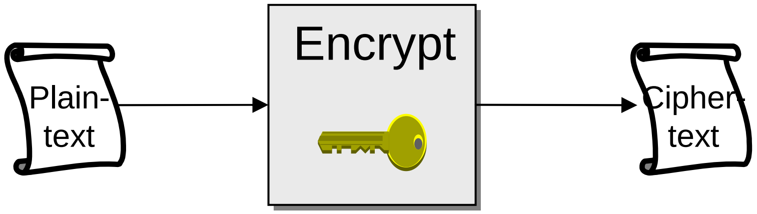
Network Security

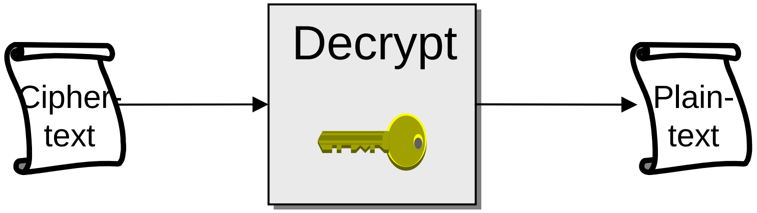
Chapter 3
Symmetric Cryptography

- General Description
- Modes of Encryption
- Data Encryption Standard (DES)
- Advanced Encryption Standard (AES)
- The Stream Cipher RC4

Symmetric Encryption

- General description:
  - The same key $K_{A,B}$ is used for enciphering and deciphering of messages:





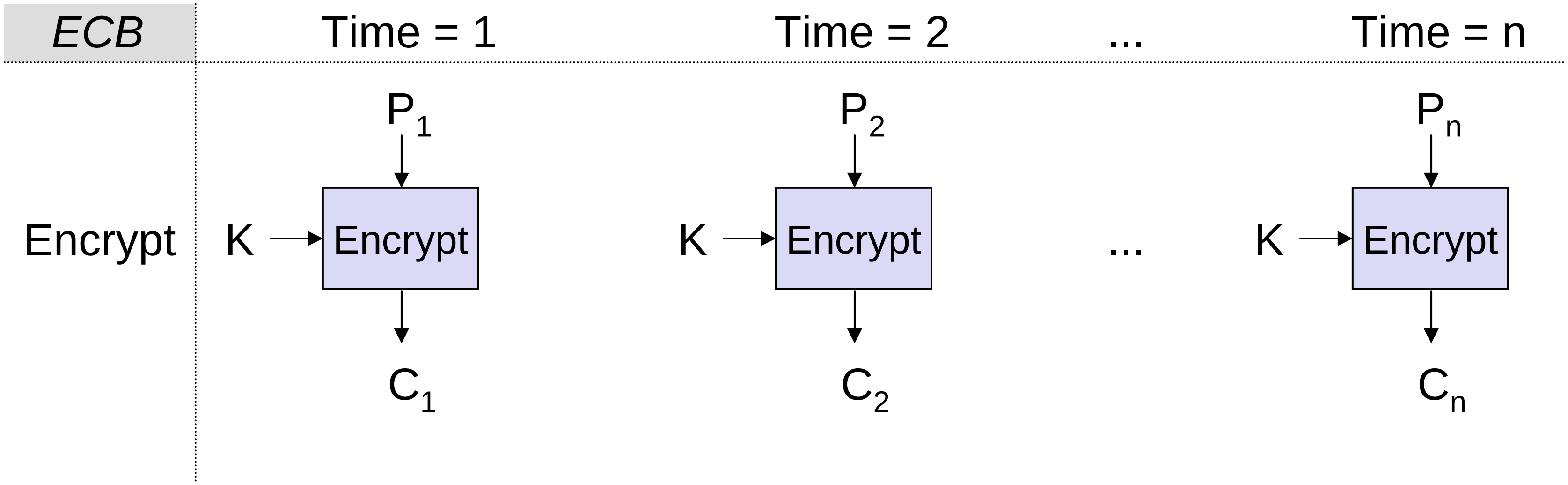
- Notation
  - If P denotes the plaintext message, $E(K_{A,B}, P)$ denotes the cipher text and it holds $D(K_{A,B}, E(K_{A,B}, P)) = P$
  - Alternatively we sometimes write $\{P\}_{K_{A,B}}$ or $E_{K_{A,B}}(P)$ for $E(K_{A,B}, P)$
- Symmetric encryption
  - $E_{K_{A,B}}$ is a bijective function
  - $D_{K_{A,B}}$ is the inverse function of $E_{K_{A,B}}$: $D_{K_{A,B}} = (E_{K_{A,B}})^{-1}$
- Examples: DES, 3DES, AES, RC4

Modes of Encryption

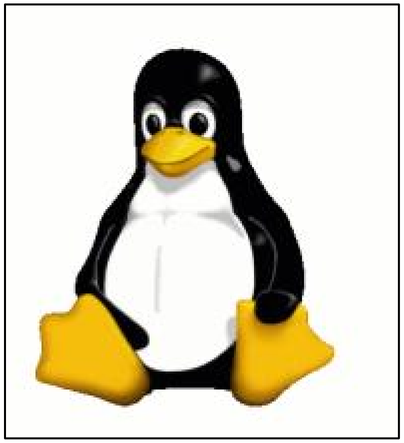
- Block cipher modes
  - A plaintext *p* is segmented in blocks p_1, p_2, ... each of length *b* or *j*, respectively, where *b* denotes the block size of the encryption algorithm and $j < b$
  - The ciphertext *c* is the combination of c_1, c_2, ... where c_i denotes the result of the encryption of the i^{th} block of the plaintext message
  - The entities encrypting and decrypting a message have agreed upon a key *K.*
  - E.g.
    - Electronic Code Book Mode (ECB)
    - Cipher Block Chaining Mode (CBC)
- Stream cipher modes
  - A pseudorandom stream of bytes, called *key stream* is generated from the symmetric key *K*
  - The key stream is XORed with the plain text to generate the cipher text.
  - E.g.
    - Output Feedback Mode (OFB)
    - Counter Mode (CTR)

Symmetric Block Ciphers - Modes of Encryption – ECB (1)

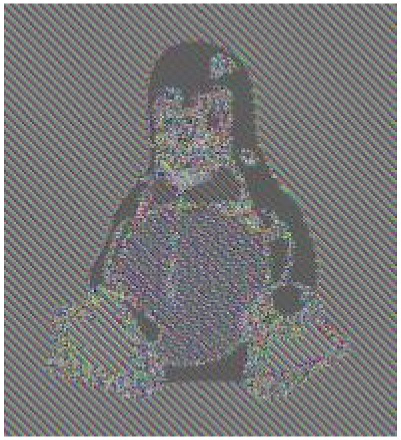
- *Electronic Code Book Mode (ECB):*
  - Every block p_i of length b is encrypted independently: $c_i = E(K, p_i)$
  - A bit error in one ciphertext block c_i results in a completely wrongly recovered plaintext block $p_i´$ (subsequent blocks are not affected)
  - Loss of synchronization does not have any effect if integer multiples of the block size b are lost.
    If any other number of bits are lost, explicit re-synchronization is needed.
  - Drawback: identical plaintext blocks are encrypted to identical ciphertext!



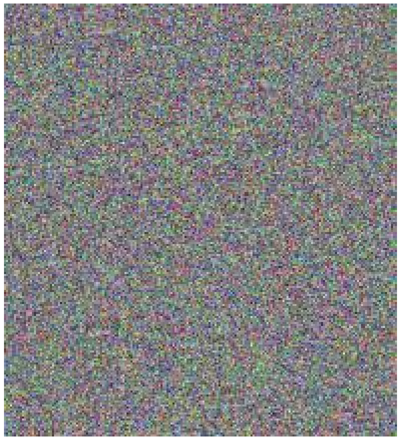
Symmetric Block Ciphers - Modes of Encryption – ECB (2)



Original



Encrypted using ECB mode



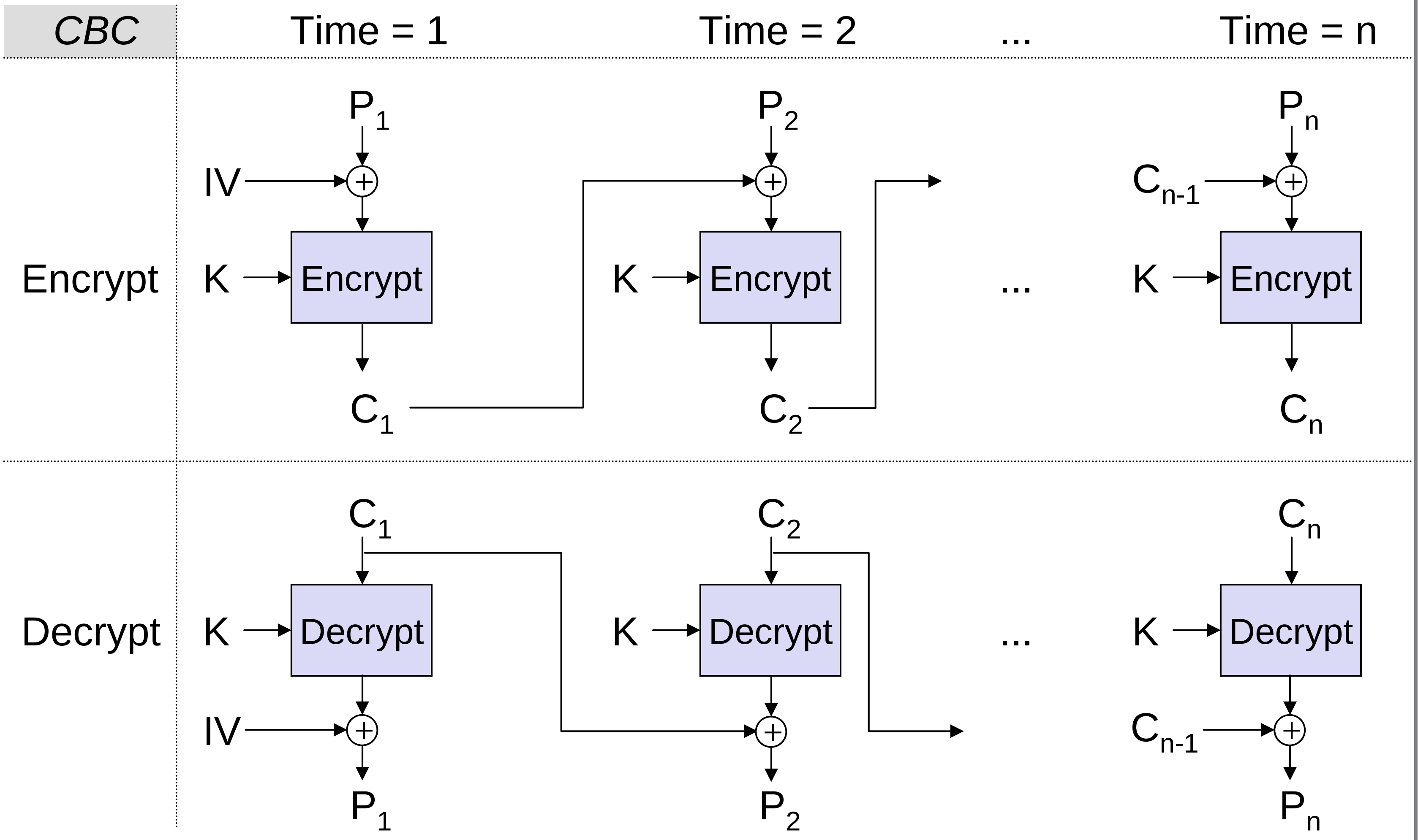
Encrypted using other modes

Source: http://www.wikipedia.org/

Symmetric Block Ciphers - Modes of Encryption – CBC (1)

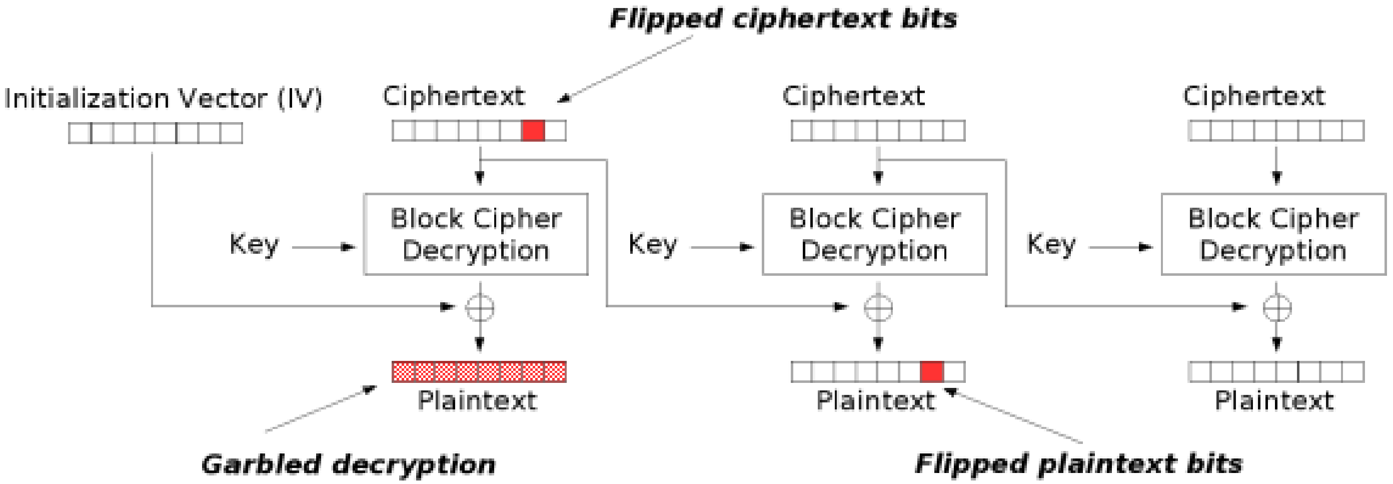
- *Cipher Block Chaining Mode (CBC):*
  - Before encrypting a plaintext block p_i , it is XORed (⊕) with the preceding ciphertext block c_{i-1}:
    - $c_i = E(K, c_{i-1} \oplus p_i)$
    - $p_i´ = c_{i-1} \oplus D(K, c_i)$
  - Both parties agree on an *initial value* for c_i *called Initialization Vector (IV)*
    - c_0 = IV
- Properties:
  - Advantage: identical plaintext blocks are encrypted to non-identical ciphertext.
  - Error propagation:
    - A distorted ciphertext block results in two distorted plaintext blocks, as $p_i´$ is computed using c_{i-1} and c_i
  - Synchronisation:
    - If the number of lost bits is a multiple integer of *b,* one additional block p_{i+1} is misreprensted before synchronization is re-established.
      If any other number of bits are lost explicit re-synchronization is needed.
  - Applicable for
    - Encryption
    - Integrity check: use last block of CBC as Message Authentication Code (MAC)

Symmetric Block Ciphers - Modes of Encryption – CBC (2)



CBC Error Propagation

- A distorted ciphertext block results in two distorted plaintext blocks, as p_i' is computed using c_{i-1} and c_i



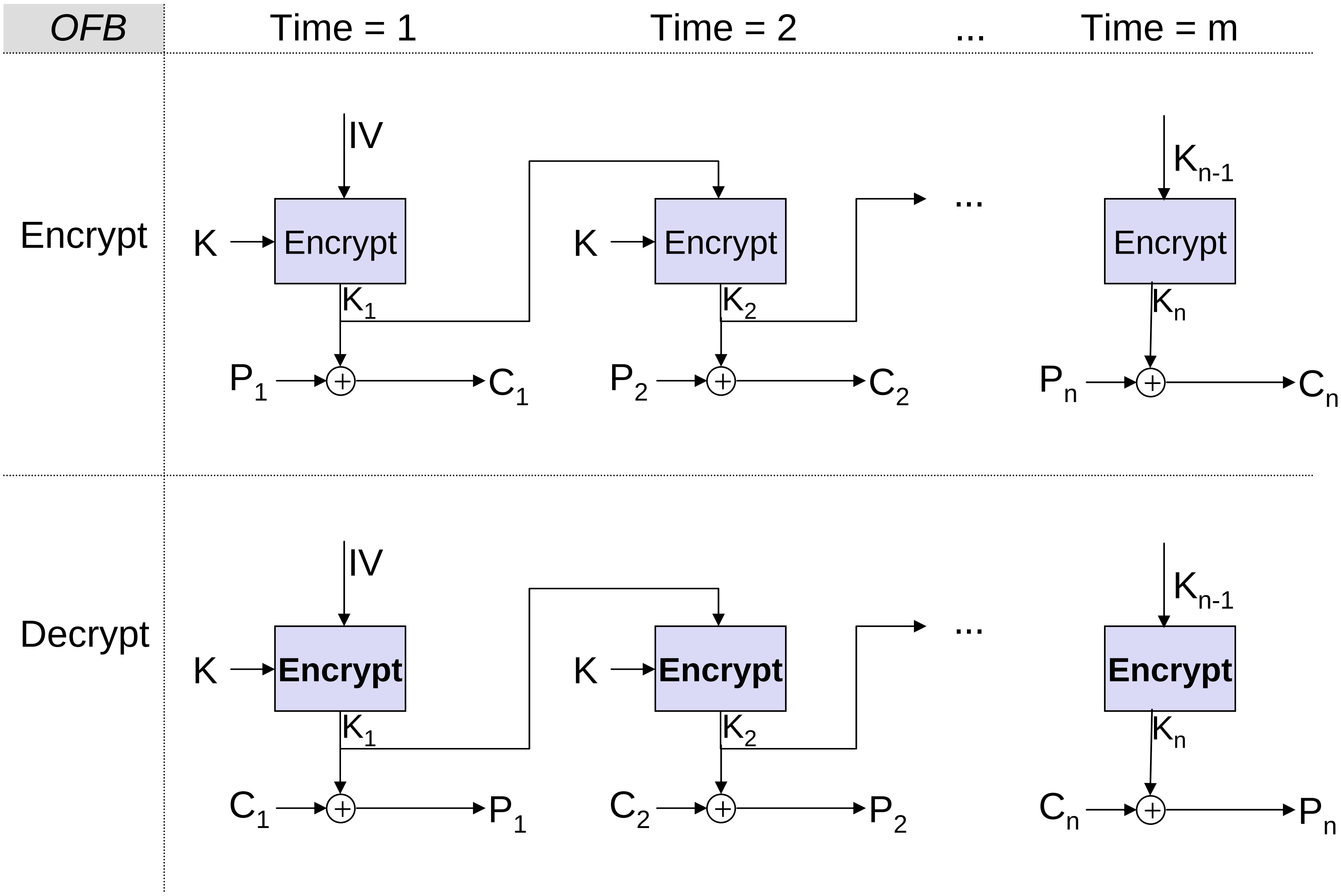
Modification attack or transmission error for CBC

Source: http://www.wikipedia.org/

Symmetric Block Ciphers - Modes of Encryption – OFB (1)

- *Output Feedback Mode (OFB):*
  - The block encryption algorithm is used to generate a key stream that depends only on *K* and *IV*
    - $K_0 = IV$
    - $K_i = E(K, K_{i-1})$
    - $C_i = P_i \oplus K_i$
  - The plaintext is XORed with the pseudo-random sequence to obtain the ciphertext and vice versa

Symmetric Block Ciphers - Modes of Encryption – OFB (2)



Symmetric Block Ciphers - Modes of Encryption – OFB (3)

- Properties of OFB:
  - Error propagation:
    - Single bit errors result only in single bit errors ⇒ no error multiplication
  - Synchronisation:
    - If some bits are lost explicit re-synchronization is needed
  - Advantage:
    - The pseudo-random sequence can be pre-computed in order to keep the impact of encryption to the end-to-end delay low
  - Drawbacks:
    - It is possible for an attacker to manipulate specific bits of the plaintext

    → However, this is not a problem, since an additional cryptographic mean is required for message integrity anyway

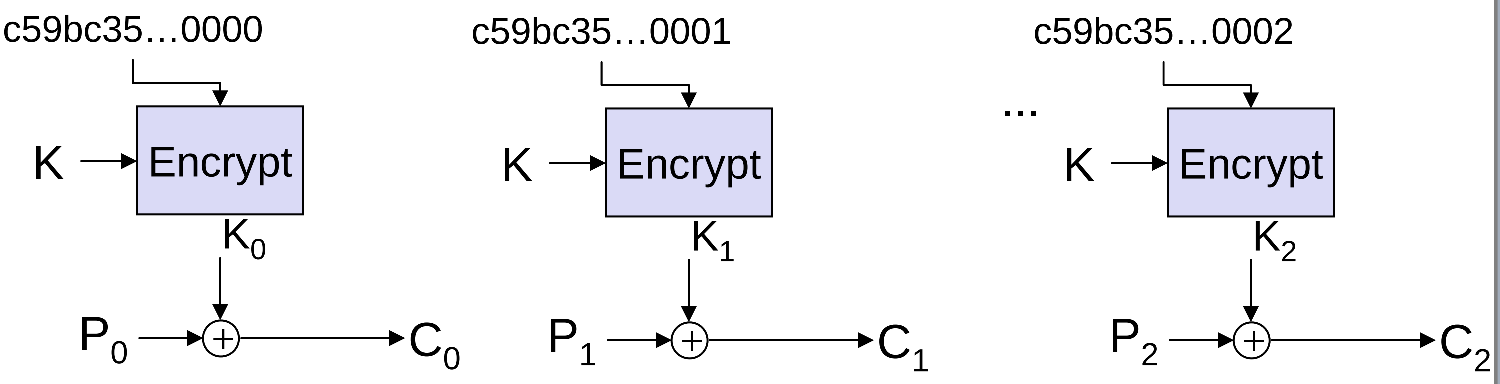
Symmetric Block Ciphers – Modes of Encryption - CTR (1)

- *Counter Mode (CTR)*
  - The block encryption algorithm is used to generate a key stream that depends on *K* and a counter function ctr_i .
  - The counter function can be simply an increment modulo 2^w, where *w* is a convenient register width, e.g.
    - $ctr_i = Nonce \,||\, i$
  - The counter function does not provide any security other than the uniqueness of the input of to the block cipher function *E*
  - The plaintext is XORed with the pseudo-random sequence to obtain the ciphertext and vice versa
  - Putting everything together:
    - $K_i = E(K, Nonce \,||\, i)$
    - $C_i = P_i \oplus K_i$

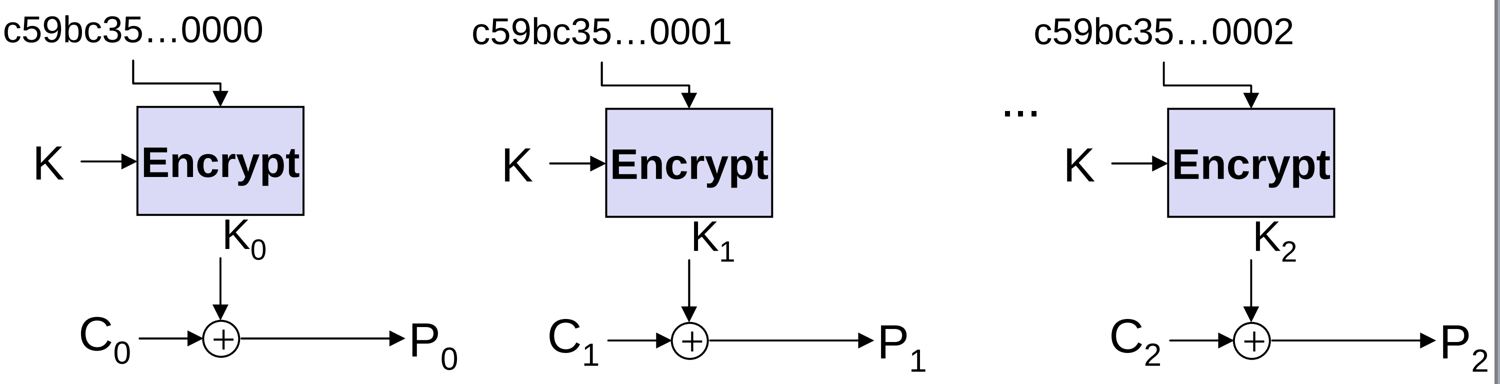
Symmetric Block Ciphers – Modes of Encryption - CTR (2)

CTR

Encrypt



Decrypt



Symmetric Block Ciphers – Modes of Encryption - CTR (3)

- Properties of CTR:
  - Error propagation:
    - Single bit errors result only in single bit errors ⇒ no error multiplication
  - Synchronisation:
    - If some bits are lost explicit re-synchronization is needed
  - Advantage:
    - The key stream can be pre-computed in order to keep the impact of encryption to the end-to-end delay low
    - The computation of the key stream can be even parallelized
  - Drawbacks:
    - It is possible for an attacker to manipulate specific bits of the plaintext

    → However, this is not a problem, since an additional cryptographic mean is required for message integrity anyway

Symmetric Block Ciphers - Algorithm Overview

- Some popular algorithms:
  - Data Encryption Standard (DES)
  - Triple encryption with DES: Triple-DES
  - Advanced Encryption Standard (AES)
  - Stream Cipher Algorithm RC4

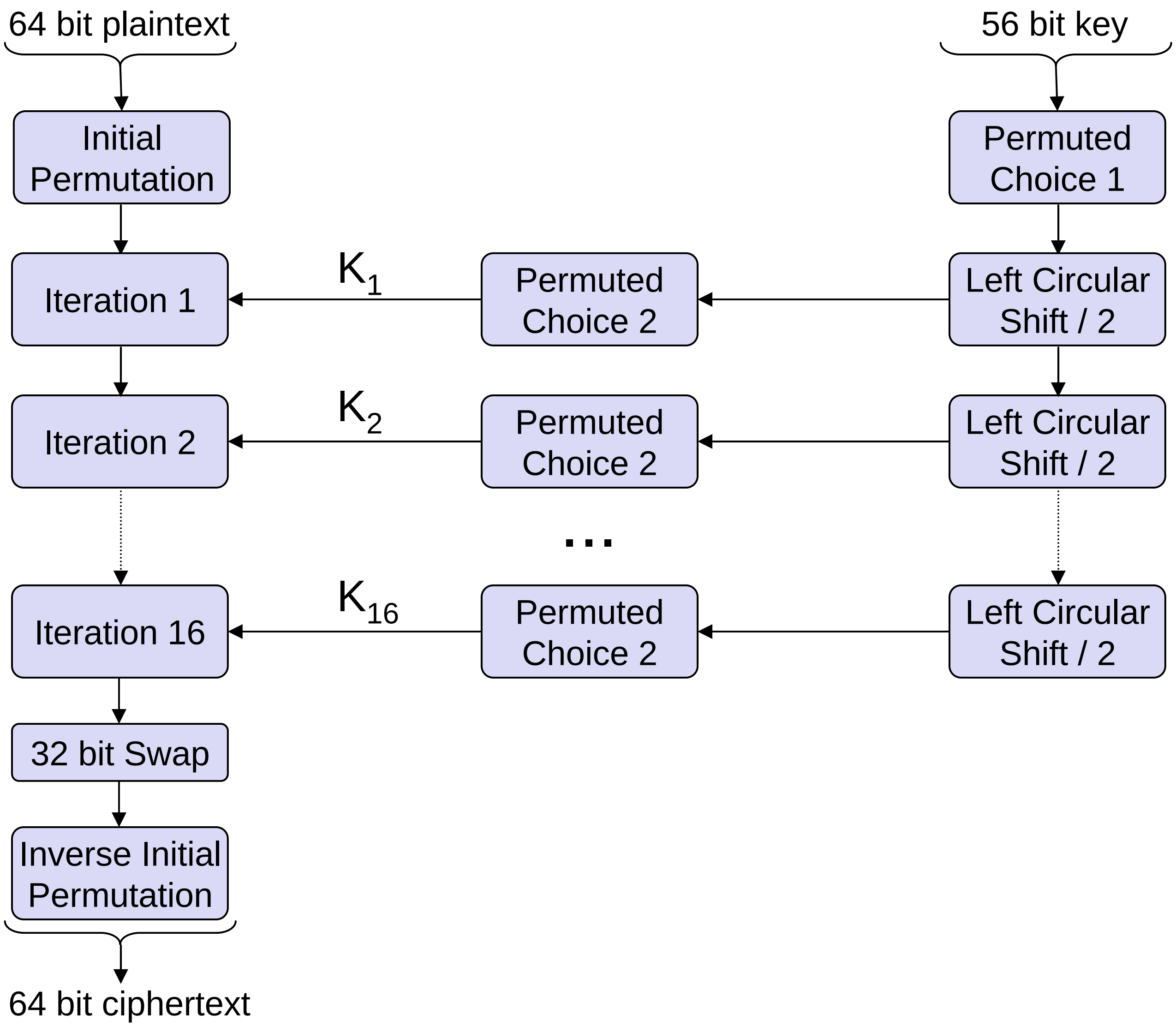
The Data Encryption Standard (DES) – History

- 1973 the National Bureau of Standards (NBS, now National Institute of Standards and Technology, NIST) issued a request for proposals for a national cipher standard, demanding the algorithm to:
  - provide a high level of security,
  - be completely specified and easy to understand,
  - provide security only by its key and not by its own secrecy,
  - be available to all users,
  - be adaptable for use in diverse applications,
  - be economically implementable in electronic devices,
  - be efficient to use,
  - be able to be validated, and
  - be exportable.
- None of the submissions to this first call came close to these criteria.
- In response to a second call, IBM submitted its' algorithm LUCIFER, a symmetric block cipher, which works on blocks of length 128 bit using keys of length 128 bit and that was the only promising candidate

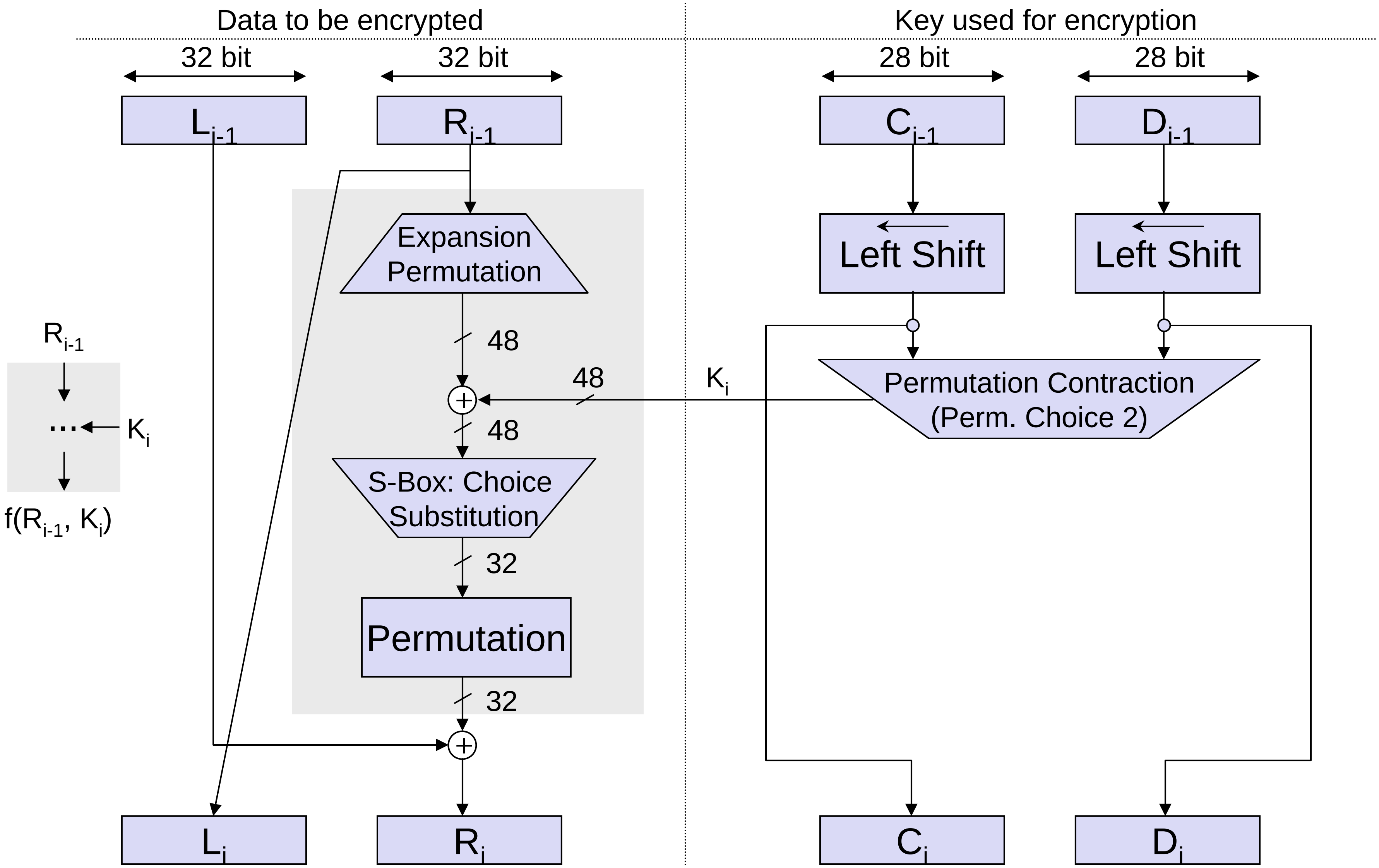
DES – History continued

- The NBS requested the help of the National Security Agency (NSA) in evaluating the algorithm's security:
  - The NSA reduced the block size to 64 bit, the size of the key to 56 bit and changed details in the algorithm's *substitution boxes.*
  - Many of the NSA's reasoning for these modifications became clear in the early 1990's, but raised great concern in the late 1970's.
- Despite all criticism the algorithm was adopted as "Data Encryption Standard" in the series of Federal Information Processing Standards in 1977 (FIPS PUB 46) and authorized for use on all unclassified government communications.
- DES has been widely adopted in the years to follow

DES – Algorithm Outline



DES – Single Iteration



DES – Security

- Main weakness is the key length:
  - As a 56 bit key can be searched in 10.01 hours when being able to perform 10^6 encryptions / µs (which is feasible today), DES can no longer be considered as sufficiently secure
- *Differential cryptanalysis:*
  - In 1990 E. Biham and A. Shamir published a cryptoanalysis method for DES
  - It looks specifically for differences in ciphertexts whose plaintexts have particular differences and tries to guess the correct key
  - The basic approach needs **chosen plaintext** together with its **ciphertext**
  - DES with 16 rounds is immune against this attack, as the attack needs 2^{47} chosen plaintexts or (when “converted” to a known plaintext attack) 2^{55} known plaintexts.
  - The designers of DES told in the 1990’s that they knew about this kind of attacks in the 1970’s and that the s-boxes were designed accordingly

Extending the Key-Length of DES by Multiple Encryption

- Triple encryption scheme, as proposed by W. Tuchman in 1979:
  - $C = E(K_3, D(K_2, E(K_1, P)))$
  - The use of the decryption function *D* in the middle allows to use triple encryption devices with peers that only own single encryption devices by setting $K_1 = K_2 = K_3$ (backwards compatibility with DES)
  - Triple encryption can be used with two (set $K_1 = K_3$) or three different keys
  - There are no known practical attacks against this scheme up to now
  - Drawback: the performance is only 1/3 of that of single encryption, so it should be a better idea to use a different cipher, which offers a bigger key-length right away

The Advanced Encryption Standard AES (1)

- Jan. 1997: the *National Institute of Standards and Technology (NIST)* of the USA announces *the AES development* effort.
  - The overall goal is to develop a Federal Information Processing Standard (FIPS) that specifies an encryption algorithm(s) capable of protecting sensitive government information well into the next century.
  - The algorithm(s) is expected to be used by the U.S. Government and, on a voluntary basis, by the private sector.
- Sep. 1997: formal *call for algorithms*, open to everyone on earth
  - AES would specify an unclassified, publicly disclosed encryption algorithm(s), available royalty-free, worldwide.
  - The algorithm(s) must implement symmetric key cryptography as a block cipher and (at a minimum) support block sizes of 128-bits and key sizes of 128-, 192-, and 256-bits.
- Aug. 1998: first AES candidate conference
  - NIST announces the selection of 15 candidate algorithms
  - Demand for public comments

The Advanced Encryption Standard AES (2)

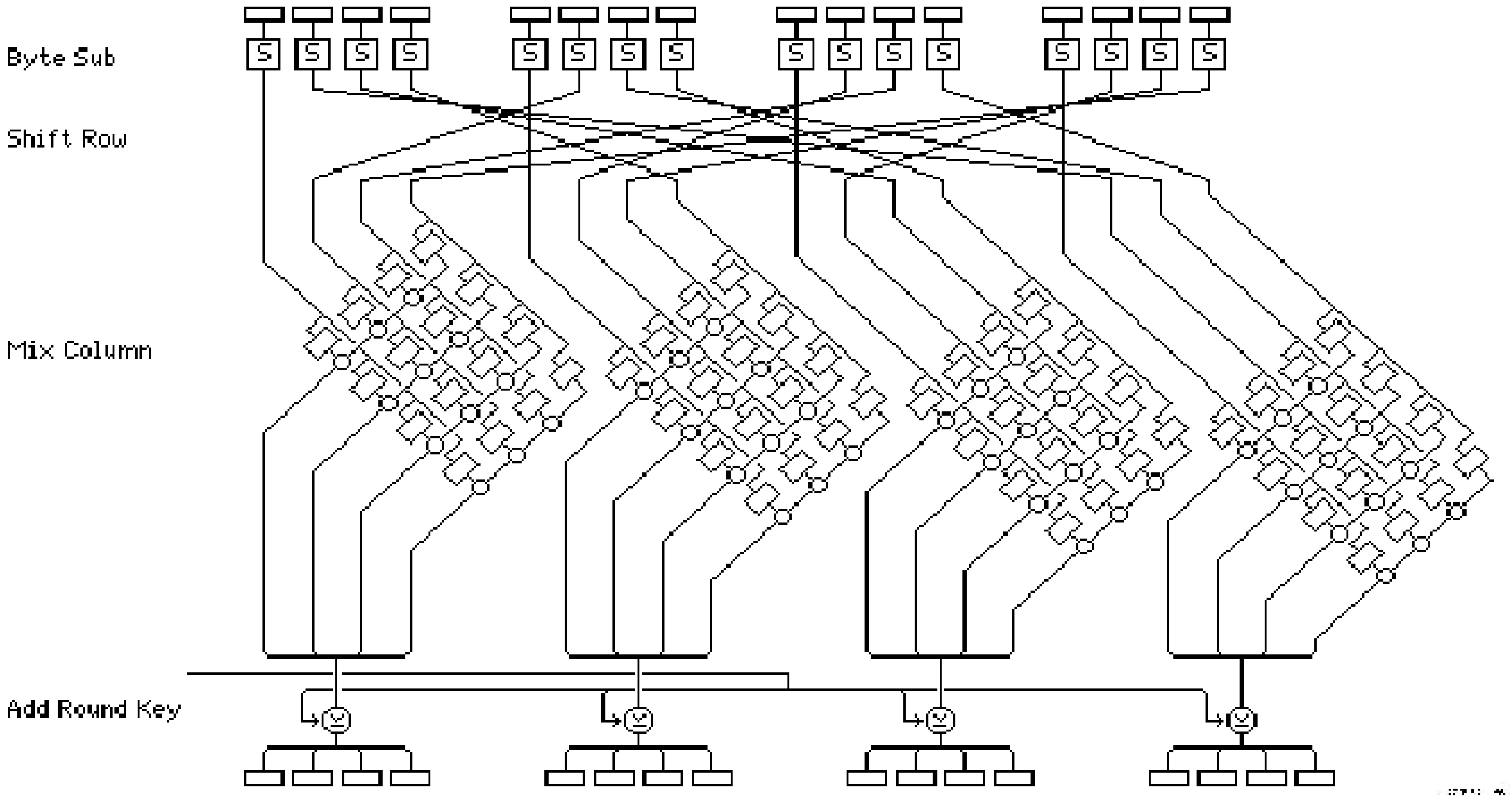
- Mar. 1999: second AES candidate conference
  - Discussion of results of the analysis conducted by the global cryptographic community on the candidate algorithms.
- April 1999:
  - Using the analyses and comments received, NIST selects five algorithms as finalist candidates: *MARS, RC6, Rijndael, Serpent,* and *Twofish*
  - Demand for public comments on any aspect of the finalists:
    - Cryptanalysis
    - Implementation issues
    - Intellectual property & Overall recommendations
- May 2000: third AES candidate conference
- October 2000: Rijndael is announced as NIST’s proposal for AES
- 28. February 2001: draft FIPS standard is published [AES01a]
- 29. May 2001: comment period ends
- 26. November 2001: official announcement of the AES standard

The Advanced Encryption Standard AES (3)

- Key and block lengths:
  - Key Length: 128, 192, or 256 bit
  - Block Length: 128, 192, or 256 bit
  - In the following only 128 bit is considered
- Number of rounds: 10 (for block and key size of 128 bit)
  - Rounds 1 - 9 make use of four different operations:
    - ByteSub: a non-linear byte substitution (basically an s-box)
    - ShiftRow: the rows of the state are cyclically shifted by various offsets
    - MixColumn: an operation based on polynomial algebra
    - RoundKey: a round-key is XORed with the state
  - Round 10 does not make use of the MixColumn operation

The Advanced Encryption Standard AES (4)

Structure of one Round in Rijndael



(source: “Rijndael”, a presentation by J. Daemen and V. Rijmen)

Properties of AES

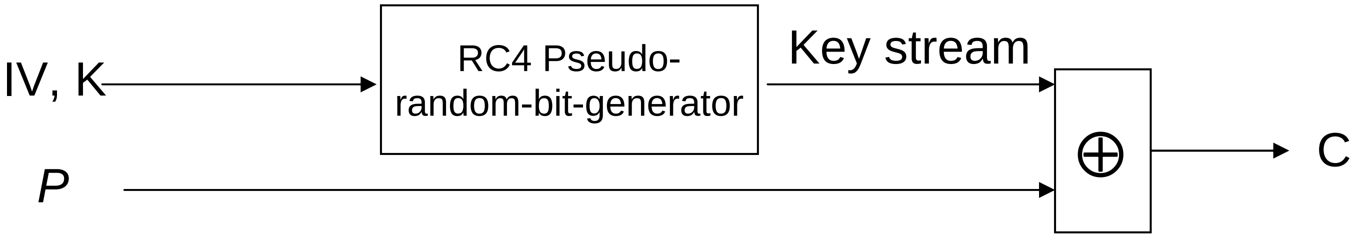
- No successful attacks known so far.
- Roughly 3 times the speed of DES (200 MBit/s vs. 80 MBit/s)
- Can be used in CBC of CTR modes in communication.
- Elegant algebraic structure: uses properties of GF(2^8).
- Highly parallel architecture.

The Stream Cipher Algorithm RC4

- RC4 is a *stream cipher* that has been invented by Ron Rivest in 1987
- It was proprietary until 1994 when someone posted it anonymously to a mailing list
- RC4 runs in OFB mode
- It uses a pseudo-random number generator (PRNG) for generating a “key stream” of arbitrary length
  - An initial state S_0 of a pseudo-random-bit generator is set using the key *K* and an initialization vector IV.
  - At each iteration a pseudo-random key k_j and a new state s_{j+1} is generated.
  - The k_j are concatenated together in order to build the key stream

RC4 Encryption

- The plaintext *P* is XORed with the key stream to obtain the cipher text and vice versa:
  - $C = P \oplus \mathrm{RC4}(IV, K)$
  - $P = C \oplus \mathrm{RC4}(IV, K)$



RC4 Encryption Block Diagram

Security of RC4

- RC4 uses a variable length key up to 2048 bit
  - The key serves as the seed for a pseudo-random-bit-generator
  - The variable key length of up to 2048 bit allows to make brute force attacks impractical (at least with the resources available in our universe)
- RSA Data Security, Inc. claims that RC4 is immune to differential and linear cryptanalysis
- Moreover, no small cycles in the generated pseudo-numbers are known:
  - "Analysis shows that the period of the cipher is overwhelmingly likely to be greater than 10^{100}." (*http://www.rsasecurity.com*)

(In)security of RC4 (1)

- The IV must be different for each message P_i
- Otherwise a simple **known plaintext attack** is possible:
  - Assume an attacker can guess a plain text P_1

    Note: It can happen quite often that the plain text can be guessed although it is encrypted. e.g. DNS requests, ARP requests. IP headers can be also easily guessed.
  - Let the same IV being used for encrypting P_1 occur again when a later message P_2 needs to be encrypted:

$$IV_1 = IV_2$$

  - Then $C_1 \oplus C_2 = P_1 \oplus \mathrm{RC4}(IV_1, K) \oplus P_2 \oplus \mathrm{RC4}(IV_2, K)$

$$= P_1 \oplus P_2 \oplus \mathrm{RC4}(IV_1, K) \oplus \mathrm{RC4}(IV_2, K)$$

$$= P_1 \oplus P_2 \oplus \mathrm{RC4}(IV_1, K) \oplus \mathrm{RC4}(IV_1, K)$$

$$= P_1 \oplus P_2$$

  - Then $P_2 = P_1 \oplus C_1 \oplus C_2$

(In)security of RC4 (2)

- Key length
  - RC4 with a 40 bit key was a default algorithm for the Secure Socket Layer (SSL) protocol, which has been designed to secure HTTP transfers.
  - 40-bit key length is not immune against brute-force attacks
  - Currently, the use of RC4 with 40-bits keys is deprecated

(In)security of RC4 (2)

- Fluhrer-Mantin-Shamir Attack [FMS01a] :
  - Over all possible RC4 keys, the statistics for the first few bytes of output keystream are strongly non-random, leaking information about the key.
  - If the long-term key and nonce are simply concatenated to generate the RC4 key, this long-term key can be discovered by analysing large number of messages encrypted with this key.
- The reaction of Ron Rivest to this attack was that applications using RC4 can defend against it by discarding the initial portion of the keystream (say the first 1024 bytes) before using it (RC4drop[n]).
- Klein’s attack:
  - There are correlations between the internal state of RC4 and the bytes in the key stream. As a result, the key can be recovered from a sufficiently large number of observed cipher text messages.
  - Used to break WEP encryption

Additional References

[AES01a] National Institute of Standards and Technology (NIST). *Specification for the Advanced Encryption Standard (AES).* Federal Information Processing Standards Publication, February 2001.

[DR97a] J. Daemen, V. Rijmen. *AES Proposal: Rijndael.* http://csrc.nist.gov/encryption/aes/rijndael/Rijndael.pdf, 1997.

[FMS01a] S. Fluhrer, I. Mantin, A. Shamir. *Weaknesses in the Key Scheduling Algorithm of RC4.* Eighth Annual Workshop on Selected Areas in Cryptography, August 2001.

[Riv01a] R. Rivest. *RSA Security Response to Weaknesses in Key Scheduling Algorithm of RC4*. http://www.rsa.com/rsalabs/technotes/wep.html, 2001.

[SIR01a] A. Stubblefield, J. Ioannidis, A. D. Rubin. *Using the Fluhrer, Mantin, and Shamir Attack to Break WEP.* AT&T Labs Technical Report TD-4ZCPZZ, August 2001.