

Cryptography

The science and study of secret writings

Cipher – Is a secret method of writing that transforms *plaintext* into *ciphertext*

The transformation is determined by a *key*

Cryptographic systems

- One key
- Two key
- Public key
- Digital signatures

Cryptography

CS177 2011

1

Cryptography

- Comes in two flavors: Symmetric and Asymmetric
- Best for protection of “online” communications
- Good for archival data
- So-so for electronic mail
- Not good for active databases

Cryptography

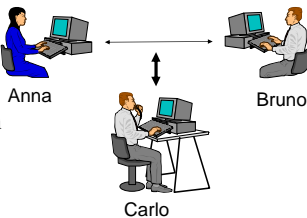
CS177 2011

2

Communication Security

Secure communication should provide:

- Privacy
- Authentication
- Integrity
- Nonrepudiation



Cryptography

CS177 2011

3

Terminology

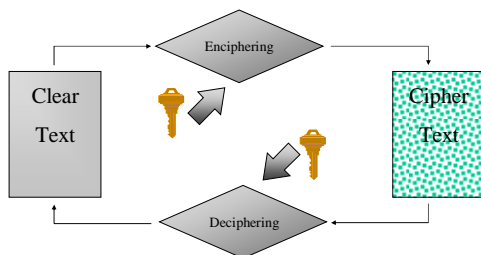
- To lock (encipher): transforms into unintelligible form based on independent data element called a *key*
- To unlock (decipher): transforms back into intelligible form, again using a *key*
- Locked data is called *ciphertext* or *black*
- Unlocked data is called *plaintext*, *cleartext*, or *red*
- Keys are themselves data and can be locked and unlocked

Cryptography

CS177 2011

4

Cryptography



Cryptography

CS177 2011

5

General Observations

- Cryptography never solves a problem; it transforms a security problem into a key management problem
- It takes a secret to keep a secret

Cryptography

CS177 2011

6

Cryptographic System (Cryptosystem)

- A plaintext message space M
- A ciphertext message space C
- A key space K
- A family of enciphering transformations

$$E_k: M \rightarrow C$$

- A family of deciphering transformations

$$D_k: C \rightarrow M$$

Crypto Systems Should Guarantee Both

- **Secrecy**
- **Authenticity**

Secrecy requirements

1. Should be computationally infeasible to systematically determine D_k from c , even if corresponding m is known
2. Should be computationally infeasible to determine m from intercepted c

Crypto Systems Should Guarantee Both

- **Secrecy**
- **Authenticity**

Authenticity requirements

1. Should be computationally infeasible to systematically determine E_k from c , even if corresponding m is known
2. Should be computationally infeasible to find c' such that $D_k(c')$ is valid plaintext in the set M

Desirable Properties of Crypto Systems

- Enciphering and deciphering must be efficient for all keys
- System must be easy to use
- The security of the system should depend on the secrecy of the keys and not on the secrecy of the algorithms E or D

Cryptanalysis

- Cryptanalysis attempts to discover the key or the plaintext of an encrypted message
 - Assume analyst knows the algorithm but not the key
- Types of attack:
 - Ciphertext only
 - Given: $C_1 = E_k(M_1), C_2 = E_k(M_2), \dots, C_i = E_k(M_i)$
 - Obtain: either M_1, M_2, \dots, M_i or k
 - Known plaintext
 - Given: $M_1, C_1 = E_k(M_1), M_2, C_2 = E_k(M_2), \dots, M_i, C_i = E_k(M_i)$
 - Obtain: either k or an algorithm to obtain M_{i+1} , from $C_{i+1} = E_k(M_{i+1})$

Cryptanalysis (continued)

- Types of attack (continued)
 - Chosen plaintext
 - Given: $M_1, C_1 = E_k(M_1), M_2, C_2 = E_k(M_2), \dots, M_i, C_i = E_k(M_i)$ where the attacker chooses M_1, M_2, \dots, M_i
 - Obtain: either k or an algorithm to obtain M_{i+1} , from $C_{i+1} = E_k(M_{i+1})$

Basis for Attacks

- Mathematical attacks
 - Based on analysis of underlying mathematics
- Statistical attacks
 - Make assumptions about the distribution of letters, pairs of letters (digrams), triplets of letters (trigrams), etc.
 - Called models of the language
 - Examine ciphertext, correlate properties with the assumptions.

Transposition Cipher

- Rearranges bits or characters in the data
- Simple transposition
 - Rail-fence cipher
 - Columnar transposition

Simple Transposition

- Ciphers simply break message into blocks and permute each block using some scheme
- Eg. Blocks of five with key (25413)
 - Consider
CMPS IS FUN FOR ALL
 - CMPS IS FU N FOR ALL
 becomes
M SCP SUFI RONF A L L

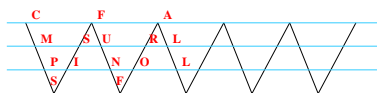
Rail Fence

- Transposition depends on a figure
- In this case the figure is a rail fence (or picket fence)



- Figure could be a scene, such as a landscape or city skyline

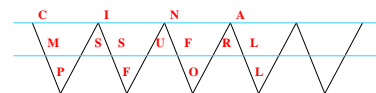
Rail Fence



If key is 2-4-3-1

MSURLSFPINOLCFA

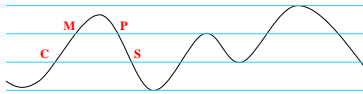
Rail Fence



If key is 1-2-3

CINAMSSUFRLPFOL

Mountain Scene



Columnar Transposition

- Uses a two dimensional array
- Text is placed in rows
- Columns are transposed
- Columns are read out as ciphered text
- Key is the transposition of the columns
 - e.g., for 4x4 matrix key could be 2-4-3-1

Columnar Transposition

Example (4x4 matrix and key = 2-4-3-1)

CMPS
ISFU
NFOR
ALLb

becomes

MSFLSURbPFOLCINA

What about (key = 1-2-3-4)?

Crypto Analysis

- Can detect transposition cipher by checking the character frequencies against the norm

a	0.080	h	0.060	n	0.070	t	0.090
b	0.015	i	0.065	o	0.080	u	0.030
c	0.030	j	0.005	p	0.020	v	0.010
d	0.040	k	0.005	q	0.002	w	0.015
e	0.130	l	0.035	r	0.065	x	0.005
f	0.020	m	0.030	s	0.060	y	0.020
g	0.015					z	0.002

Crypto Analysis

- Brute force by trying possible permutations and looking for readable text in the result
- Anagramming
 - If 1-gram frequencies match English frequencies, but other n -gram frequencies do not, probably transposition
 - Rearrange letters to form n -grams with highest frequencies

Substitution Ciphers

- Simple substitution
- Polyalphabetic
- Running key
- Vernam

Alphabet

0 – A	7 – H	14 – O	21 – V
1 – B	8 – I	15 – P	22 – W
2 – C	9 – J	16 – Q	23 – X
3 – D	10 – K	17 – R	24 – Y
4 – E	11 – L	18 – S	25 – Z
5 – F	12 – M	19 – T	
6 – G	13 – N	20 – U	

Cryptography

CS177 2011 25

Simple Substitution

- Caesar cipher is most common example of simple substitution
 - Julius used shift of 4
 - Augustus used key of 3
- (letter value + key) mod 26
- Example (key = 3)
CMPS IS FUN FOR ALL
becomes
FPSV LV IXQ IRU DOO

Cryptography

CS177 2011 26

Attacking the Cipher

- Exhaustive search
 - If the key space is small enough, try all possible keys until you find the right one
 - Caesar cipher has 26 possible keys
- Statistical analysis
 - Compare to 1-gram model of English

Cryptography

CS177 2011 27

Statistical Attack

- Compute frequency of each character in the ciphertext:
D .067 F .067 I .133 L .067
O .133 P .067 Q .067 R .067
S .067 U .067 V .133 X .067
- Apply 1-gram model of English
 - Frequency of characters (1-grams) in English is on next slide

Cryptography

CS177 2011 28

Character Frequencies

a	0.080	h	0.060	n	0.070	t	0.090
b	0.015	i	0.065	o	0.080	u	0.030
c	0.030	j	0.005	p	0.020	v	0.010
d	0.040	k	0.005	q	0.002	w	0.015
e	0.130	l	0.035	r	0.065	x	0.005
f	0.020	m	0.030	s	0.060	y	0.020
g	0.015					z	0.002

Cryptography

CS177 2011 29

Statistical Analysis

- $f(c)$ frequency of character c in ciphertext
 - $\phi(i)$ correlation of frequency of letters in ciphertext with corresponding letters in English, assuming key is i
 - $\phi(i) = \sum_{0 \leq c \leq 25} f(c)p(c-i)$
- $p(x)$ is frequency of character x in English

Cryptography

CS177 2011 30

Example Analysis from Text

- Caesar cipher
 - Plaintext is **HELLO WORLD**
 - Key is 3
 - Ciphertext is **KHOOR ZRUOG**
- Frequency of each letter in ciphertext:

G	0.1	H	0.1	K	0.1	O	0.3
R	0.2	U	0.1	Z	0.1		

Statistical Analysis

- $\varphi(i)$ correlation of frequency of letters in ciphertext with corresponding letters in English, assuming key is i
- $\varphi(i) = \sum_{0 \leq c \leq 25} f(c)p(c-i)$ so here,

$$\varphi(i) = 0.1p(6-i) + 0.1p(7-i) + 0.1p(10-i) + 0.3p(14-i) + 0.2p(17-i) + 0.1p(20-i) + 0.1p(25-i)$$
 - $f(x)$ is frequency of character c in ciphertext
 - $p(x)$ is frequency of character x in English

Correlation: $\varphi(i)$ for $0 \leq i \leq 25$

i	$\varphi(i)$	i	$\varphi(i)$	i	$\varphi(i)$	i	$\varphi(i)$
0	0.0482	7	0.0442	13	0.0520	19	0.0315
1	0.0364	8	0.0202	14	0.0535	20	0.0302
2	0.0410	9	0.0267	15	0.0226	21	0.0517
3	0.0575	10	0.0635	16	0.0322	22	0.0380
4	0.0252	11	0.0262	17	0.0392	23	0.0370
5	0.0190	12	0.0325	18	0.0299	24	0.0316
6	0.0660					25	0.0430

The Result

- Most probable keys, based on φ :
 - $i = 6$, $\varphi(i) = 0.0660$
 - plaintext **EBHIL TLOLA**
 - $i = 10$, $\varphi(i) = 0.0635$
 - plaintext **AXEEH PHKEW**
 - $i = 3$, $\varphi(i) = 0.0575$
 - plaintext **HELLO WORLD**
 - $i = 14$, $\varphi(i) = 0.0535$
 - plaintext **WTAAD LDGAS**
- Only English phrase is for $i = 3$
 - That's the key (3 or 'D')

Caesar's Problem

- Key is too short
 - Can be found by exhaustive search
 - Statistical frequencies not concealed well
 - They look too much like regular English letters
- So make it longer
 - Multiple letters in key
 - Idea is to smooth the statistical frequencies to make cryptanalysis harder

Polyalphabetic Ciphers

- Use multiple substitutions
- Most are periodic
 - These are essentially multiple Caesar ciphers
- Instead of adding the same key each time, each successive letter gets a different key added, but the keys repeat themselves
- When period is 1, this is equivalent to simple substitution

Polyalphabetic Ciphers

Example (key = SECUR)

CMPS IS FUN FOR ALL
 SECU RS ECU RSE CUR

becomes

UQRN ZK

Attacking the Cipher

- Approach
 - Establish period; call it n
 - Break message into n parts, each part being enciphered using the same key letter
 - Solve each part
 - You can leverage one part from another

Establish Period

- Kasiski: repetitions in the ciphertext occur when characters of the key appear over the same characters in the plaintext
- Example:

key VIGVIGVIGVIGVIGV
 plain THEBOYHASTHEBALL
 cipher OPKWECIYOPKWIRG

Note the key and plaintext line up over the repetitions (underlined). As distance between repetitions is 9, the period is a factor of 9 (that is, 1, 3, or 9)

Sample Cipher from Bishop

ADQYS MIUSB OXKKT MIBHK IZOOO
 EQOOG IFBAG KAUMF VVTAA CIDTW
 MOCIO EQOOG BMBFV ZGGWP CIEKQ
 HSNEW VECNE DLA AV RWKXS VNSVP
 HCEUT QOIOF MEGJS WTPCH AJMOC
 HIUIX

Repetitions in Example

Letters	Start	Repeats	Distance	Factors
MI	5	15	10	2, 5
OO	22	27	5	5
OEQOOG	24	54	30	2, 3, 5
FV	39	63	24	2, 2, 2, 3
AA	43	87	44	2, 2, 11
MOC	50	122	72	2, 2, 2, 3, 3
QO	56	105	49	7, 7
PC	69	117	48	2, 2, 2, 2, 3
NE	77	83	6	2, 3
SV	94	97	3	3
CH	118	124	6	2, 3

Estimate of Period

- OEQOOG is probably not a coincidence
 - It's too long for that
 - Period may be 1, 2, 3, 5, 6, 10, 15, or 30
- Most others (7/10) have 2 in their factors
- Almost as many (6/10) have 3 in their factors
- Begin with period of $2 \times 3 = 6$

Index of Coincidence (IC)

- Index of coincidence is probability that two randomly chosen letters from ciphertext will be the same

$$IC = [n(n-1)]^{-1} \sum_{0 \leq i \leq 25} [F_i(F_i-1)]$$

– where n is length of ciphertext and F_i the number of times character i occurs in ciphertext

Compute IC

- Tabulated for different periods:

1	0.066	3	0.047	5	0.044
2	0.052	4	0.045	10	0.041
Large			0.038		
- For sample cipher IC = 0.043
 - Indicates a key of slightly more than 5
 - A statistical measure, so it can be in error, but it agrees with the previous estimate (which was 6)

Splitting Into Alphabets

alphabet 1: AIKHOIATTOBGEEERNEOSAI
alphabet 2: DUKKEFUAWEMGKWDWSUFWJU
alphabet 3: QSTIQBMAMQBWQVLKVTMTMI
alphabet 4: YBMZOAFCOOPFHEAXPQEPOX
alphabet 5: SOIOOGVICOVCSVASHOGCC
alphabet 6: MXBOGKVDIGZINNVVCIJHH

Use same approach as for monoalphabet on each of the six alphabets

Running Key Ciphers

- Cipher has key as long as the text
- Since security of substitution cipher increases with key length, this is more secure
- Uses nonrepeating text, such as a book
 - key specified by page and paragraph number

Consider Bishop Section 8.2.2.2 (p. 107)

Example (key = The one time pad is ...)

CMPS IS FUN FOR ALL
THEO NE TIM EPA DIS

becomes

VTTG VW YCZ

Vernam Cipher

- Uses random characters as the key
- One time pads
 - Provably unbreakable
 - Why? Look at ciphertext DXQR. Equally likely to correspond to plaintext DOIT (key AJTY) and to plaintext DONT (key AJDY) and any other 4 letters
- Warning: keys *must* be random, or you can attack the cipher by trying to regenerate the key
 - Approximations, such as using pseudorandom number generators to generate keys, are *not* random

Product Ciphers

- Compose substitution and transposition ciphers
 - Lucifer
 - DES
 - AES

Cryptography

CS177 2011 49

Conventional Cryptosystems

- One key
- Encipher and decipher with same key

Asymmetric Cryptosystems

- Two keys
- Encipher and decipher with different keys
- Computationally infeasible to determine one key from the other

Cryptography

CS177 2011 50

Public-key Cryptosystems

- Each user has both a public and a private key
- Two users can communicate knowing only each other's public key
- It must be computationally infeasible to determine a user's private key from their public key

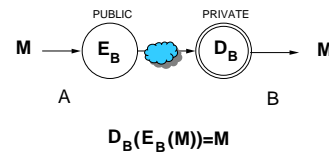
Cryptography

CS177 2011 51

Secrecy

Assume Public Key for User $K = E_K$

Assume Private Key for User $K = D_K$



Cryptography

CS177 2011 52

Digital Signature

A property private to a user that is used for signing messages

Cryptography

CS177 2011 53

Digital Signature

For A to sign a message sent to B the following properties must be satisfied by A's signature:

- B must be able to validate A's signature on the message
- It must be impossible for anyone, including B, to forge A's signature
- It must be possible for a judge or third party to settle a dispute between A and B

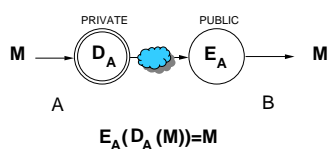
Cryptography

CS177 2011 54

Authentication

Assume Public Key for User K = Ek

Assume Private Key for User K = Dk



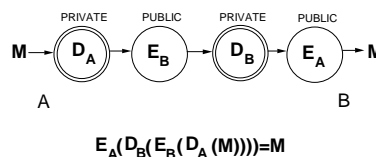
Cryptography

CS177 2011 55

Secrecy and Authentication

Assume Public Key for User K = Ek

Assume Private Key for User K = Dk



Cryptography

CS177 2011 56

Public Key Encryption

Based on problems that are known to be hard to solve

Cryptography

CS177 2011 57

Merkle-Hellman Knapsack

- Uses two knapsacks
 - Easy knapsack - superincreasing sequence
 - Hard knapsack - derived by modifying elements of the easy one
- Modification is such that any solution of one knapsack is a solution of the other

Cryptography

CS177 2011 58

Merkle-Hellman Key Selection

- Choose superincreasing sequence S of m integers
- Choose a modulus n greater than the sum of the elements of S
- Choose multiplier w that is relatively prime to n
- Construct H by replacing each integer in S by $H_i = w * S_i \text{ mod } n$

Cryptography

CS177 2011 59

- Encryption

$$C = H * M$$

- Decryption

$$\begin{aligned} w^{-1} * C &= w^{-1} * H * M \\ &= w^{-1} * w * S * M \\ &= S * M \end{aligned}$$

Cryptography

CS177 2011 60


```

10000000, 10000018, 20000020, 40000006, 80000013, 160000225, 320000443, 640000878, 1280001759,
2560003533, 5120007062, 10240014126, 20480028251, 40960056499, 819200112984, 163840225955,
327680451959, 655360903910, 13107201807638, 26214403615672, 52428807231331, 104857614442662,
209715292052329, 419430404695049, 838860913701222, 167771811402638, 335544002605289,
6713887325610536, 13421774651221080, 26843549302442161, 53687098604884319, 107374197209798651,
21474834413537297, 42949678883074394, 858993577078149178, 171798715535628355, 3435974310712596711,
6871948621425193425, 137439672486396849, 274877448070773704, 5497588971401617465,
10993117794263094811, 21987535885666180614, 43980471171212379242, 8796094235424278472,
1759218847084849519346, 351847368169690933887, 7036953383399807778, 14073707769796133554,
2814750155335799227115, 5629500310871518442219, 1125900621349396984451, 22518001242686673816889,
4503602483371476337775, 9007204970744285267654, 1801440994148803321117,
360280198899771810702335, 72057603976594362140483, 14411520753199724260855,
2882941590681748541725, 57646831612763497123537, 1152921663625299794247180,
2305843327251053958494193, 46118865621079176886701, 92237339004213353077465,
1844674681800843167079546812, 3689346323601686334159993636, 737869647263372668318187271,
147579254405745336636574536, 2951478488481549097327740601, 590296917762681346545468121,
1180591783552539629290996254, 236118356710507925386181993513, 47223671342101685072363885014,
9447429542031701447797009, 18894883684034309815540057, 3778937078612686178911880129,
755578741473626381237823760256, 15115748494250722415647520514, 3023149589401448431295041934,
6046393178029898620602032, 120925863578000779725180164073,
241781977156011850946036028142, 48357094312023119000729505285,
96714078908240462381801441312557, 1934281578172480924763602852626120,
3868563156344661848972057620028, 773126312689236990441183000484,
15474252625798473981882206100090, 3094850250799694796217646122001814,
61907100131938950439292400044, 123794210003879184810844807090,
247580420067758369741168976015378, 495176084012155116739423395020076,
9903216802410233476046704061001, 19807643604802046692926180612002,
39614085720974003391585703616246004, 7923817346184818675317467232402008,
1584563468838963735603434814464884016, 316912693767792747132686928929968036,
632823878355854642653739257695960677

```

Cryptography CS177 2011 67

Knapsack with 100 numbers, largest $\sim 10^{38}$
Sum: 63382538753555854942653739257859936077
N: 63382538753555854942653739257859936127

Cryptography CS177 2011 68

RSA Algorithm

- Uses multiplication of large primes to produce keys
- Relies on the difficulty of factoring large numbers for secrecy

Cryptography CS177 2011 69

Background

- Totient function $\phi(n)$
 - Number of positive integers less than n and relatively prime to n
 - *Relatively prime* means with no factors in common with n
- Example: $\phi(10) = 4$
 - 1, 3, 7, 9 are relatively prime to 10
- Example: $\phi(21) = 12$
 - 1, 2, 4, 5, 8, 10, 11, 13, 16, 17, 19, 20 are relatively prime to 21

Cryptography CS177 2011 70

RSA Key Selection

- Select two large primes p and q
- Compute $n = p \cdot q$
- Compute $\phi(n) = (p-1) \cdot (q-1)$
- Choose an integer e between 3 and $\phi(n)$ that has no common factor with $\phi(n)$
- Select an integer d such that $d \cdot e \pmod{\phi(n)} = 1$
- e, n are made public
- $p, q, d, \phi(n)$ are kept secret

Cryptography CS177 2011 71

- **Encryption**

$$C = M^e \pmod n$$
- **Decryption**

$$M = C^d \pmod n$$

Cryptography CS177 2011 72

Example

$p = 5$
 $q = 7$
 $n = p * q = 5 * 7 = 35$
 $\phi(n) = (p-1)(q-1) = 4 * 6 = 24$
 $e = 11$
 $d = 11$

Example

$p = 53$
 $q = 61$
 $n = p * q = 53 * 61 = 3233$
 $\phi(n) = (p-1)(q-1) = 52 * 60 = 3120$
 $e = 71$
 $d = 791$

Product Ciphers

- Compose substitution and transposition ciphers
 - Lucifer
 - DES
 - AES

Lucifer

- Developed by IBM in 1974
- S Boxes - Nonlinear Substitution Boxes
 - 4
- P Boxes - Permutation Boxes
 - 128
- 128 bit key

DES - Data Encryption Standard

- Enciphers 64-bit blocks
- Outputs 64 bits of ciphertext
- Uses 56-bit key
- Adapted by NBS(NIST) for unclassified US government applications
- Initial and final permutation
- 16 rounds (iterations)
 - S boxes and P boxes

Controversy

- Considered too weak
 - Diffie, Hellman said in a few years technology would allow DES to be broken in days
 - Design using 1999 technology published
 - Design decisions not public
 - S-boxes may have backdoors

Strength of DES

- Undesirable properties
- Special purpose machine attacks
- Double DES
- Triple DES

Undesirable Properties

- 4 weak keys
 - They are their own inverses
- 12 semi-weak keys
 - Each has another semi-weak key as inverse
- S-boxes exhibit irregular properties
 - Distribution of odd, even numbers non-random
 - Outputs of fourth box depends on input to third box

Electronic Frontier Foundation

- Built a special purpose machine
- Cost budget \$210,000
 - \$80,000 design
 - \$130,000 material
- Crack DES key in 4.5 days
- Design and algorithms published in scannable form

Double DES

- Encrypt (k1) Encrypt(k2)
- Susceptible to “meet-in-the-middle” attack
 - Plaintext attack
 - Reduces the number of keys to check from 2^{112} to 2^{57}

Triple DES

- Encrypt(k1) Decrypt(k2) Encrypt(k3)
- By using same key for all three it is identical to DES
- Having all three keys unique is referred to as “triple key triple DES”

Advanced Encryption Standard

- NIST initiated a competition for AES in 1999
- Rijndael was selected in October 2000
 - Vincent Rijmen and Joan Daemen
- Became a Federal Information Processing standard (FIPS 197) in November 2001
- NSA approved for classified information in June 2003

Rijndael

- Encrypts 128 word blocks
- Various key lengths
 - 128 uses 10 rounds
 - 192 uses 12 rounds
 - 256 uses 14 rounds
- Single S box - one byte in one byte out
- P box based on square of bytes
- 16 bytes of key per round

Cryptography

CS177 2011 85

Rijndael (continued)

- Linear transformation
- 16 bytes arranged in a 4x4 matrix (square)
 - Step 1: Row shift
 - Row 1 no shift
 - Row 2 left shift 1
 - Row 3 left shift 2
 - Row 4 left shift 3
 - Step 2: Column shifting using matrix multiplication

Cryptography

CS177 2011 86

Block Ciphers

- Break message M into successive blocks M1, M2, ...
- Encipher each Mi with the same key k

$$E_k(M) = E_k(M_1)E_k(M_2) \dots$$

Cryptography

CS177 2011 87

Block Ciphers

- Advantages
 - Only one execution of the encryption algorithm per n characters
 - Errors in one ciphertext block have no effect on other blocks
- Disadvantages
 - More susceptible to cryptanalysis
 - Identical blocks of plaintext yield identical blocks of ciphertext
 - Vulnerable to ciphertext searching
 - More susceptible to replay

Cryptography

CS177 2011 88

Block Chaining

- Inserts some bits of the previous ciphertext block into unused portions of the current plaintext block before encrypting it
- Reduces the number of available message bits per block

Cryptography

CS177 2011 89

Cipher Block Chaining

- Exclusive ORs previous ciphertext block with the current plaintext block then encrypts the result

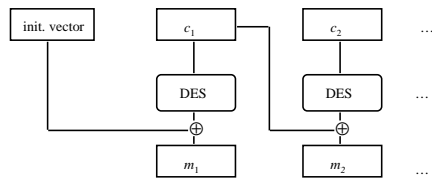
$$C_i = E_k(M_i \oplus C_{i-1})$$

- Ci is functionally dependent on all previous blocks
- Useful for checksumming and digital signatures

Cryptography

CS177 2011 90

CBC Mode Decryption



Cryptography

CS177 2011 91

Cipher Feedback

- Part of previous ciphertext is shifted into a shift register
- Shift register is encrypted with the user's key and the result is XOR'd with the plaintext block

Cryptography

CS177 2011 92

One-way Hash Function

- Takes a variable length input and produces a fixed length output
 - input is called the *preimage*
 - output is called the *hash value* or *message digest*
- Transformation is irreversible
- Called digest function, cryptographic checksum, message integrity check

Cryptography

CS177 2011 93

Where to Encrypt and Decrypt

Link encryption

End-to-End encryption

Cryptography

CS177 2011 94

Link Encryption

- Enciphers and deciphers a message M at each node between the source and destination
 - Each host need only know the keys for its immediate neighbors
 - Data is exposed at each intermediate node

Cryptography

CS177 2011 95

End-to-End Encryption

- Encipher the message at the source and decipher it at the destination
 - User needs a separate key for each correspondent
 - More susceptible to traffic flow analysis because the destination is always exposed

Cryptography

CS177 2011 96

Two Approaches Can Be Combined

Source sends a message that is the encrypted version of the original message over link encrypted communication system

Character Frequencies

a	0.080	h	0.060	n	0.070	t	0.090
b	0.015	i	0.065	o	0.080	u	0.030
c	0.030	j	0.005	p	0.020	v	0.010
d	0.040	k	0.005	q	0.002	w	0.015
e	0.130	l	0.035	r	0.065	x	0.005
f	0.020	m	0.030	s	0.060	y	0.020
g	0.015					z	0.002