INFO 6001

Revision 3

Information Security

Lesson 2

FANSHAWE

Housekeeping

- Discussion Forum – participate early each week!

- Emails – Subject lines of messages must be clear as to your course, section and purpose. If you are not getting a timely answer, this is why. Messages and posts must also be *professional* or they will be deleted. Please review our Communications Expectations – these are not just suggestions!

- Lab 2 – Creating & solving simple ciphers
- Lab 3 – VMWare Workstation is required for this work.

In the News…?

- OPEN TOPIC FOR DISCUSSION – IoT Security (Internet of Threats?)

- We all love smart homes..

- Zigbee is a popular low power mesh network for home IoT devices, but also a wonderful pivot into your home!
- https://blog.checkpoint.com/2020/02/05/the-dark-side-of-smart-lighting-check-point-research-shows-how-business-and-home-networks-can-be-hacked-from-a-lightbulb/

- IOT Security in General
- https://www.intertrust.com/blog/10-critical-concerns-in-iot-security/
- https://echoinnovateit.com/iot-security-issues/
- https://www.iotforall.com/5-worst-iot-hacking-vulnerabilities

- Industry Guidance – Includes C-I-A.

https://pages.checkpoint.com/ultimate-iot-security-guide.html

In the News...?

- OPEN TOPIC FOR DISCUSSION - Current Security Concerns:

- https://tech.co/news/data-breaches-2022-so-far (includes 2023 to date)
- https://www.techradar.com/features/top-data-breaches-and-cyber-attacks-of-2022
- https://securityintelligence.com/articles/cybersecurity-trends-ibm-predictions-2023/

- Stats check:
- https://www.fortinet.com/resources/cyberglossary/cybersecurity-statistics

Chapter 2

Symmetric Encryption and Message Confidentiality

This Week's Lesson

- Explain the concepts of **symmetric cryptography**
- Explain the difference between **cryptanalysis** and **brute-force attack**
- Understand the functionality of **DES**
- Present an overview of **AES**
- Explain the concepts of **randomness** and **unpredictability**
- Explain the difference between true random number generators, pseudorandom number generators, and pseudorandom functions
- Present an overview of **stream ciphers** and **RC4**
- Compare and contrast **ECB**, **CBC**, and **CFB** and counter modes of operation

FANSHAWE

Guiding Questions for this lesson…

1. What are the essential ingredients of a symmetric cipher?
2. What are the two basic functions used in encryption algorithms?
3. How many keys are required for two people to communicate in a symmetric cipher?
4. What is the difference between a block cipher and a stream cipher?
5. What is triple encryption? And why is the middle portion of 3DES a *decryption* instead of an *encryption*?
6. Who were Feistel and Rijndael?

Introduction / Overview

- Symmetric encryption is referred to as
  - Conventional encryption
  - Standard encryption
  - Secret-key encryption
  - **Single-key** or **private key** encryption
- This was the only type of encryption in place prior to the development of public-key encryption in the late 1970's
- We will look at a model for the symmetric encryption process and the algorithms used
- We also look at 3 block encryption algorithms (DES, triple DES or 3DES, and AES)
- Random and pseudorandom number generation
- Symmetric stream encryption (especially RC4)
- Block cipher modes of operation

Symmetric Encryption has 5 Ingredients…

1. **Plaintext**: the original message or data, the input into the algorithm
2. **Encrypt algorithm**: performs substitutions and transforms the plaintext (aka: the cipher)
3. **Secret key**: An input that defines the exact substitutions and transformations performed algorithm
4. **Ciphertext**: The scrambled message output
5. **Decrypt algorithm**: reverses the encryption process using the ciphertext and secret key to produce the original message

Two requirements for Symm. Encryption

1. A strong encryption **algorithm**
2. A **Secret key** that both parties have (and that they can keep secure!)

- They **key** has to stay secret – not the algorithm!
- Algorithm is not a secret
  - This makes it feasible for widespread use
  - Manufacturers can and have developed low-cost chip implementations of data encryption algorithms
  - These chips are widely available and incorporated into a number of products
- So the biggest drawback is our ability to keep the key a secret. (aka: the "**key distribution/management problem**")

In-class Discussion Questions

- There are standards! What off the record (OTR) standards exist?

- Is the signal protocol a standard?
- https://en.wikipedia.org/wiki/Signal_Protocol

- Should Telegram have made their own OTR protocol. Good or bad idea? Your thoughts…..?

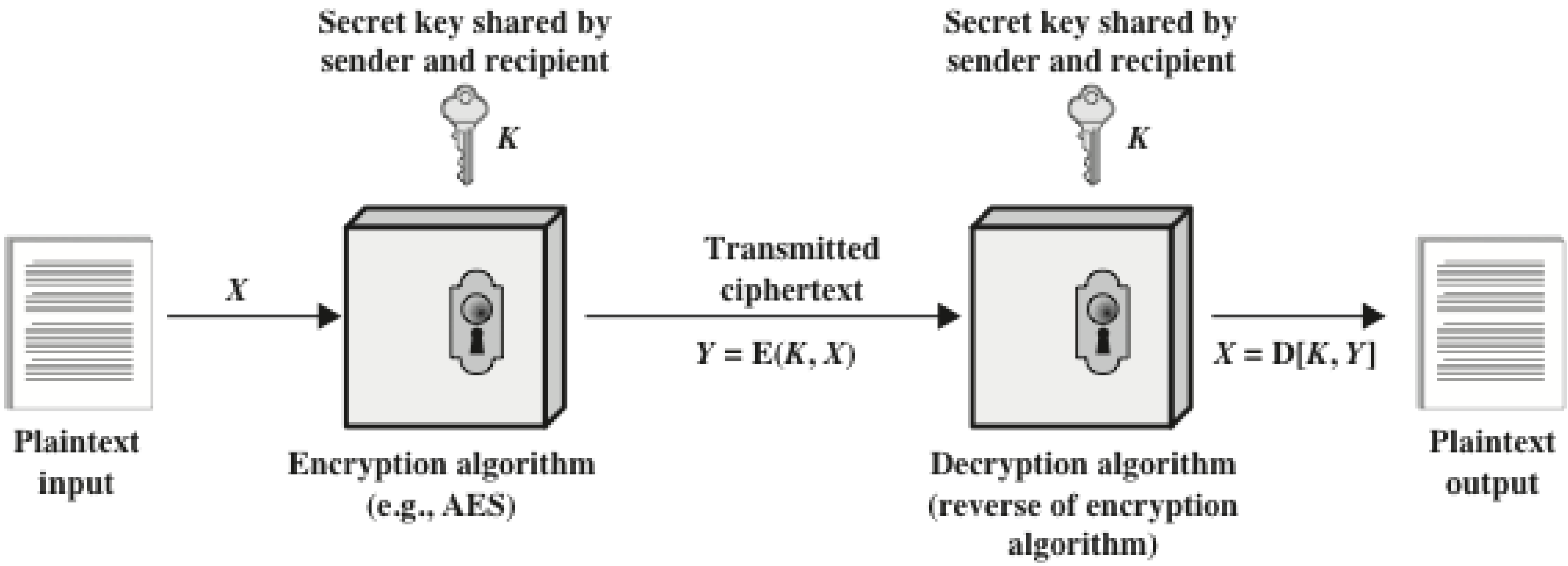


Figure 2.1 Simplified Model of Symmetric Encryption

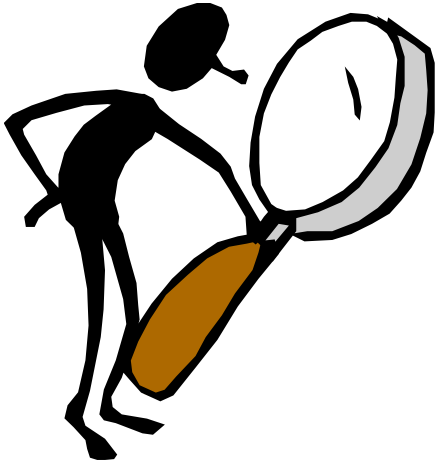
Classifying Cryptographic Systems: Three Dimensions

1. **The type of operations used for transforming plaintext to ciphertext**
   - Substitution
     - Each element in the plaintext is mapped into another element
   - Transposition
     - Elements in the plaintext are rearranged
     - Fundamental requirement is that no information be lost
   - Product systems
     - Involve multiple stages of substitutions and transpositions
2. **The number of keys used**
   - Referred to as symmetric, single-key, secret-key, or conventional encryption if both sender and receiver use the same key
   - Referred to as asymmetric, two-key (key-pair), or public-key encryption if the sender and receiver each use a different key
3. **The way in which the plaintext is processed**
   - *Block cipher* processes the input one block of elements at a time, producing an output block for each input block
   - *Stream cipher* processes the input elements *continuously*, producing output one element at a time, as it goes along

Cryptanalysis

- Cryptanalysis is the decryption and analysis of codes, ciphers or encrypted text. Cryptanalysis uses mathematical formulas to search for algorithm vulnerabilities and break into cryptography or information security systems.

- An encryption scheme is ***computationally secure*** if the ciphertext generated by the scheme meets one or both of the following criteria:
  - The cost of breaking the cipher exceeds the value of the encrypted information;
  - The time required to break the cipher exceeds the useful lifetime of the information.



Brute Force attack

- Involves trying every possible key until an intelligible translation of the ciphertext into plaintext is obtained
- On average, half of all possible keys must be tried to achieve success
- Unless known plaintext is provided, the analyst must be able to recognize plaintext as plaintext
- To supplement the brute-force approach
  - Some degree of knowledge about the expected plaintext is needed
  - Some means of automatically distinguishing plaintext from garble is also needed

Type of Attack	Known to Cryptanalyst
Ciphertext only	•Encryption algorithm •Ciphertext to be decoded
Known plaintext	•Encryption algorithm •Ciphertext to be decoded •One or more plaintext-ciphertext pairs formed with the secret key
Chosen plaintext	•Encryption algorithm •Ciphertext to be decoded •Plaintext message chosen by cryptanalyst, together with its corresponding ciphertext generated with the secret key
Chosen ciphertext	•Encryption algorithm •Ciphertext to be decoded •Purported ciphertext chosen by cryptanalyst, together with its corresponding decrypted plaintext generated with the secret key
Chosen text	•Encryption algorithm •Ciphertext to be decoded •Plaintext message chosen by cryptanalyst, together with its corresponding ciphertext generated with the secret key •Purported ciphertext chosen by cryptanalyst, together with its corresponding decrypted plaintext generated with the secret key

Table 2.1 Types of Attacks on Encrypted Messages

Block Cipher Design Principles

(most block ciphers will follow a Feistel Structure)

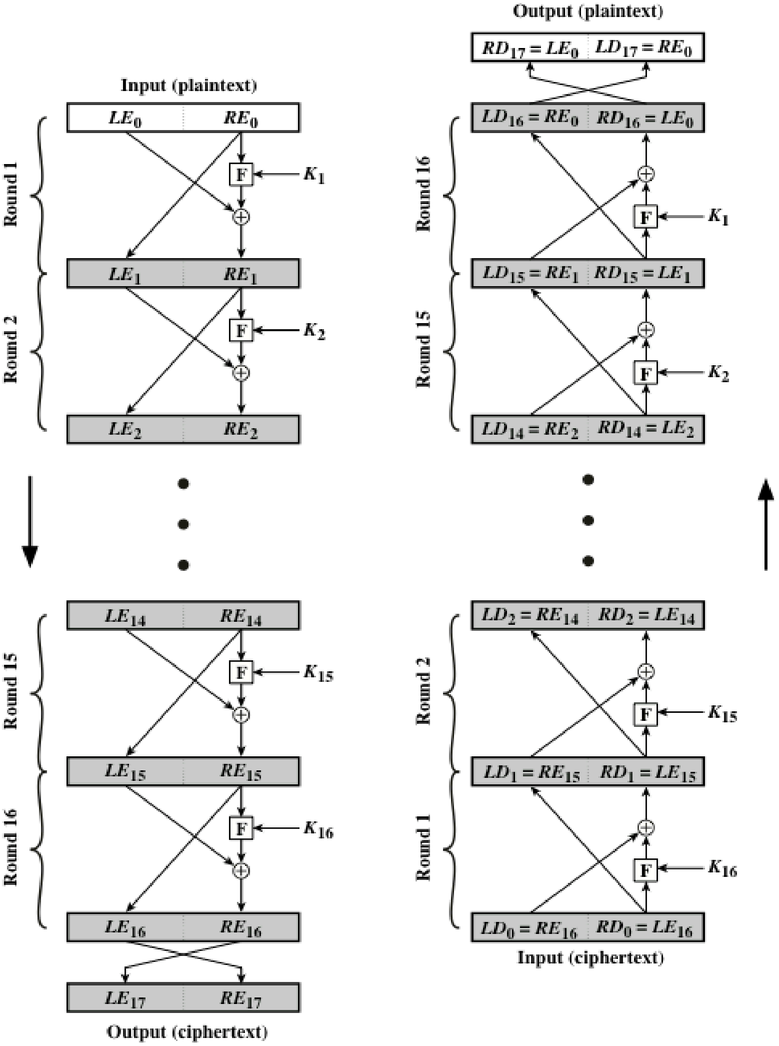


Figure 2.2 Feistel Encryption and Decryption (16 rounds)

Feistel Cipher Design Elements

Block size

- Larger block sizes mean greater security but reduced encryption/decryption speed

Key size

- Larger key size means greater security but may decrease encryption/decryption speed

Number of rounds

- The essence of a symmetric block cipher is that a single round offers inadequate security but that multiple rounds offer increasing security

Subkey generation algorithm

- Greater complexity in this algorithm should lead to greater difficulty of cryptanalysis

Round function

- Greater complexity generally means greater resistance to cryptanalysis

Fast software encryption/decryption

- In many cases, encryption is embedded in applications or utility functions in such a way as to preclude a hardware implementation; accordingly, the seed of execution of the algorithm becomes a concern

Ease of analysis

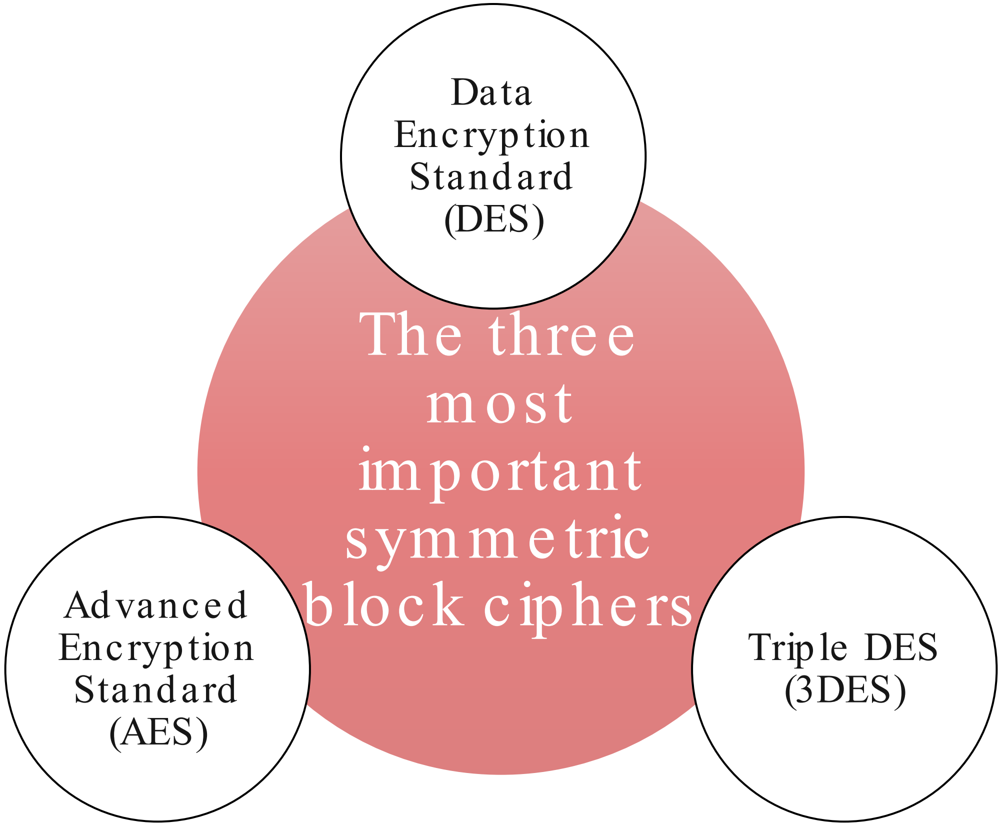
- If the algorithm can be concisely and clearly explained, it is easier to analyze that algorithm for cryptanalytic vulnerabilities and therefore develop a higher level of assurance as to its strength

Symmetric Block Encryption Algorithms

FANSHAWE

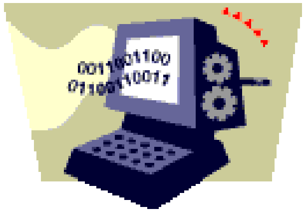
Symmetric Block encryption algorithms

- Block cipher
  - The most commonly used symmetric encryption algorithms
  - Processes the plaintext input in fixed-sized blocks and produces a block of ciphertext of equal size for each plaintext block



Data Encryption Standard (DES)

- Most widely used encryption scheme
- Issued in 1977 as Federal Information Processing Standard 46 (FIPS 46) by the National Institute of Standards and Technology (NIST)
- The algorithm itself is referred to as the Data Encryption Algorithm (DEA)



DES algorithm

- Description of the algorithm:
  - Plaintext is 64 bits in length
  - Key is 56 bits in length
  - Structure is a minor variation of the Feistel network
  - There are 16 rounds of processing
  - Process of decryption is essentially the same as the encryption process
- The strength of DES:
  - Concerns fall into two categories
    - The algorithm itself
      - Refers to the possibility that successful cryptanalysis is possible by exploiting the characteristics of the algorithm
    - The use of a 56-bit key
      - Speed of commercial, off-the-shelf processors threatens the security

Table 2.2

Average Time Required for Exhaustive Key Search

Key size (bits)	Cipher	Number of Alternative Keys	Time Required at 10^9 decryptions/s	Time Required at 10^{13} decryptions/s
56	DES	$2^{56} \approx 7.2 \times 10^{16}$	2^{55} ns = 1.125 years	1 hour
128	AES	$2^{128} \approx 3.4 \times 10^{38}$	2^{127} ns = 5.3×10^{21} years	5.3×10^{17} years
168	Triple DES	$2^{168} \approx 3.7 \times 10^{50}$	2^{167} ns = 5.8×10^{33} years	5.8×10^{29} years
192	AES	$2^{192} \approx 6.3 \times 10^{57}$	2^{191} ns = 9.8×10^{40} years	9.8×10^{36} years
256	AES	$2^{256} \approx 1.2 \times 10^{77}$	2^{255} ns = 1.8×10^{60} years	1.8×10^{56} years

Key size of 384 bits = 6.2×10^{94} years
Key size 512 = 2.1×10^{133} years

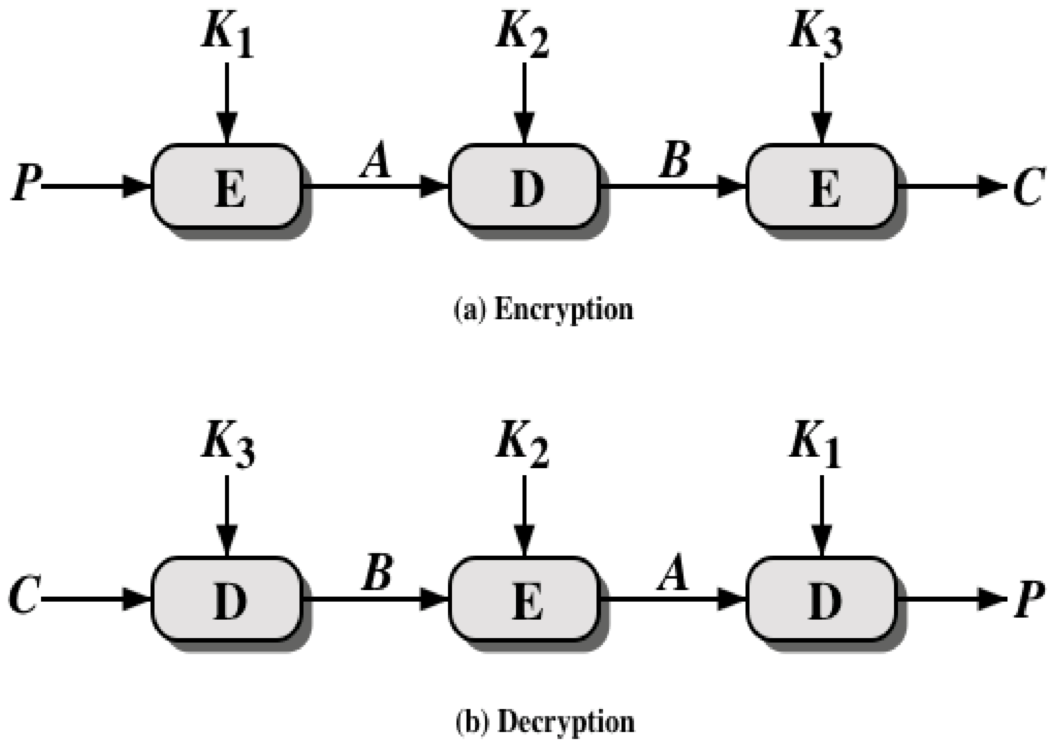


Figure 2.3 Triple DES

3DES guidelines

- FIPS 46-3 includes the following guidelines for 3DES:
  - 3DES is the FIPS-approved symmetric encryption algorithm of choice
  - The original DES, which uses a single 56-bit key, is permitted under the standard for legacy systems only; new procurements should support 3DES
  - Government organizations with legacy DES systems are encouraged to transition to 3DES
  - It is anticipated that 3DES and the Advanced Encryption Standard (AES) will coexist as FIPS-approved algorithms, allowing for a gradual transition to AES

Advanced encryption standard (AES)

- In 1997 NIST issued a call for proposals for a new AES:
  - Should have a security strength equal to or better than 3DES and significantly improved efficiency
  - Must be a symmetric block cipher with a block length of 128 bits and support for key lengths of 128, 192, and 256 bits
  - Evaluation criteria included security, computational efficiency, memory requirements, hardware and software suitability, and flexibility
- NIST selected Rijndael as the proposed AES algorithm
  - FIPS PUB 197 (in 2001)
  - Developers were two cryptographers from Belgium: Dr. Joan Daemen and Dr. Vincent Rijmen

128 bits means 16 bytes. Arranged in a grid.

Byte 0	Byte 04	Byte 08	Byte 12
Byte 01	Byte 05	Byte 09	Byte 13
Byte 02	Byte 06	Byte 10	Byte 14
Byte 03	Byte 07	Byte 11	Byte 15

Remember: ZERO is a number!

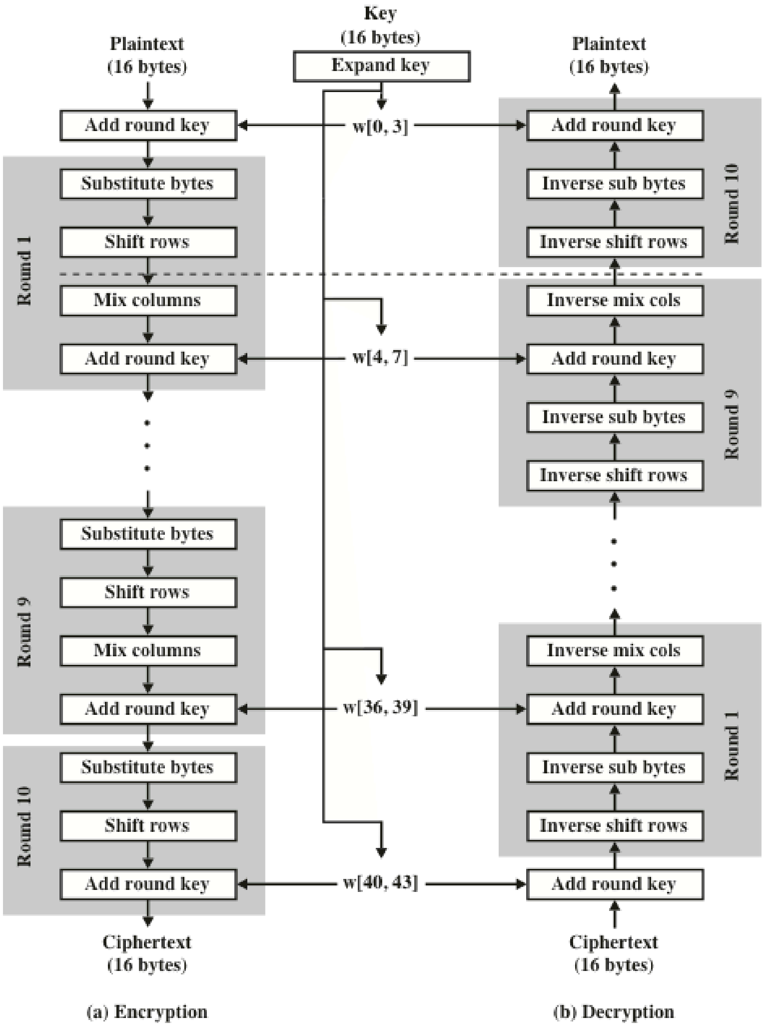


Figure 2.4 AES Encryption and Decryption

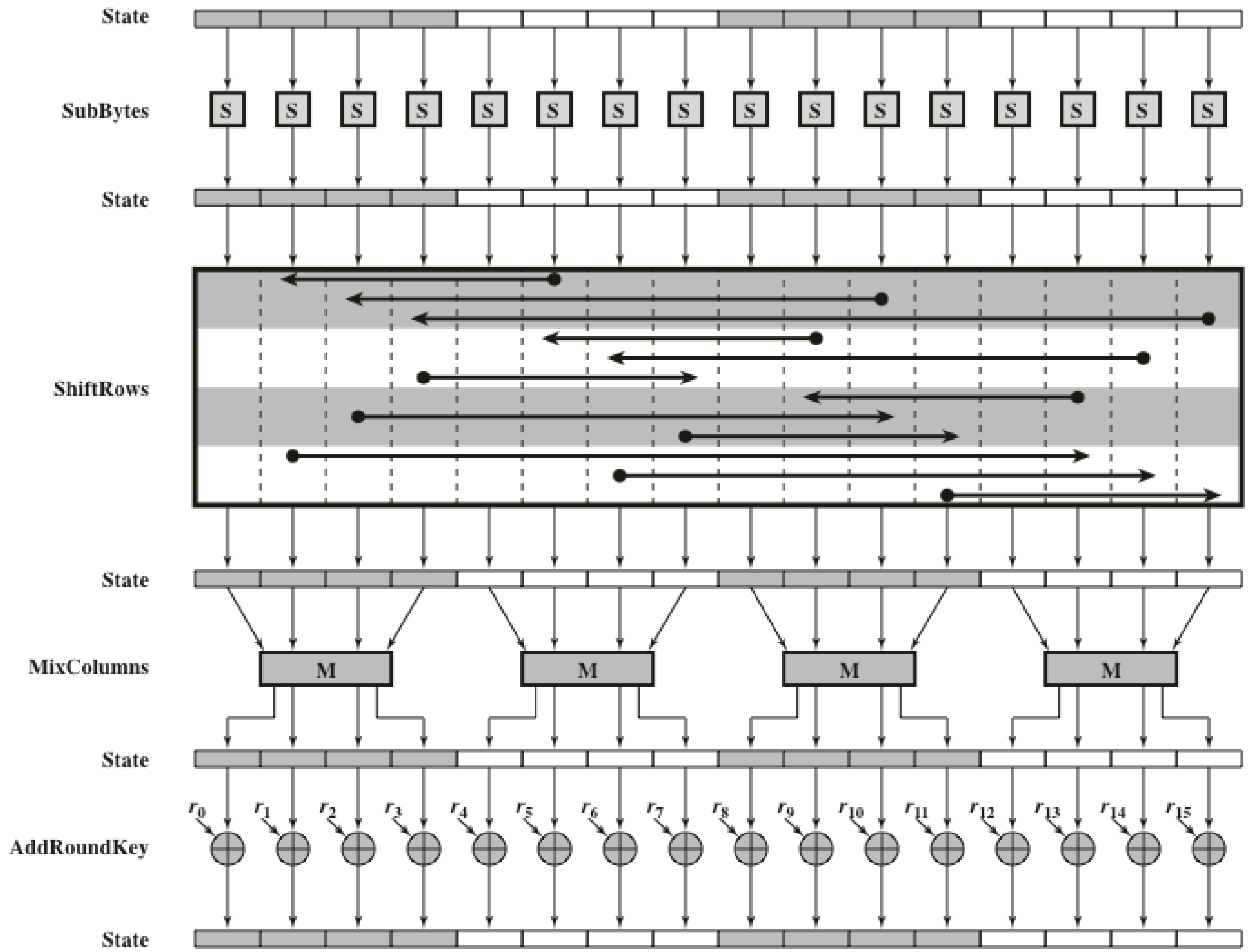


Figure 2.5 AES Encryption Round

Random and Pseudorandom Numbers

Random and pseudorandom Numbers

- A number of network security algorithms based on cryptography make use of random numbers
  - Examples:
    - Generation of keys for the **RSA** public-key encryption algorithm and other public-key algorithms
    - Generation of a symmetric key for use as a **temporary** session key; used in a number of networking applications such as Transport Layer Security (TLS), Wi-Fi, e-mail security, and IP security
    - In a number of key distribution scenarios, such as Kerberos, random numbers are used for handshaking to prevent replay attacks

Two distinct (and not necessarily compatible) requirements for a sequence of random numbers are:

1. Randomness
2. Unpredictability

Discussion - Randomness

- How we get randomness!
- https://www.youtube.com/watch?v=GtOt7EBNEwQ

- What sort of seeds could be used for a PRNG?

Randomness

- The following criteria are used to validate that a sequence of numbers is random:

Uniform distribution

- The distribution of bits in the sequence should be uniform
- Frequency of occurrence of ones and zeros should be approximately the same

Independence

- No one subsequence in the sequence can be inferred from the others
- There is no test to "prove" independence
- The general strategy is to apply a number of tests until the confidence that independence exists is sufficiently strong

Unpredictability

- In applications such as reciprocal authentication and session key generation, the requirement is not so much that the sequence of numbers be statistically random but that the successive members of the sequence are unpredictable
- With "true" random sequences, each number is statistically independent of other numbers in the sequence and therefore unpredictable
- Care must be taken that an opponent not be able to predict future elements of the sequence on the basis of earlier elements

TRNGs, PRNGs, and PRFs

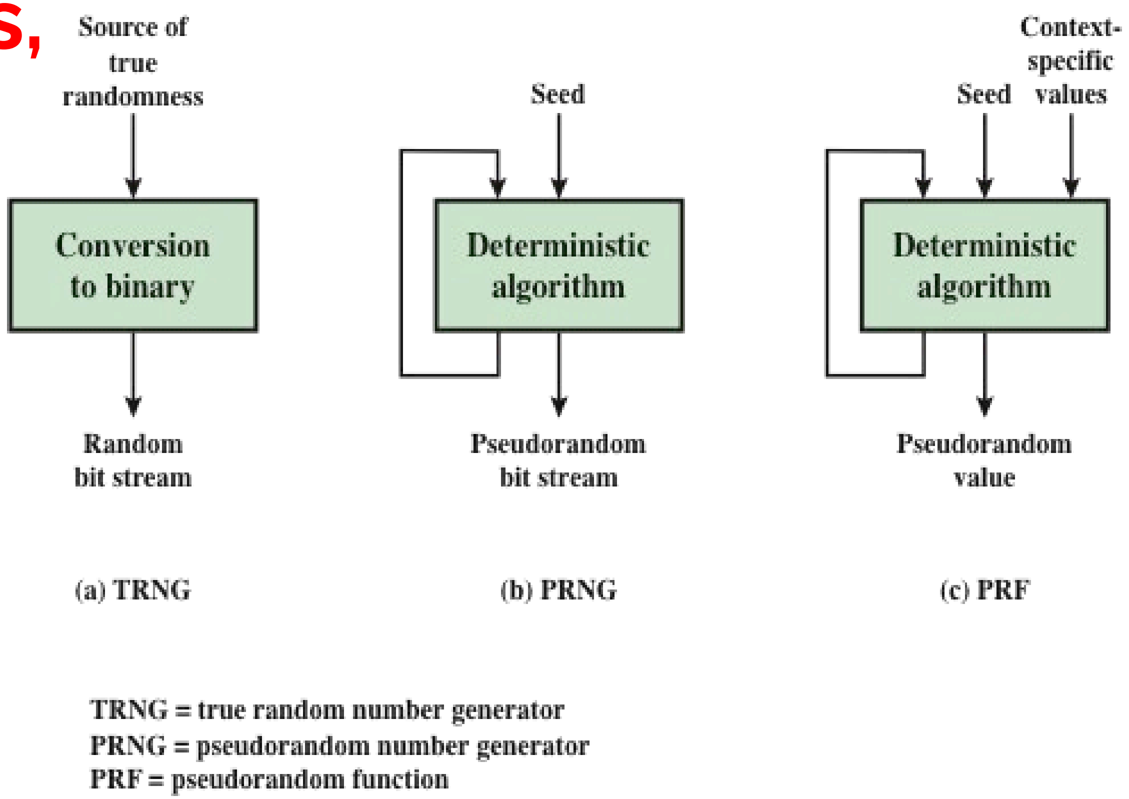


Figure 2.6 Random and Pseudorandom Number Generators

Algorithm design

Purpose-built algorithms	• Designed specifically and solely for the purpose of generating pseudorandom bit streams
Algorithms based on existing cryptographic algorithms	• Cryptographic algorithms have the effect of randomizing input • Can serve as the core of PRNGs
Three broad categories of cryptographic algorithms are commonly used to create PRNGs:	• Symmetric block ciphers • Asymmetric ciphers • Hash functions and message authentication codes

Stream Ciphers and RC4

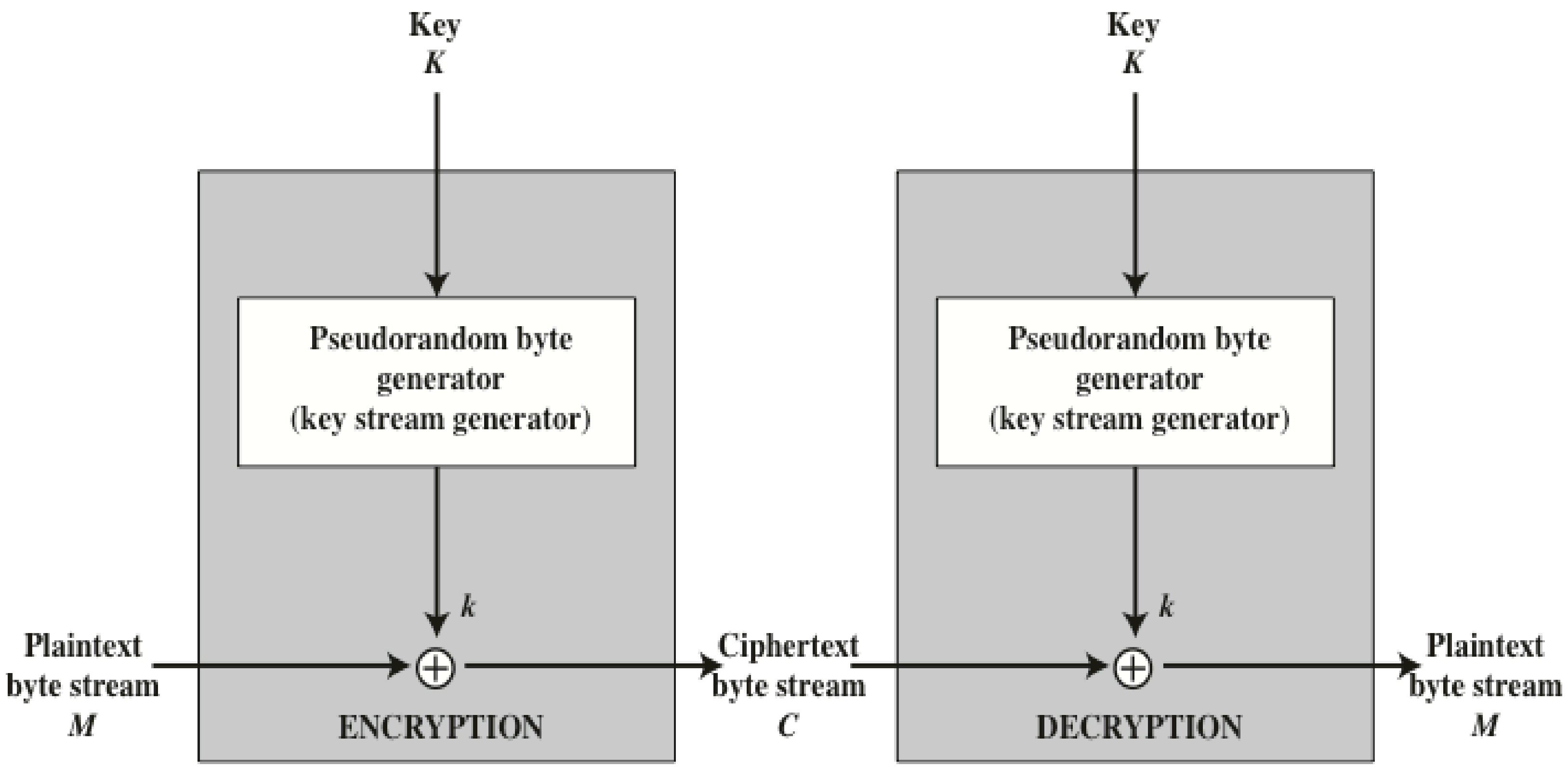


Figure 2.7 Stream Cipher Diagram

Stream Cipher design considerations

- The encryption sequence should have a large period
  - The longer the period of repeat, the more difficult it will be to do cryptanalysis
- The keystream should approximate the properties of a true random number stream as close as possible
  - The more random-appearing the keystream is, the more randomized the ciphertext is, making cryptanalysis more difficult
- The pseudorandom number generator is conditioned on the value of the input key
  - To guard against brute-force attacks, the key needs to be sufficiently long
  - With current technology, a key length of at least 128 bits is desirable

RC4 algorithm

- A stream cipher designed in 1987 by Ron Rivest for RSA Security
- It is a variable key-size stream cipher with byte-oriented operations
- The algorithm is based on the use of a random permutation
- Is used in the Secure Sockets Layer/ Transport Layer Security (SSL/ TLS) standards that have been defined for communication between Web browsers and servers
- Also used in the Wired Equivalent Privacy (WEP) protocol and the newer WiFi Protected Access (WPA) protocol that are part of the IEEE 802.11 wireless LAN standard

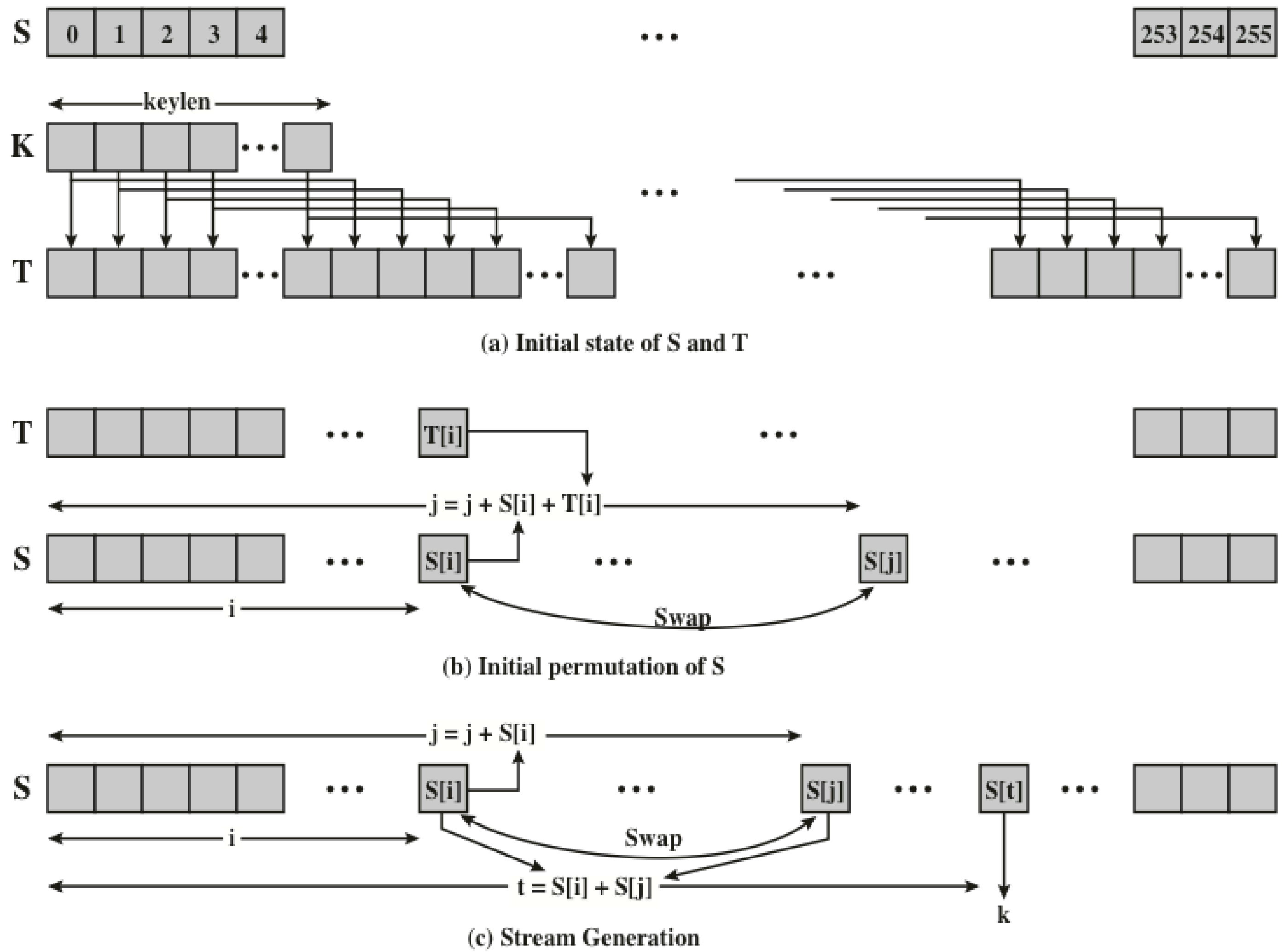


Figure 2.8 RC4

Cipher Block Modes of Operation

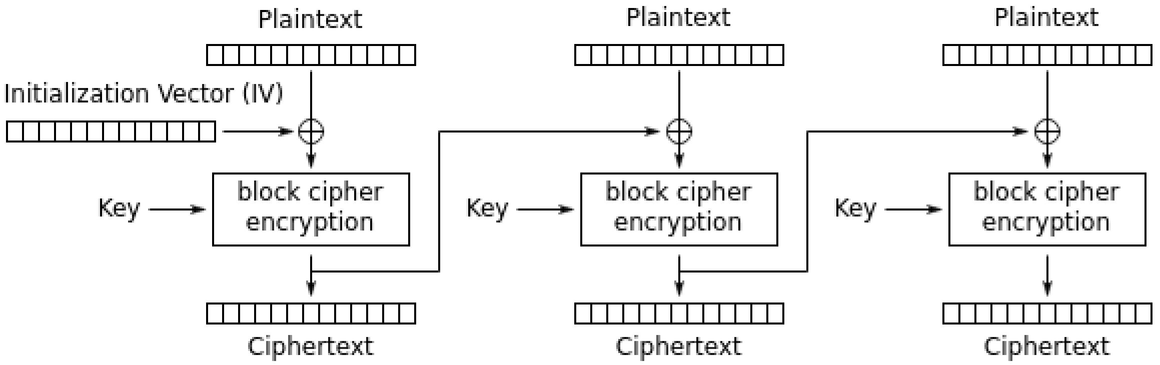
FANSHAWE

Cipher block Modes of Operation

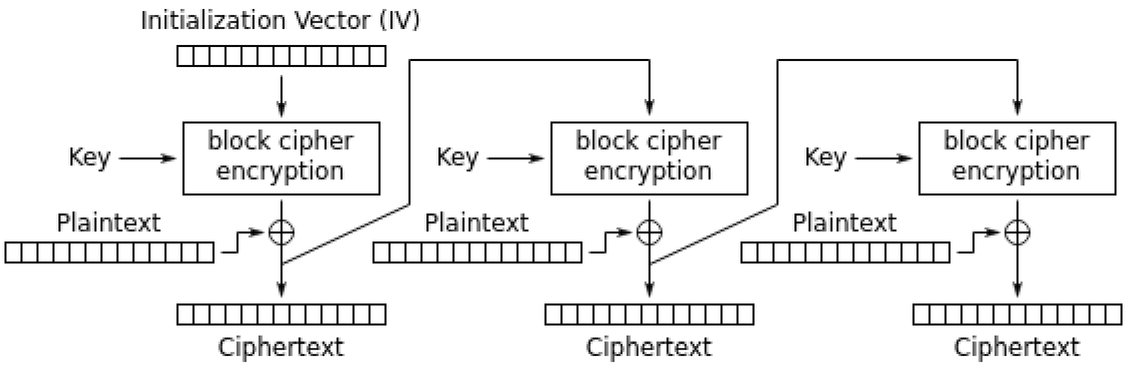
- A symmetric block cipher processes one block of data at a time
  - In the case of DES and 3DES, the block length is b=64 bits
  - For AES, the block length is b=128
  - For longer amounts of plaintext, it is necessary to break the plaintext into b-bit blocks, padding the last block if necessary
- Five modes of operation have been defined by NIST
  - Intended to cover virtually all of the possible applications of encryption for which a block cipher could be used
  - Intended for use with any symmetric block cipher, including triple DES and AES

Electronic Codebook Mode (ECB)

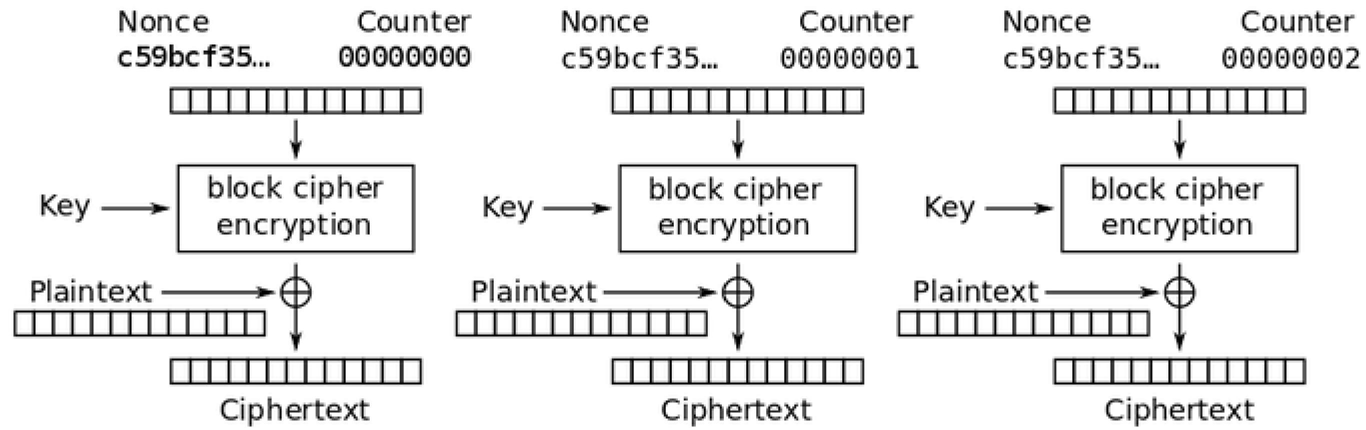
- Plaintext is handled *b* bits at a time and each block of plaintext is encrypted using the same key
- The term "codebook" is used because, for a given key, there is a unique ciphertext for every *b*-bit block of plaintext
  - One can imagine a gigantic codebook in which there is an entry for every possible *b*-bit plaintext pattern showing its corresponding ciphertext
- With ECB, if the same *b*-bit block of plaintext appears more than once in the message, it always produces the same ciphertext
  - Because of this, for lengthy messages, the ECB mode may not be secure
  - If the message is highly structured, it may be possible for a cryptanalyst to exploit these regularities



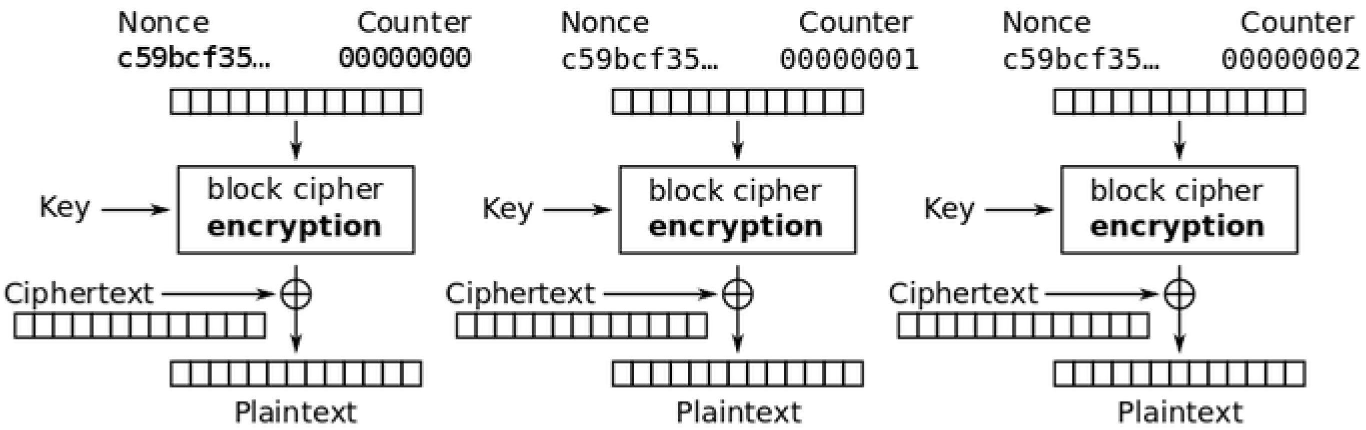
Cipher Block Chaining (CBC) mode encryption



Cipher Feedback (CFB) mode encryption



Counter (CTR) mode encryption



Counter (CTR) mode decryption

Advantages of CTR mode

- Hardware efficiency
  - Encryption/ decryption can be done in parallel on multiple blocks of plaintext or ciphertext
  - Throughput is only limited by the amount of parallelism that is achieved
- Software efficiency
  - Because of the opportunities for parallel execution, processors that support parallel features can be effectively utilized
- Preprocessing
  - The execution of the underlying encryption algorithm does not depend on input of the plaintext or ciphertext --- when the plaintext or ciphertext input is presented, the only computation is a series of XORs, greatly enhancing throughput
- Random access
  - The *i*th block of plaintext or ciphertext can be processed in random-access fashion
- Provable security
  - It can be shown that CTR is at least as secure as the other modes discussed in this section
- Simplicity
  - Requires only the implementation of the encryption algorithm and not the decryption algorithm

Summary

- Symmetric encryption principles
  - Cryptography
  - Cryptanalysis
  - Feistel cipher structure
- Symmetric block encryption algorithms
  - Data encryption standard
  - Triple DES
  - Advanced encryption standard
- Random and pseudorandom numbers
  - The use of random numbers
  - TRNGs, PRNGs, PRFs
  - Algorithm design
- Stream ciphers and RC4
  - Stream cipher structure
  - RC4 algorithm
- Cipher block modes of operation
  - ECB
  - CBC
  - CFB
  - CTR

References and Reminders

- Read the key terms at the end of chapter 2 (p.73)

- Please keep up with your discussion posts! Your discussion forum post for week 2 can be related to any element of the course so far, but if you need help, you could write about one of the news articles from today's lesson or your answers (*with sources*) to the guiding questions for this lesson.

- Next week we will be discussing
  - **Public Key Cryptography and Message Authentication**