COMPUTER SECURITY

PRINCIPLES AND PRACTICE

SECOND EDITION

William Stallings | Lawrie Brown

Chapter 2

Cryptographic Tools

Contents

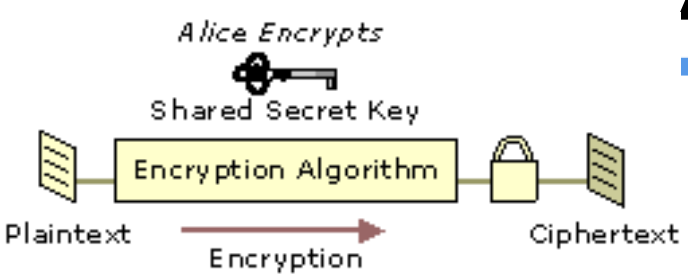
- **Confidentiality with Symmetric Encryption**
- Message Authentication and **Hash Functions**
- Public-Key Encryption
- Digital Signatures and Key Management
- Random and Pseudorandom Numbers

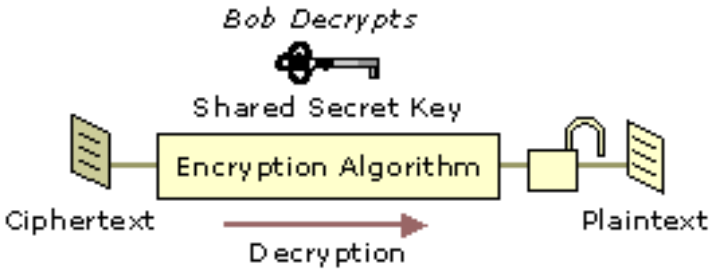
Learning Objectives

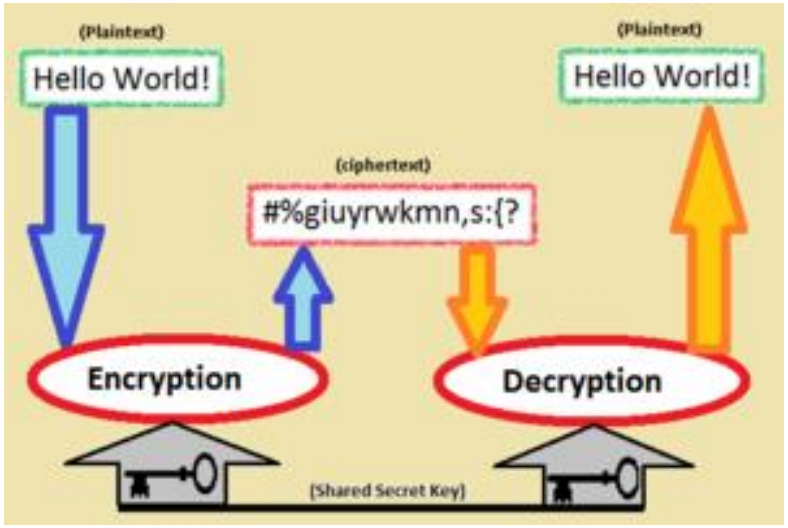
After studying this chapter, you should be able to:

- **Explain the basic operation of symmetric block encryption algorithms**
- **Discuss the use of secure hash functions for message authentication**
- **List other applications of secure hash functions**
- **Compare and contrast block encryption and secure hash functions**

Cryptography







- **Symmetric encryption**

From Computer Desktop Encyclopedia
© 2004 The Computer Language Co. Inc.

USING EXCLUSIVE OR (XOR) IN CRYPTOGRAPHY

XOR LOGIC

XOR Symbol ⊕

0 XOR 0 = 0	Same Bits
1 XOR 1 = 0	Same Bits
1 XOR 0 = 1	Different Bits
0 XOR 1 = 1	Different Bits

ENCRYPT

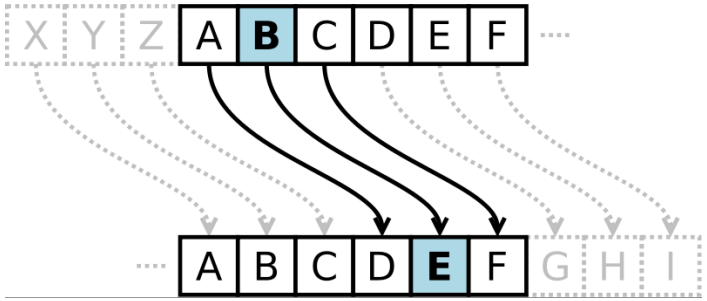
0 0 1 1 0 1 0 1	Plaintext
⊕ 1 1 1 0 0 0 1 1	Secret Key
= 1 1 0 1 0 1 1 0	Ciphertext

DECRYPT

1 1 0 1 0 1 1 0	Ciphertext
⊕ 1 1 1 0 0 0 1 1	Secret Key
= 0 0 1 1 0 1 0 1	Plaintext

Caesar Cipher & Vigenère cipher

- A simple substitution cipher, known as Caesar cipher
  - Replace each letter with the one "three over" in the alphabet.
    - Plain: **meet me after the toga party**
    - Cipher: **PHHW PH DIWHU WKH WRJD SDUWB**
  - No key, just one mapping (translation)
    - Plain: **ABCDEFGHIJKLMNOPQRSTUVWXYZ**
    - Cipher: **DEFGHIJKLMNOPQRSTUVWXYZABC**
  - $c_i = E(p_i) = (p_i + 3) \bmod 26$;
    $p_i = D(c_i) = (c_i - 3) \bmod 26$



- Polyalphabetic ciphers, known as Vigenère cipher

Key:	deceptivedeceptivedeceptive
Plaintext:	wearediscoveredsaveyourself
Cipheretxt:	ZICVTWQNGRZGVTWAVZHCQYGLMGJ

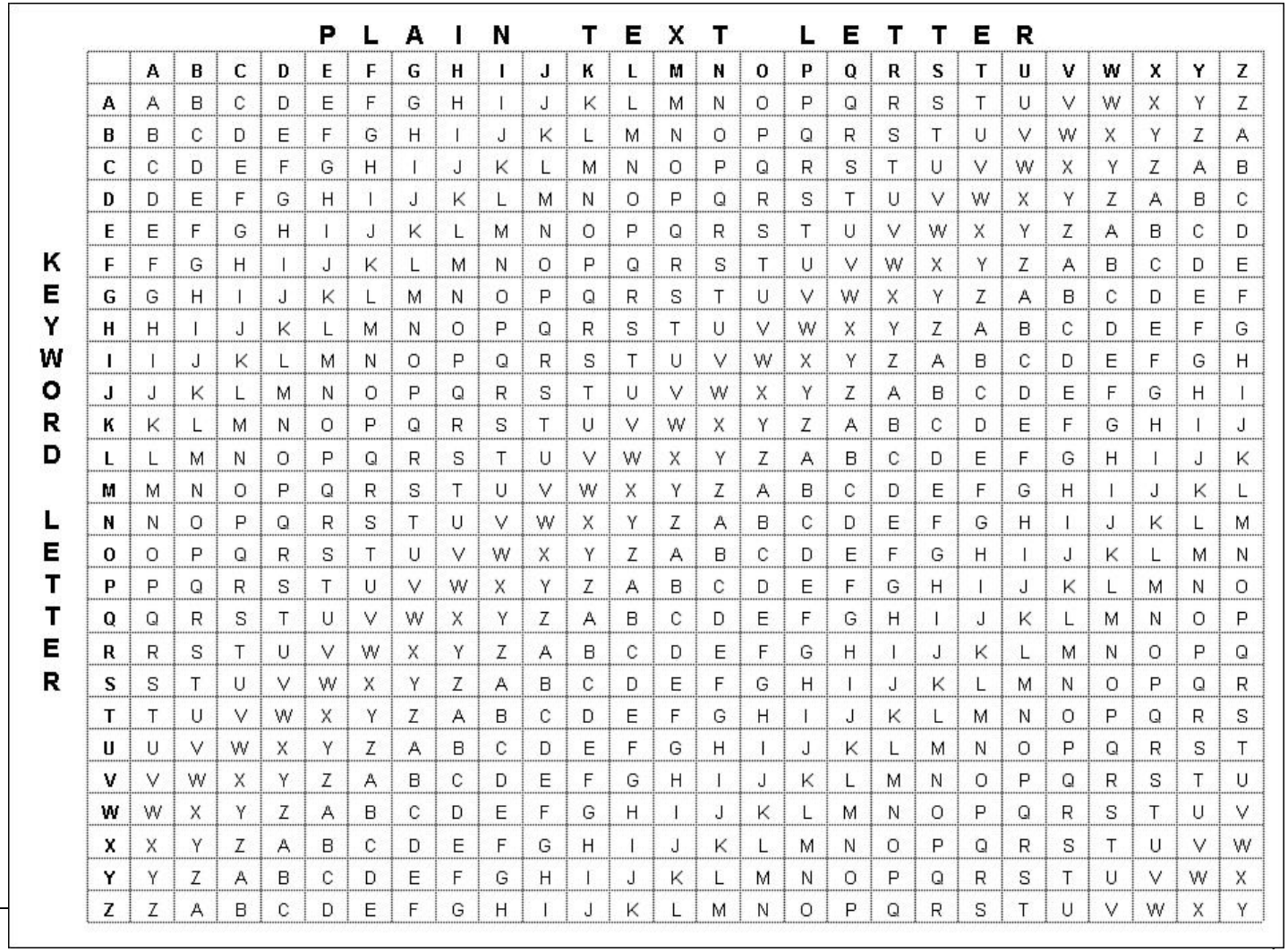
Vigenère cipher (Code Table)

Key: deceptivedeceptivedeceptive
Plaintext: wearediscoveredsaveyourself
Ciphertext: ZICVTWQNGRZGVTWAVZHCQYGLMGJ

D행의 W열: Z

E행의 E열: I

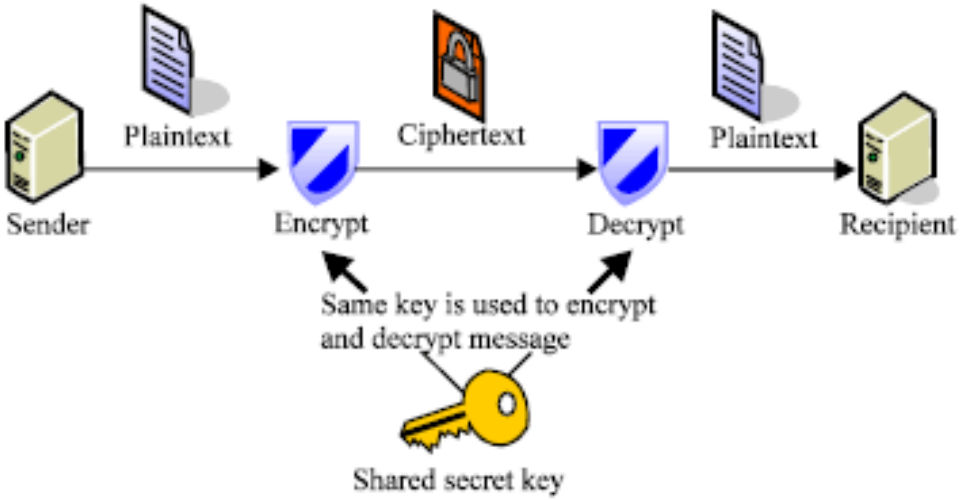
C행의 A열: C

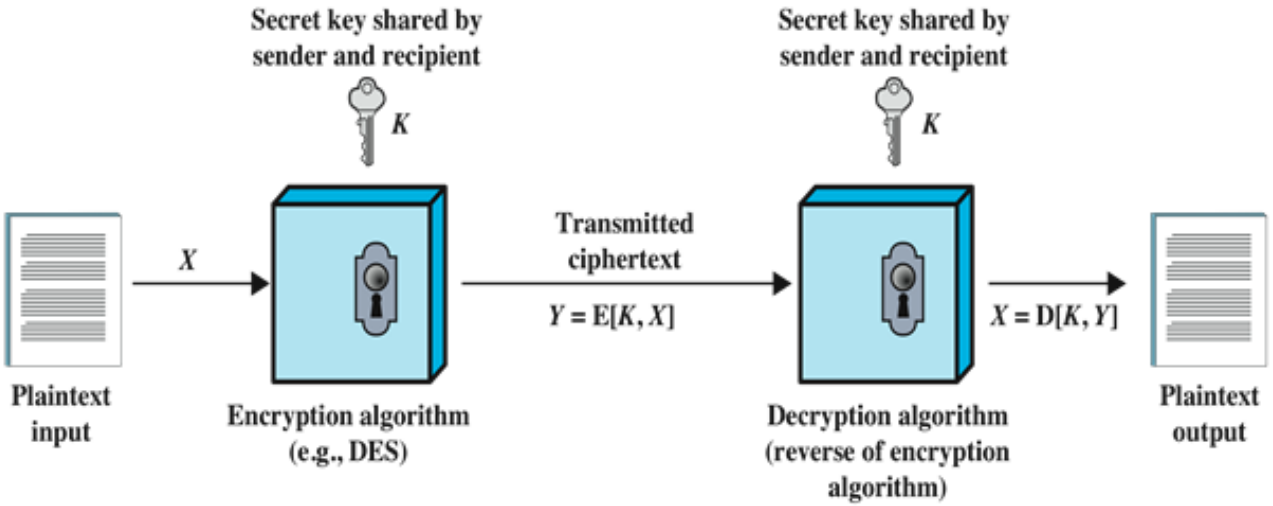


	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
A	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
B	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
C	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
D	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
E	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T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

Symmetric Cryptography

- **Symmetric encryption & decryption** (Microsoft corporation, 2005)





Comparison of Three Popular Symmetric Encryption Algorithms

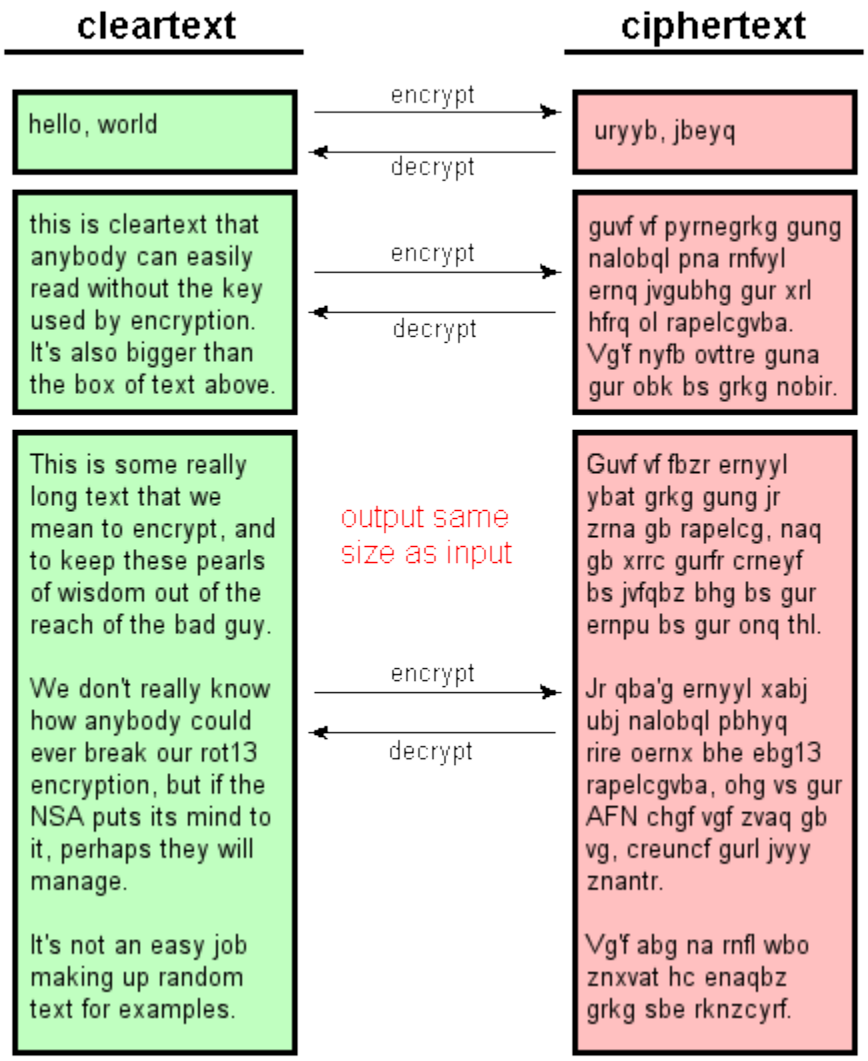
DES (Data Encryption Standard)
Triple-DES
AES (Advanced Encryption Standard)

	AES	Triple-DES
Type of algorithm	Symmetric, block cipher	Symmetric, feistel cipher
Key size (in bits)	128, 192, 256	112 or 168
Speed	High	Low
Time to crack (assume a machine could try 255 keys per second - NIST)	149 trillion years	4,6 billion years
Resource consumption	Low	Medium

	DES	AES
Date	1976	1999
Block size	64	128
Key length	56	128, 192, 256
Number of rounds	16	9,11,13
Encryption primitives	Substitution, permutation	Substitution, shift, bit mixing
Cryptographic primitives	Confusion, diffusion	Confusion, diffusion
Design	Open	Open
Design rationale	Closed	Open
Selection process	Secret	Secret, but accept open public comment
Source	IBM, enhanced by NSA	Independent cryptographers

Encryption/Decryption

- **Two-way operation**
- **Two texts should roughly correspond to each other in size**
- **A shared secret key**



Practical Application: Encryption of Stored Data

common to encrypt transmitted data

much less common for stored data

there is often little protection beyond domain authentication and operating system access controls

data are archived for indefinite periods

even though erased, until disk sectors are reused data are recoverable

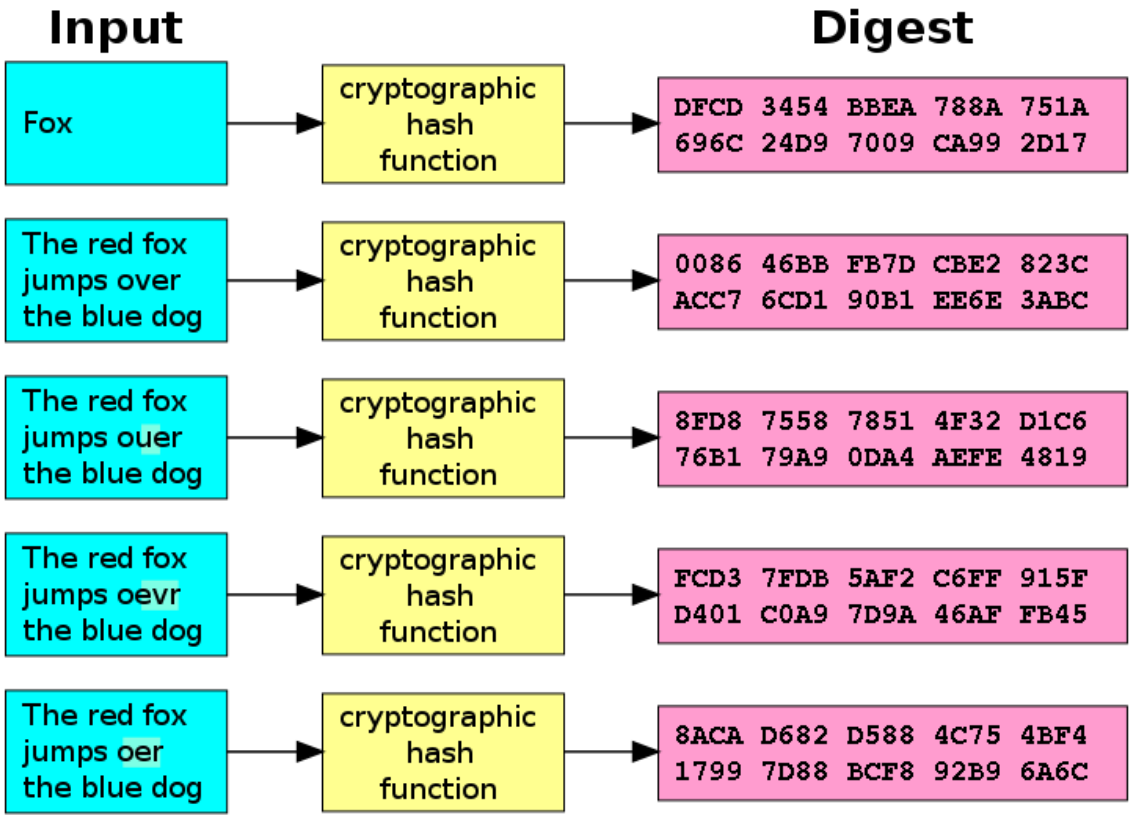
approaches to encrypt stored data:

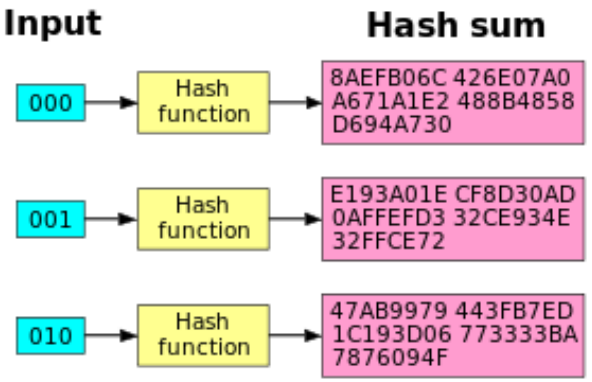
- use a commercially available encryption package
- back-end appliance
- library based tape encryption
- background laptop/PC data encryption

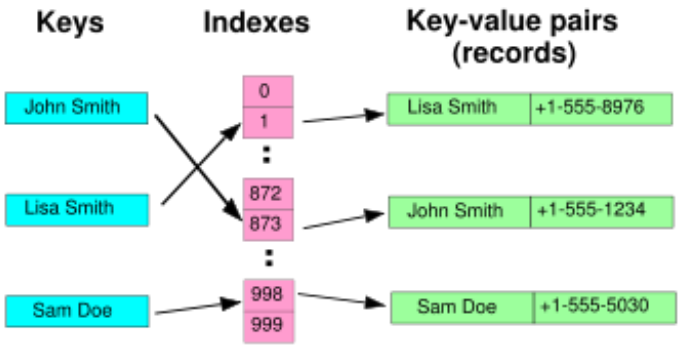
Hash Functions

- MD5 (Message Digest 5): 128 bit hash value
- SHA-1 (Secure Hash Algorithm – 1): 160 bit hash value
- SHA-256

Cryptographic Hash Functions







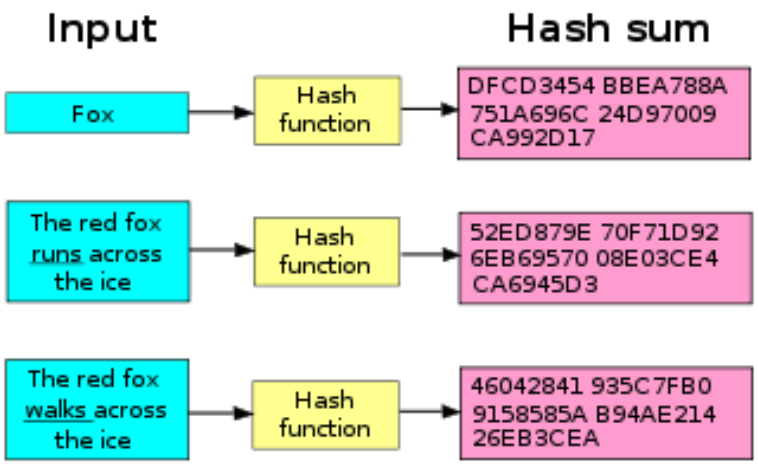
http://en.wikipedia.org/wiki/Avalanche_effect

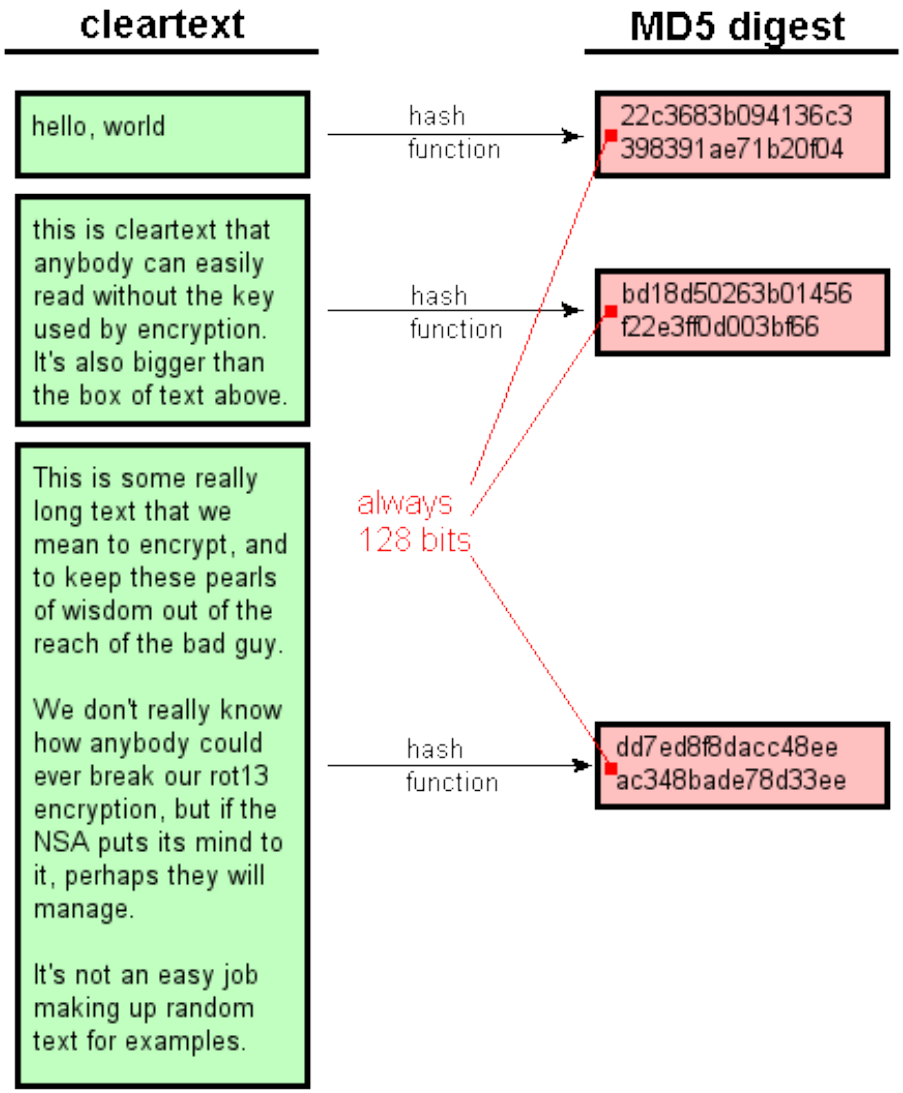
http://cse.csusb.edu/tong/courses/cs330/notes/hash.php

Cryptographic Hash Functions

- Hashes are "digests" ("checksum")
  - a fixed-length hash value
- One-way operation
- may be used with or without a key

http://www.unixwiz.net/techtips/iguide-crypto-hashes.html





Secure Hash Functions

- One-way hash function
  - Input: A variable-size message *M*
  - Output: A fixed-size digest *h*
- A hash function does not take a secret key as input
- *The message is padded out to an integer multiple* of some fixed length (e.g., 1024 bits) and the padding includes the value of the length of the original message in bits.
- The length field is a security measure to increase the difficulty for an attacker to produce an alternative message with the same hash value.

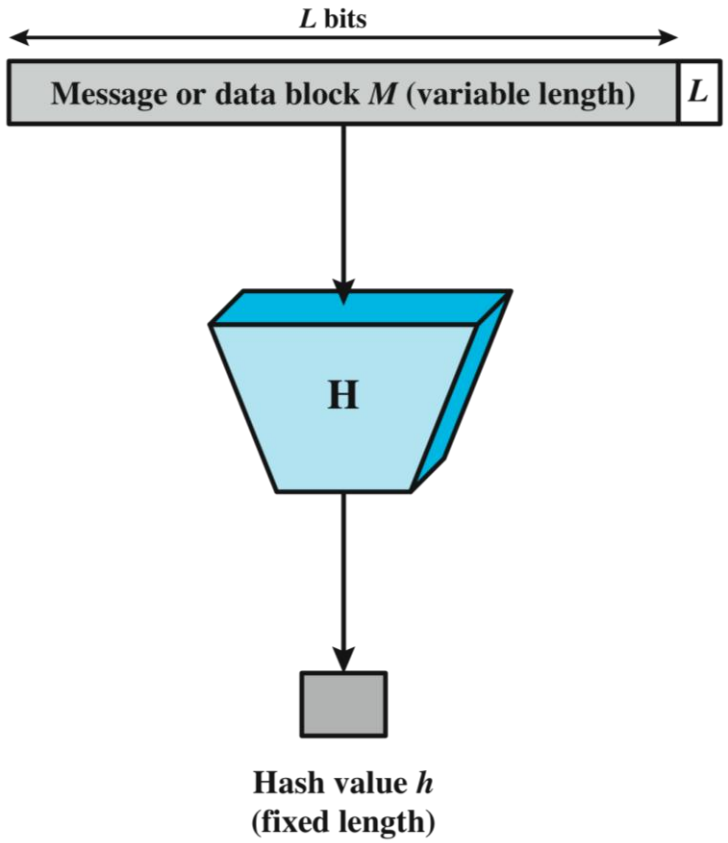
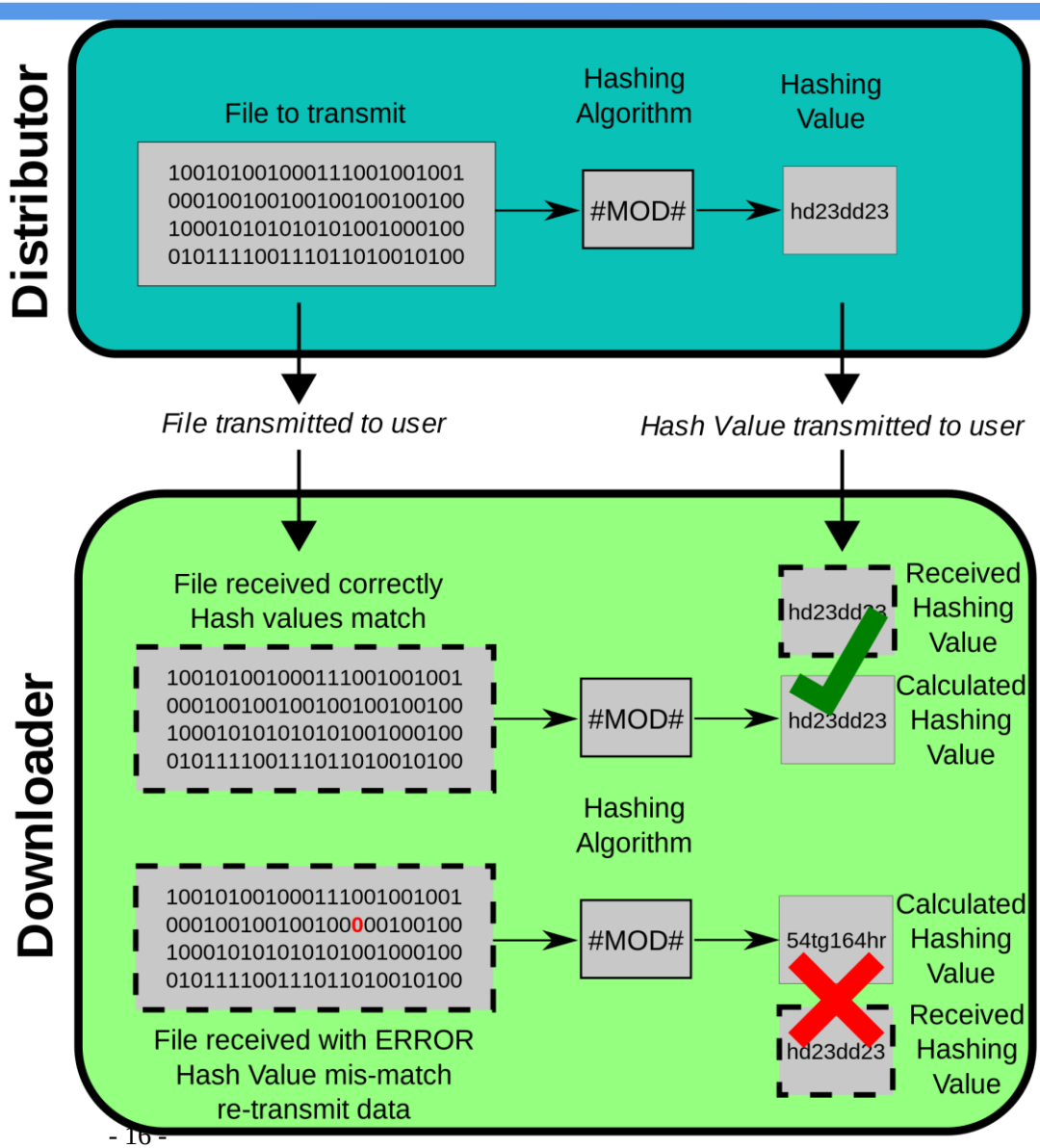
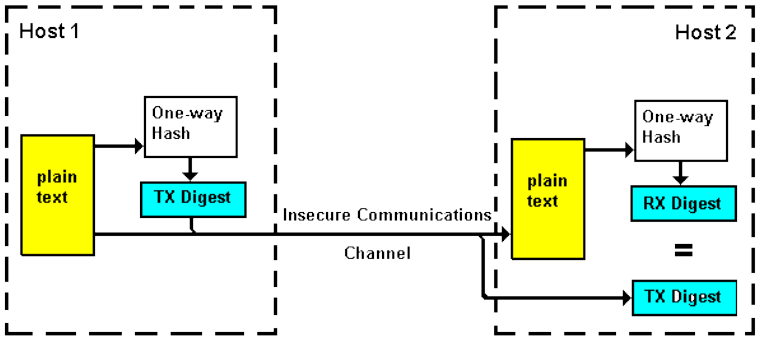


Figure 2.5 Block Diagram of Secure Hash Function; $h = H(M)$

Hash functions

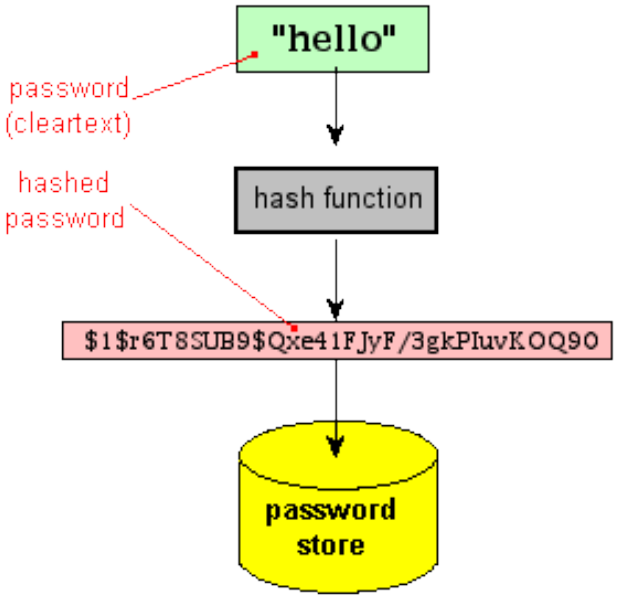
- **How are hashes used?**
  - **Verifying file integrity**
  - **Hashing passwords**
  - **Digitally signed documents**

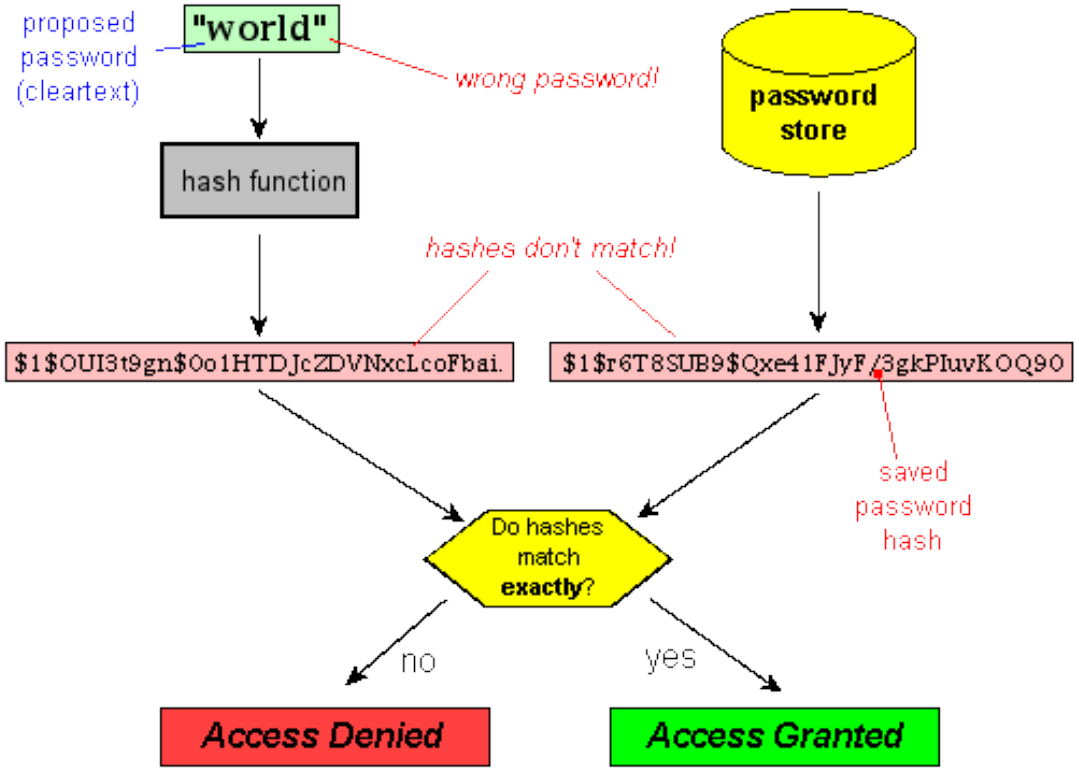




Cryptographic Hash: Hashing passwords

- Storing a hash instead of a password
- Testing a proposed password against the stored hash





source: *http://www.unixwiz.net/techtips/iguide-crypto-hashes.html*