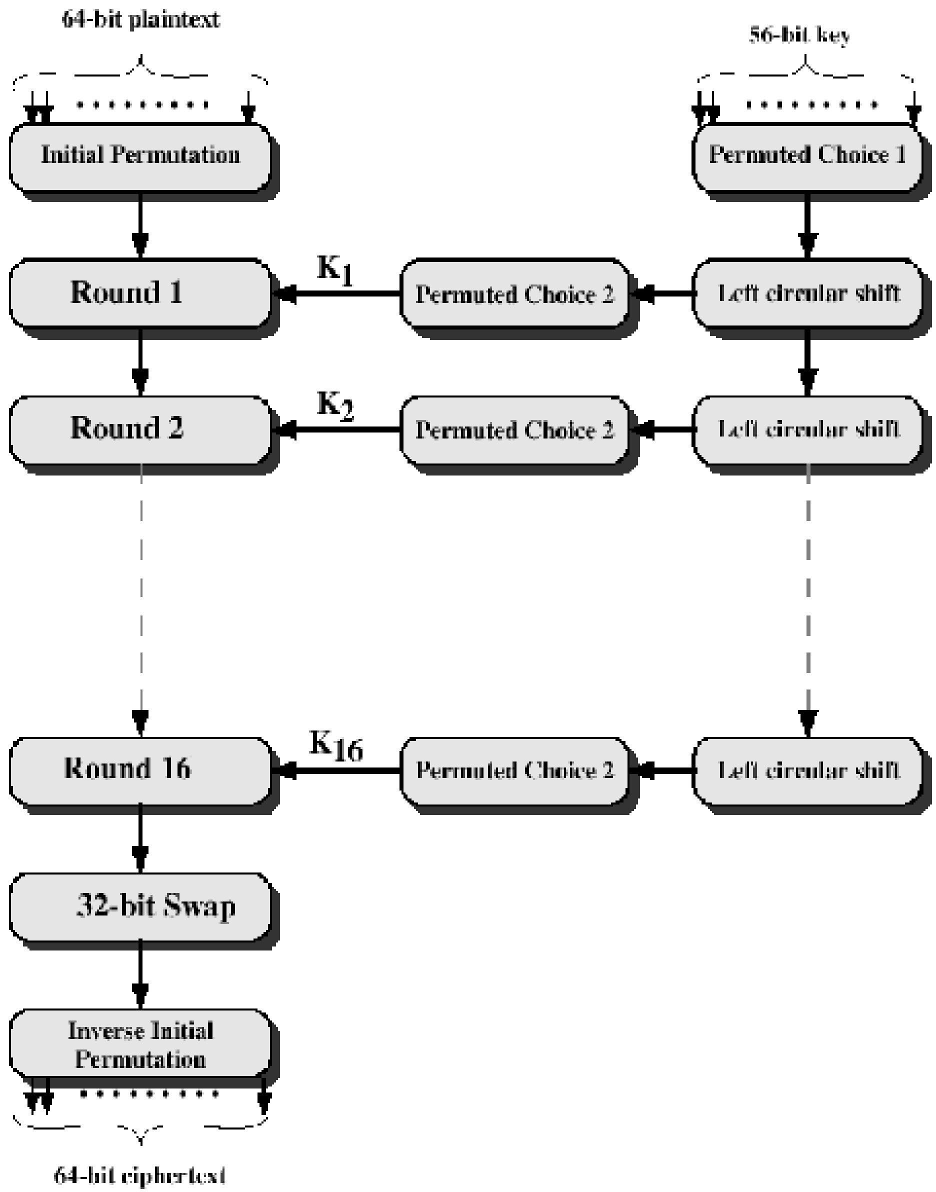


Figure 2.3 General Depiction of DES Encryption Algorithm

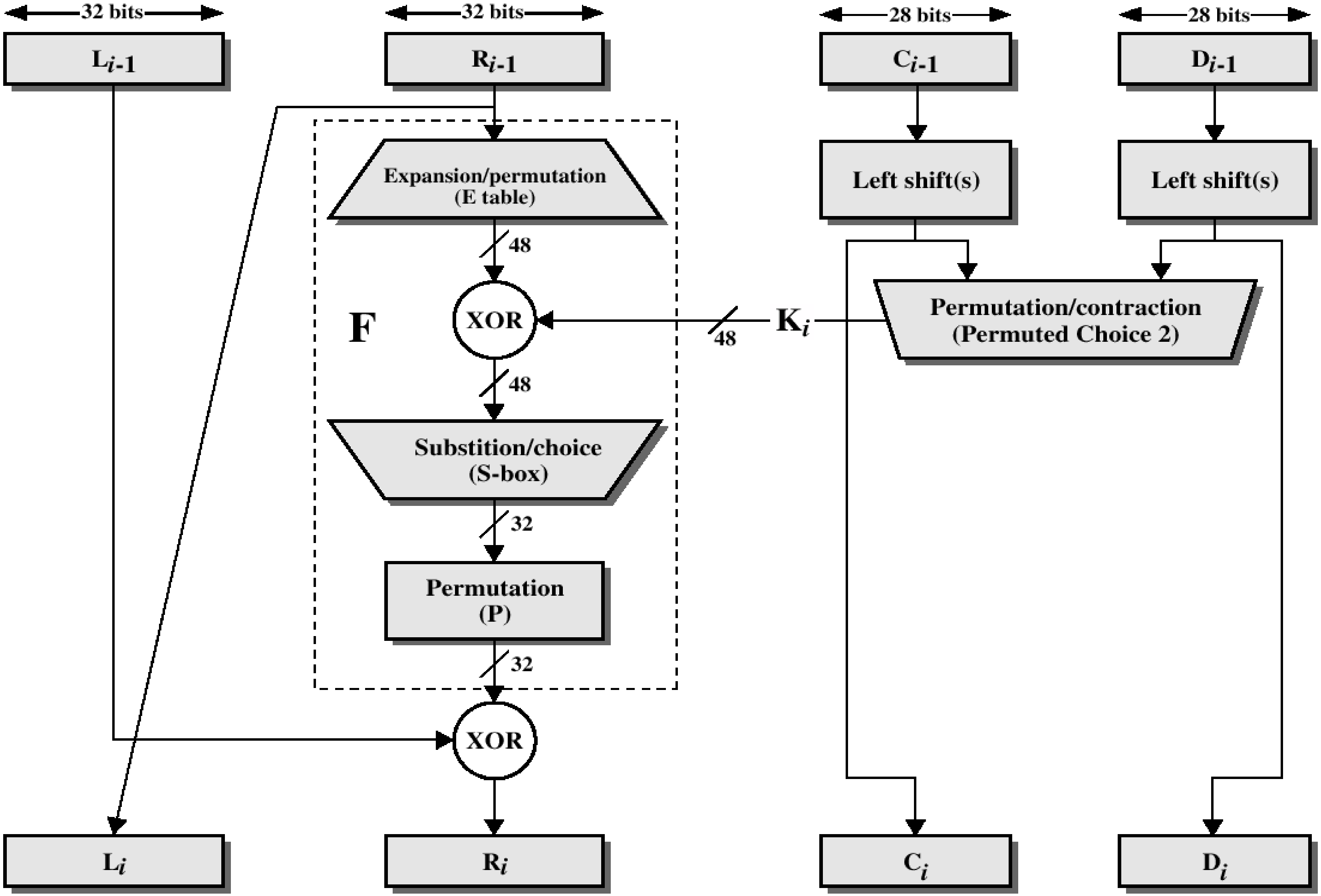


Figure 2.4 Single Round of DES Algorithm

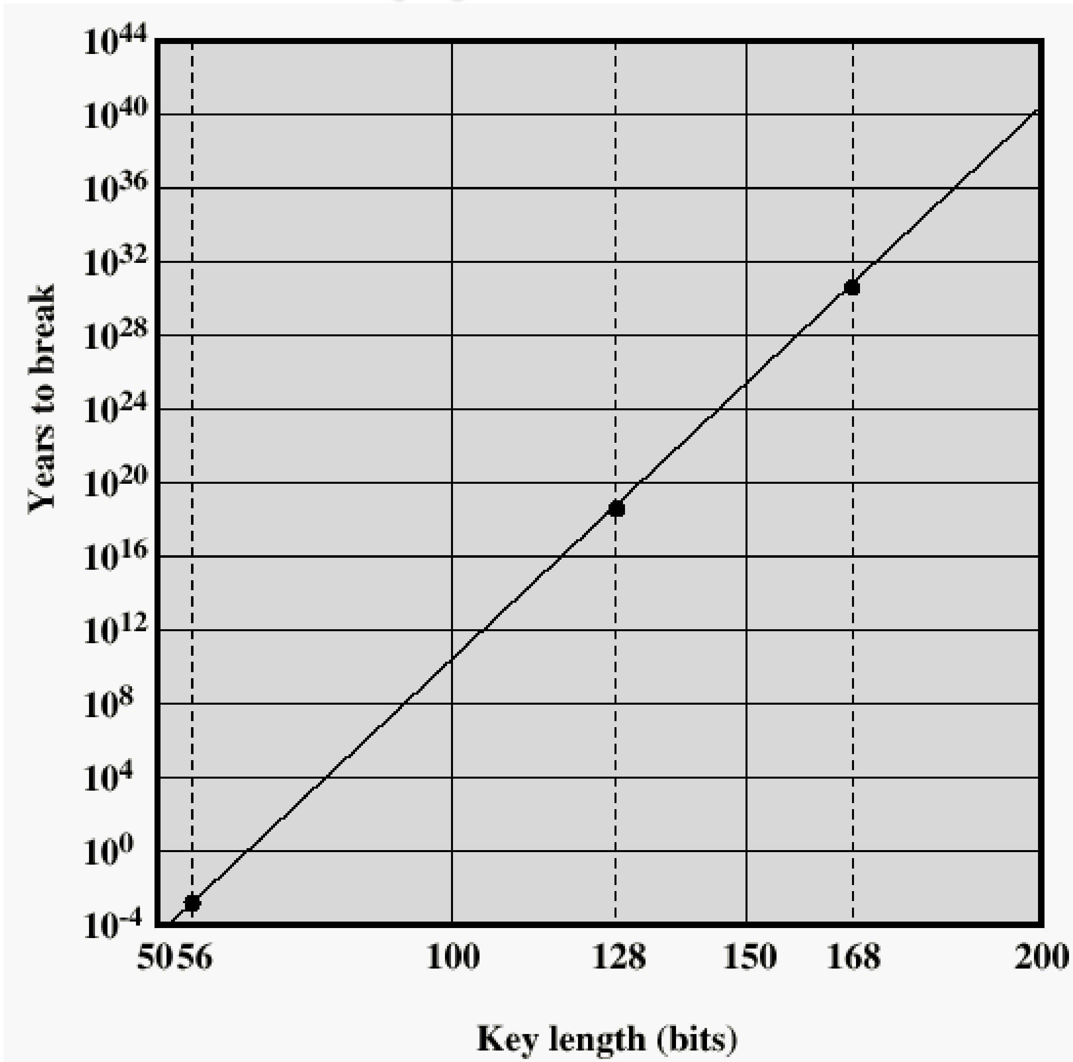
DES

- **Mathematically, the overall processing at each iteration:**
  - $L_i = R_{i-1}$
  - $R_i = L_{i-1} \otimes F(R_{i-1}, K_i)$
- Concerns about:
  - The algorithm and the key length (56-bits)

Avalanche Effect

- An important desirable property of encryption algorithm
  - A change of **one** input or key bit results in changing approx **half** output bits
  - Making attacks by guessing keys impossible
- DES exhibits strong avalanche

Time to break a code (10^6 decryptions/μs)



Strength of DES – Key Size

- 56-bit keys have $2^{56} = 7.2 \times 10^{16}$ values
  - Brute force search looks hard
- Recent advances have shown possibilities
  - In 1997 on the Internet in a few months
  - In 1998 on dedicated h/w (EFF) in a few days
  - In 1999 above combined in 22 hrs!
- Still must be able to recognize plaintext

Strength of DES – Analytic Attacks

- Several analytic attacks utilizing some deep structure of the cipher
  - By gathering information about encryptions
  - Can eventually recover some/all of the sub-key bits
  - If necessary then exhaustively search for the rest
- Statistical attacks
  - Differential cryptanalysis
  - Linear cryptanalysis
  - Related key attacks
- Must now consider alternatives to DES

Alternatives to DES

- A replacement for DES was needed
  - Have theoretical attacks that can break it
  - have demonstrated exhaustive key search attacks
- A strengthened DES
  - Triple-DEA (Triple-DES)

Triple DEA

- Use three keys and three executions of the DES algorithm (encrypt-decrypt-encrypt)

$$C = E_{K3}[D_{K2}[E_{K1}[P]]]$$

  - C = ciphertext
  - P = Plaintext
  - EK[X] = encryption of X using key K
  - DK[Y] = decryption of Y using key K
- Effective key length of

Triple DEA

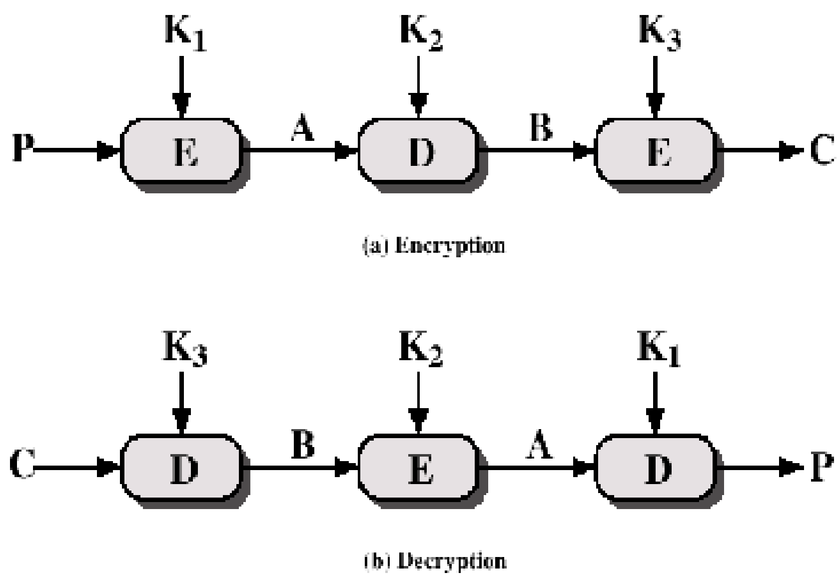


Figure 2.6 Triple DEA

Alternatives to DES

- Triple-DES
  - Slow
    - Use small blocks
- AES Cipher – Rijndael
  - Designed by Joan. Daemen and Vincent Rijmen in Belgium
  - Has 128/192/256-bit keys, 128-bit data
  - An **iterative** rather than **feistel** cipher
    - Processes data as block of 4 columns of 4 bytes
    - Operates on entire data block in every round

Substitution-Permutation (S-P) Ciphers

- S-P: substitution-permutation
  - Two primitive cryptographic operations
    - *Substitution* (S-box)
    - *Permutation* (P-box)
  - Introduced by Claude Shannon in 1949 paper
  - Form the basis of modern block ciphers
  - Provide *confusion* & *diffusion* of message & key

Confusion and Diffusion

- Cipher needs to completely obscure statistical properties of the original message
  - A one-time pad does this
- More practically, Shannon suggested combining S & P elements to obtain:
  - **Diffusion**
    - Dissipates statistical structure of plaintext over bulk of ciphertext
  - **Confusion**
    - Makes relationship between ciphertext and key as complex as possible

Other Symmetric Block Ciphers

- **International Data Encryption Algorithm (IDEA)**
  - 128-bit key
  - Used in PGP
- **Blowfish**
  - Easy to implement
  - High execution speed
  - Run in less than 5K of memory

Other Symmetric Block Ciphers

- **RC5**
  - Suitable for hardware and software
  - Fast, simple
  - Adaptable to processors of different word lengths
  - Variable number of rounds
  - Variable-length key
  - Low memory requirement
  - High security
  - Data-dependent rotations
- **Cast-128**
  - Key size from 40 to 128 bits
  - The round function differs from round to round

Advanced Encryption Standard

Origins

- clear a replacement for DES was needed
  - have theoretical attacks that can break it
  - have demonstrated exhaustive key search attacks
- can use Triple-DES – but slow, has small blocks
- US NIST issued call for ciphers in 1997
- 15 candidates accepted in Jun 98
- 5 were shortlisted in Aug-99
- Rijndael was selected as the AES in Oct-2000
- issued as FIPS PUB 197 standard in Nov-2001

AES Requirements

- private key symmetric block cipher
- 128-bit data, 128/192/256-bit keys
- stronger & faster than Triple-DES
- active life of 20-30 years (+ archival use)
- provide full specification & design details
- both C & Java implementations
- NIST have released all submissions & unclassified analyses

AES Evaluation Criteria

- initial criteria:
  - security – effort for practical cryptanalysis
  - cost – in terms of computational efficiency
  - algorithm & implementation characteristics
- final criteria
  - general security
  - ease of software & hardware implementation
  - implementation attacks
  - flexibility (in en/decrypt, keying, other factors)

AES Shortlist

- after testing and evaluation, shortlist in Aug-99:
  - MARS (IBM) - complex, fast, high security margin
  - RC6 (USA) - v. simple, v. fast, low security margin
  - Rijndael (Belgium) - clean, fast, good security margin
  - Serpent (Euro) - slow, clean, v. high security margin
  - Twofish (USA) - complex, v. fast, high security margin
- then subject to further analysis & comment
- saw contrast between algorithms with
  - few complex rounds verses many simple rounds
  - which refined existing ciphers verses new proposals

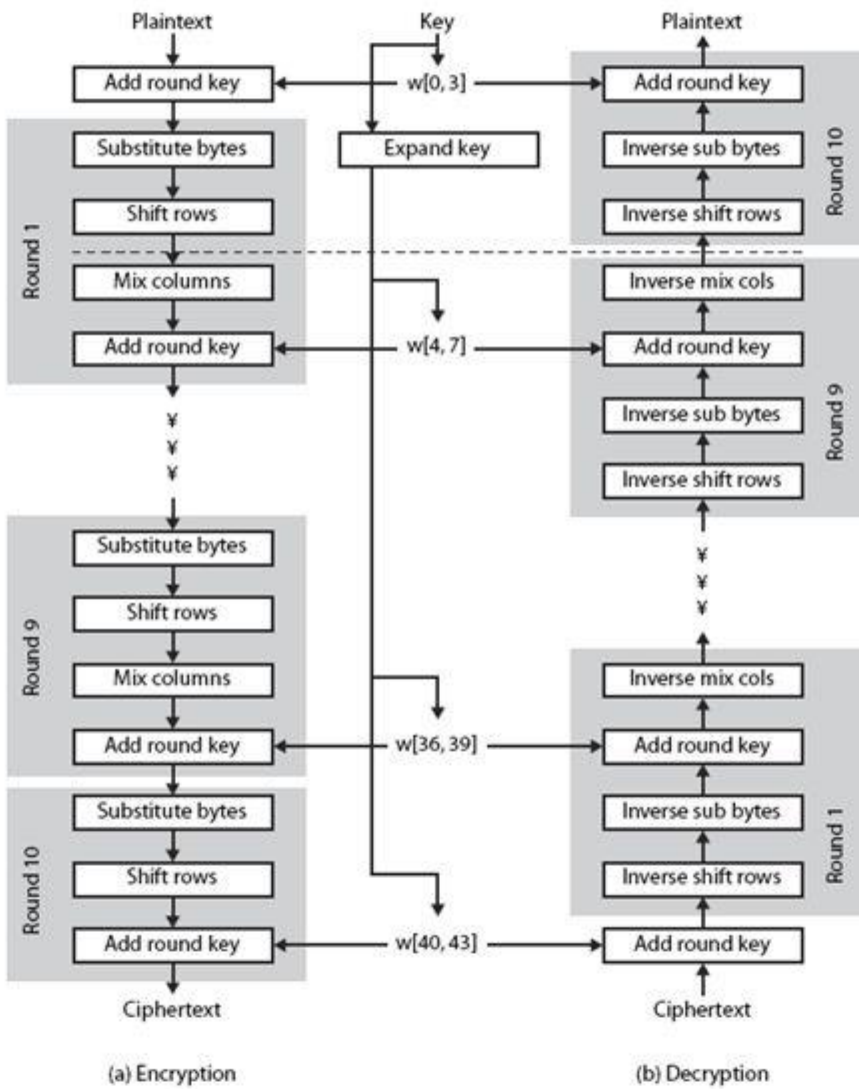
The AES Cipher - Rijndael

- designed by Rijmen-Daemen in Belgium
- has 128/192/256 bit keys, 128 bit data
- an **iterative** rather than **feistel** cipher
  – processes data as block of 4 columns of 4 bytes
  – operates on entire data block in every round
- designed to be:
  – resistant against known attacks
  – speed and code compactness on many CPUs
  – design simplicity

Rijndael

- data block of 4 columns of 4 bytes is state
- key is expanded to array of words
- has 9/11/13 rounds in which state undergoes:
  - byte substitution (1 S-box used on every byte)
  - shift rows (permute bytes between groups/columns)
  - mix columns (subs using matrix multipy of groups)
  - add round key (XOR state with key material)
  - view as alternating XOR key & scramble data bytes
- initial XOR key material & incomplete last round
- with fast XOR & table lookup implementation

Rijndael

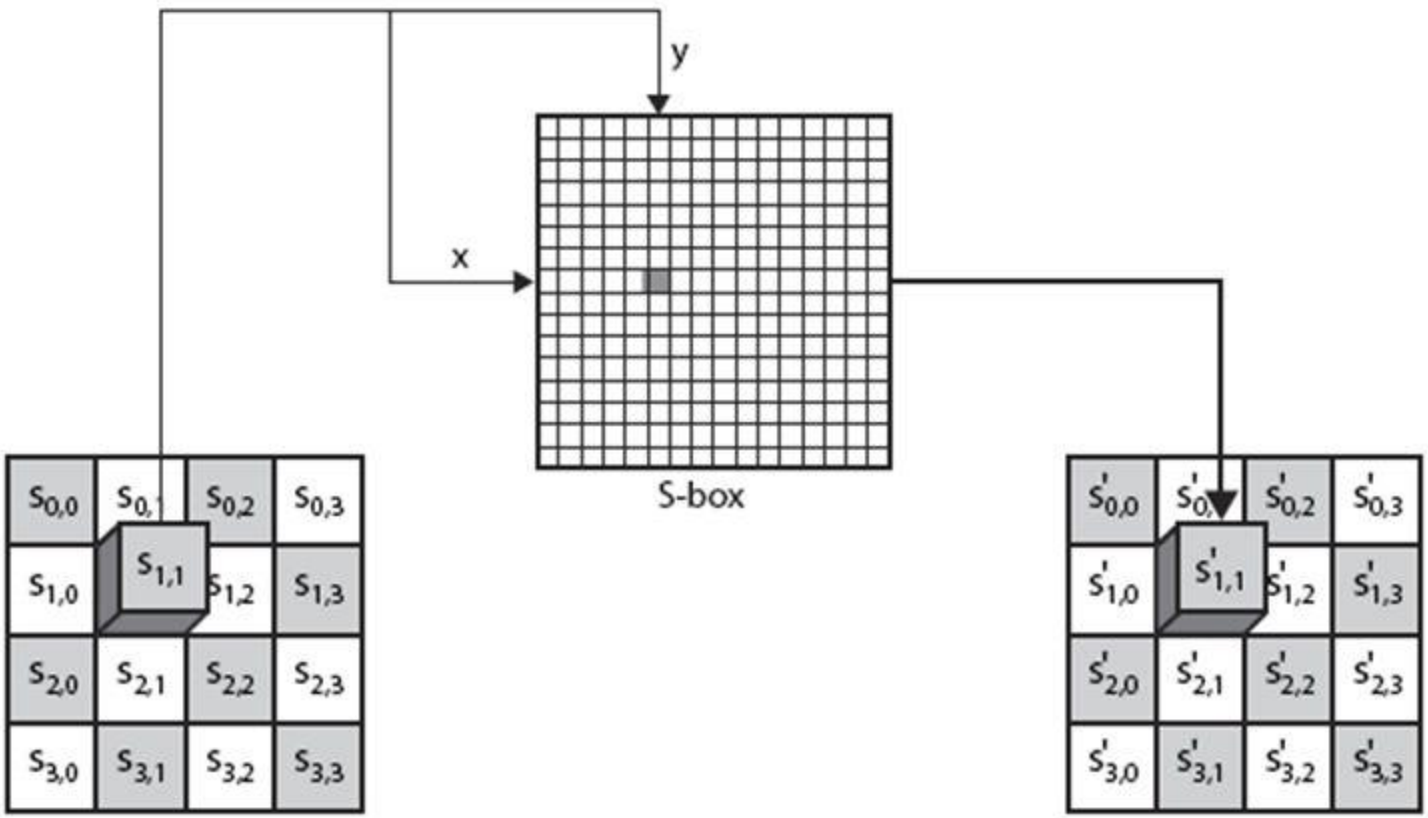


Plaintext
Key
Plaintext
Add round key
w[0, 3]
Add round key
Substitute bytes
Expand key
Inverse sub bytes
Round 10
Shift rows
Inverse shift rows
Round 1
Mix columns
Inverse mix cols
Add round key
w[4, 7]
Add round key
Inverse sub bytes
Round 9
Inverse shift rows
Substitute bytes
Round 9
Shift rows
Mix columns
Inverse mix cols
Add round key
w[36, 39]
Add round key
Substitute bytes
Inverse sub bytes
Round 10
Shift rows
Inverse shift rows
Round 1
Add round key
w[40, 43]
Add round key
Ciphertext
Ciphertext
(a) Encryption
(b) Decryption

Byte Substitution

- a simple substitution of each byte
- uses one table of 16x16 bytes containing a permutation of all 256 8-bit values
- each byte of state is replaced by byte indexed by row (left 4-bits) & column (right 4-bits)
  - eg. byte {95} is replaced by byte in row 9 column 5
  - which has value {2A}
- S-box constructed using defined transformation of values in $GF(2^8)$
- designed to be resistant to all known attacks

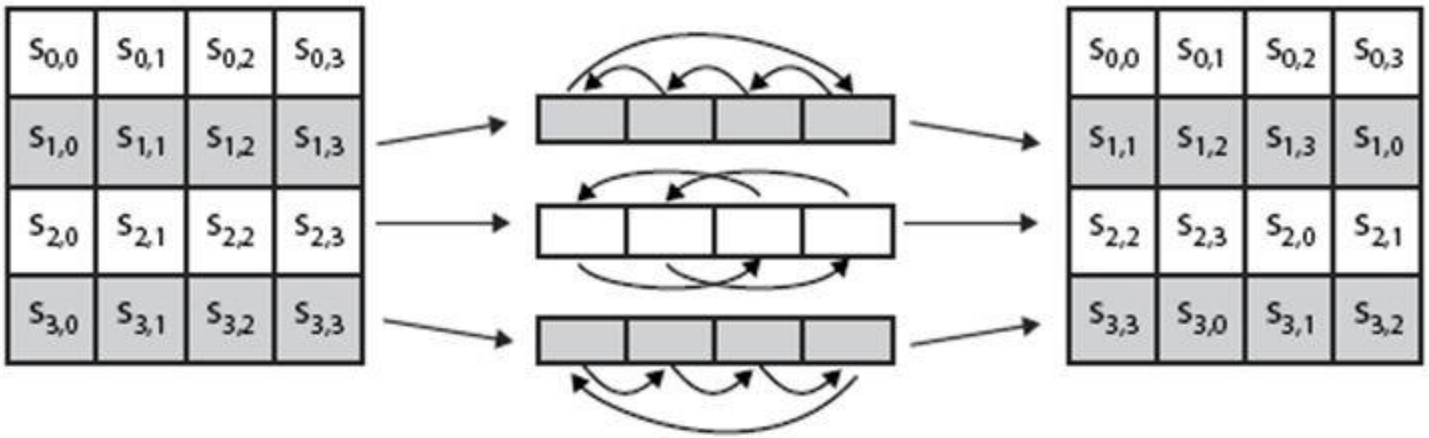
Byte Substitution



Shift Rows

- a circular byte shift in each each
  - 1st row is unchanged
  - 2nd row does 1 byte circular shift to left
  - 3rd row does 2 byte circular shift to left
  - 4th row does 3 byte circular shift to left
- decrypt inverts using shifts to right
- since state is processed by columns, this step permutes bytes between the columns

Shift Rows



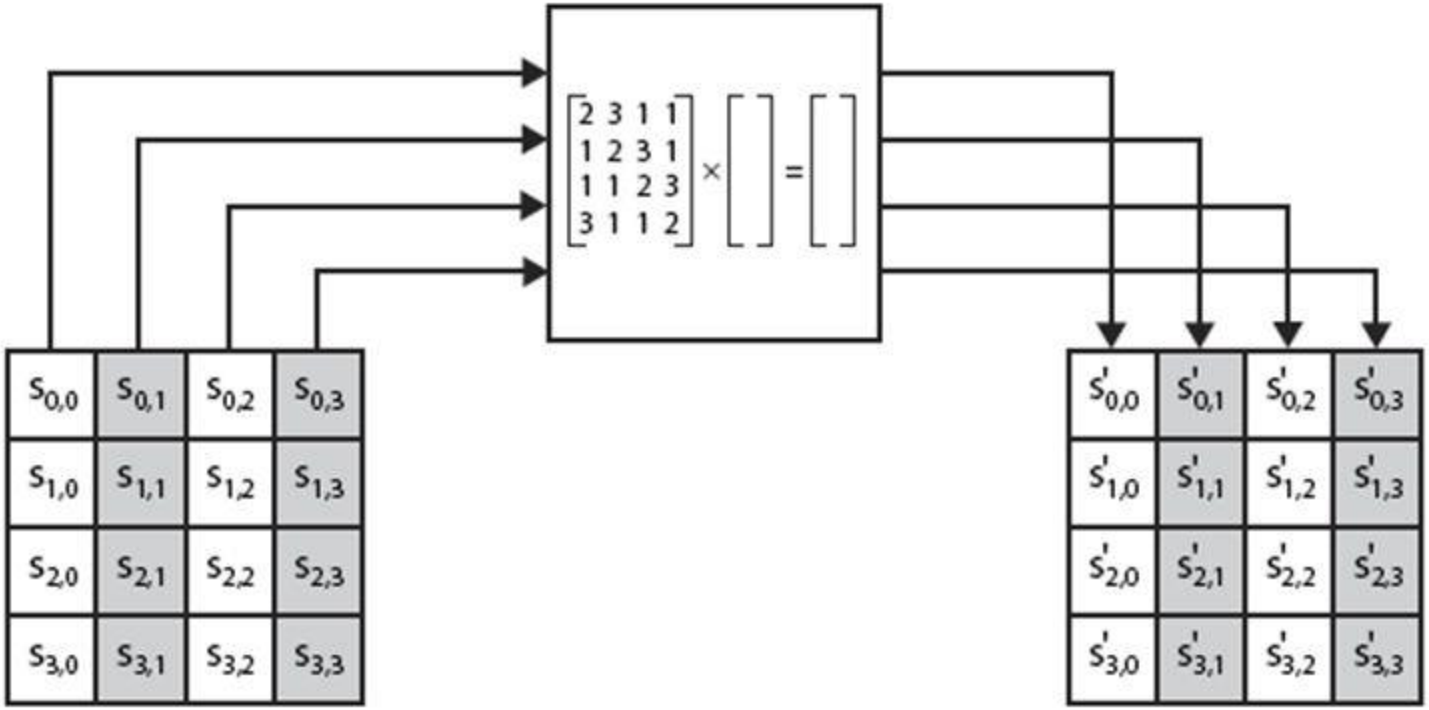
$s_{0,0}$ $s_{0,1}$ $s_{0,2}$ $s_{0,3}$
$s_{1,0}$ $s_{1,1}$ $s_{1,2}$ $s_{1,3}$
$s_{2,0}$ $s_{2,1}$ $s_{2,2}$ $s_{2,3}$
$s_{3,0}$ $s_{3,1}$ $s_{3,2}$ $s_{3,3}$
$s_{0,0}$ $s_{0,1}$ $s_{0,2}$ $s_{0,3}$
$s_{1,1}$ $s_{1,2}$ $s_{1,3}$ $s_{1,0}$
$s_{2,2}$ $s_{2,3}$ $s_{2,0}$ $s_{2,1}$
$s_{3,3}$ $s_{3,0}$ $s_{3,1}$ $s_{3,2}$

Mix Columns

- each column is processed separately
- each byte is replaced by a value dependent on all 4 bytes in the column
- effectively a matrix multiplication in $GF(2^8)$ using prime poly $m(x) = x^8+x^4+x^3+x+1$

$$\begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \begin{bmatrix} s_{0,0} & s_{0,1} & s_{0,2} & s_{0,3} \\ s_{1,0} & s_{1,1} & s_{1,2} & s_{1,3} \\ s_{2,0} & s_{2,1} & s_{2,2} & s_{2,3} \\ s_{3,0} & s_{3,1} & s_{3,2} & s_{3,3} \end{bmatrix} = \begin{bmatrix} s'_{0,0} & s'_{0,1} & s'_{0,2} & s'_{0,3} \\ s'_{1,0} & s'_{1,1} & s'_{1,2} & s'_{1,3} \\ s'_{2,0} & s'_{2,1} & s'_{2,2} & s'_{2,3} \\ s'_{3,0} & s'_{3,1} & s'_{3,2} & s'_{3,3} \end{bmatrix}$$

Mix Columns



2 3 1 1
1 2 3 1
1 1 2 3
3 1 1 2
×
=
$s_{0,0}$ $s_{0,1}$ $s_{0,2}$ $s_{0,3}$
$s_{1,0}$ $s_{1,1}$ $s_{1,2}$ $s_{1,3}$
$s_{2,0}$ $s_{2,1}$ $s_{2,2}$ $s_{2,3}$
$s_{3,0}$ $s_{3,1}$ $s_{3,2}$ $s_{3,3}$
$s'_{0,0}$ $s'_{0,1}$ $s'_{0,2}$ $s'_{0,3}$
$s'_{1,0}$ $s'_{1,1}$ $s'_{1,2}$ $s'_{1,3}$
$s'_{2,0}$ $s'_{2,1}$ $s'_{2,2}$ $s'_{2,3}$
$s'_{3,0}$ $s'_{3,1}$ $s'_{3,2}$ $s'_{3,3}$

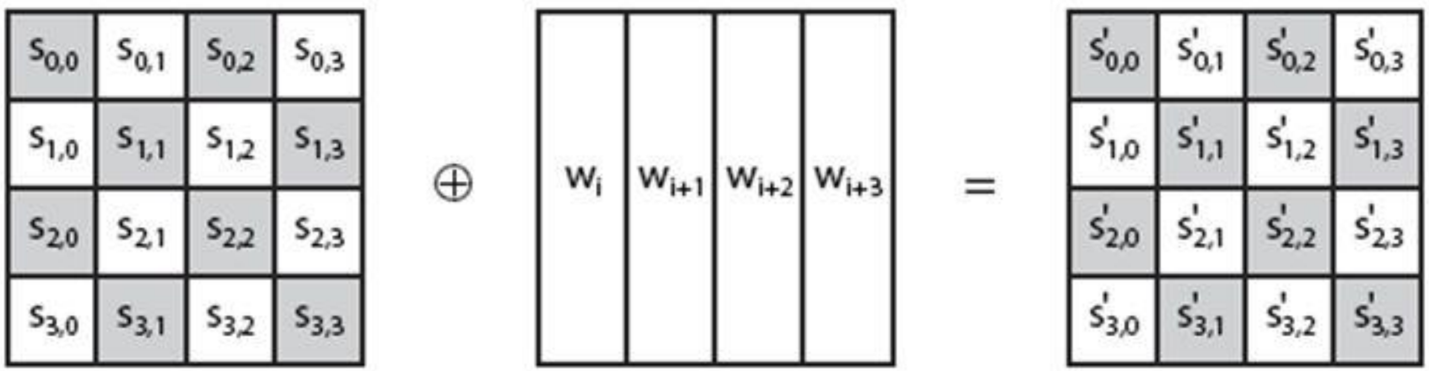
Mix Columns

- can express each col as 4 equations
  - to derive each new byte in col
- decryption requires use of inverse matrix
  - with larger coefficients, hence a little harder
- have an alternate characterisation
  - each column a 4-term polynomial
  - with coefficients in $GF(2^8)$
  - and polynomials multiplied modulo (x^4+1)

Add Round Key

- XOR state with 128-bits of the round key
- again processed by column (though effectively a series of byte operations)
- inverse for decryption identical
  - since XOR own inverse, with reversed keys
- designed to be as simple as possible
  - a form of Vernam cipher on expanded key
  - requires other stages for complexity / security

Add Round Key



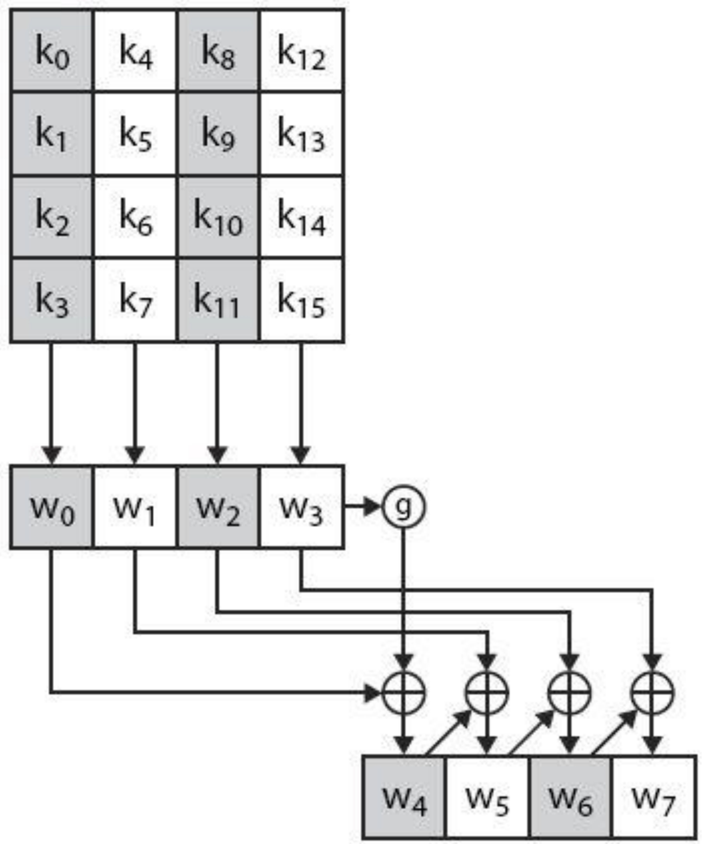
AES Round


State
SubBytes
S
State
ShiftRows
State
MixColumns
M
State
r_0 r_1 r_2 r_3 r_4 r_5 r_0 r_7 r_8 r_9 r_{10} r_{11} r_{12} r_{13} r_{14} r_{15}
AddRoundKey
State

AES Key Expansion

- takes 128-bit (16-byte) key and expands into array of 44/52/60 32-bit words
- start by copying key into first 4 words
- then loop creating words that depend on values in previous & 4 places back
  - in 3 of 4 cases just XOR these together
  - 1st word in 4 has rotate + S-box + XOR round constant on previous, before XOR 4th back

AES Key Expansion



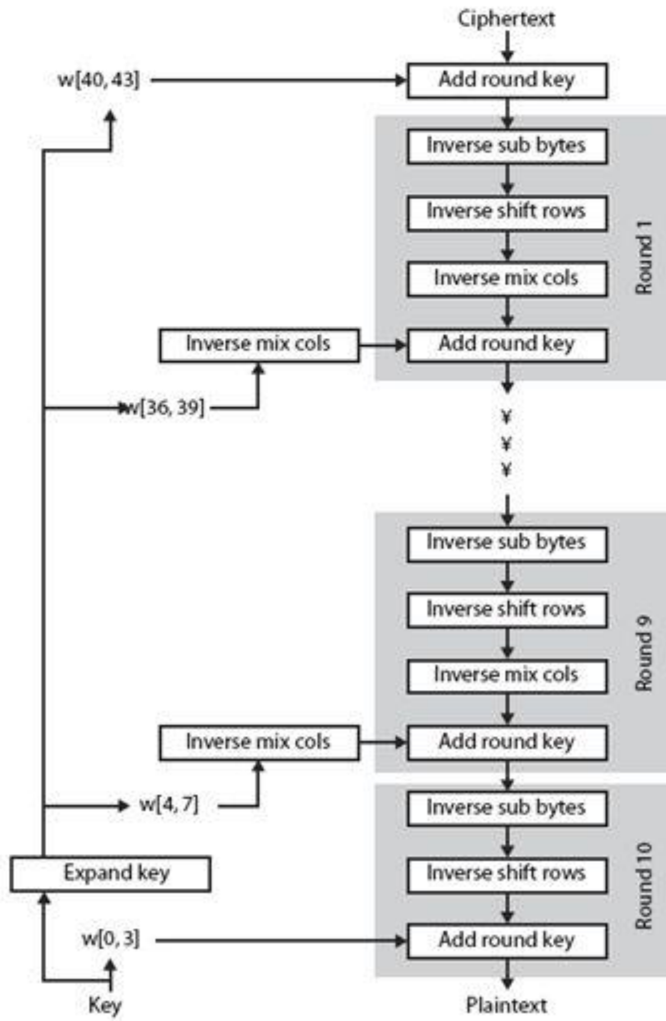
Key Expansion Rationale

- designed to resist known attacks
- design criteria included
  - knowing part key insufficient to find many more
  - invertible transformation
  - fast on wide range of CPU's
  - use round constants to break symmetry
  - diffuse key bits into round keys
  - enough non-linearity to hinder analysis
  - simplicity of description

AES Decryption

- AES decryption is not identical to encryption since steps done in reverse
- but can define an equivalent inverse cipher with steps as for encryption
  - but using inverses of each step
  - with a different key schedule
- works since result is unchanged when
  - swap byte substitution & shift rows
  - swap mix columns & add (tweaked) round key

AES Decryption



Implementation Aspects

- can efficiently implement on 8-bit CPU
  - byte substitution works on bytes using a table of 256 entries
  - shift rows is simple byte shift
  - add round key works on byte XOR's
  - mix columns requires matrix multiply in $GF(2^8)$ which works on byte values, can be simplified to use table lookups & byte XOR's

Implementation Aspects

- can efficiently implement on 32-bit CPU
  - redefine steps to use 32-bit words
  - can precompute 4 tables of 256-words
  - then each column in each round can be computed using 4 table lookups + 4 XORs
  - at a cost of 4Kb to store tables
- designers believe this very efficient implementation was a key factor in its selection as the AES cipher