Overview

- What is cryptography?
- Classic cryptosystems
  - The Caesar cipher
  - Monoalphabetic replacement cipher
  - The one-time pad
- Types of cryptosystems
  - Codes vs. ciphers
  - Symetric-key vs. assymetric-key (public key)

What is Cryptography?

- **Cryptology** is the art and science of making and breaking “secret codes”
- **Cryptography** is the making
- **Cryptanalysis** is the breaking
- Caesars cipher
  - Replace every ‘A’ in the message with a ‘D’
  - Replace every ‘B’ in the message with a ‘E’
  - Replace every ‘C’ in the message with a ‘F’, etc.

The Caesar Cipher

- Camouflage the message "ATTACK AT DAWN" by writing "DWWDFN DW GDZQ"
- "ATTACK AT DAWN" is the **plaintext**
- "DWWDFN DW GDZQ" is the **ciphertext**
- **Encryption:** plaintext ⇒ ciphertext
- **Decryption:** ciphertext ⇒ plaintext

The Key

- Assumptions
  - Algorithms are public (Kerchoff's Principle)
  - Encrypt/decrypt depends on a **key**
  - The only secret is the key
  - For Caesars cipher, key is n, since shift forward n to encrypt, shift backward n to decrypt
    - Encryption: $C_i = (P_i + n) \bmod 26$
    - Decryption: $P_i = (C_i - n) \bmod 26$

Keyspace for a Cryptosystem

- For the Caesar cipher, any value from the set {1, 2, …, 25} can be a key
- The set of usable keys is referred to as a cryptosystem's **keyspace**
- Cryptosystems with a small keyspace are vulnerable to a **brute-force search** for the key (exhaustive key search)

What is Cryptanalysis?

- **Cryptanalysis** is the science of attacking cryptosystems
  - Deduce the key and/or recover the plaintext
- Assume adversary knows the ciphertext and encryption algorithm (maybe more)

Cryptanalysis of Caesar Cipher

- Ciphertext = "GRR MGAR OY JOBOJKJ OT ZNXKK VGXZY"
- Perform decryption with each possible key:
  - Putative plaintext with key 1
    FQQ LFZQ NX INANIJI NS YMWJJ UFWYX
  - Putative plaintext with key 2
    EPP KEYP MW HMZMHIH MR XLVII TEVXW
  - Putative plaintext with key 3
    DOO JDXO LV GLYLGHG LQ WKUHH SDUWV

Cryptanalysis (continued)

- Decryption with each possible key (continued)
  - Putative plaintext with key 4
    CNN ICWN KU FKXKFGF KP VJTGG RCTVU
  - Putative plaintext with key 5
    BMM HBVM JT EJWJEFE JO UISFF QBSUT
  - Putative plaintext with key 6
    ALL GAUL IS DIVIDED IN THREE PARTS
  - And so on....
- Only one of the putative plaintexts makes sense

Monoalphabetic Replacement

- Similar to the Caesar cipher but much larger keyspace
- A key is any permutation of the 26 letters
  - Example: JQPLMZKOWHANXIEURYTGSFDVCB
- **Cipher alphabet**

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

MR Cipher - Encryption

- Plaintext (by Thomas Jefferson):
  - “I prefer freedom with danger to slavery with ease.”
- Cipher alphabet

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

- Encryption: replace each plaintext letter with the corresponding cipher letter
  - Replace every “A” in the plaintext with a “J”
  - Replace every “B” in the plaintext with a “Q”
  - Replace every “C” in the plaintext with a “P”, etc.

MR Cipher - Encryption (cont)

- Plaintext:
  - “I prefer freedom with danger to slavery with ease.”
- Cipher alphabet:

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

- Ciphertext:
  - “W uymzmy zymmlex dwgo ljikmy ge tnjfmyc dwgo mjtm.”

MR Cipher - Decryption

- Ciphertext
  - "W uymzmy zymmlex dwgo ljikmy ge tnjfmyc dwgo mjtm."
- Cipher alphabet

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

- Decryption: replace each plaintext letter with the corresponding cipher letter from the cipher alphabet
- Plaintext
  - "I prefer freedom with danger to slavery with ease."

MR Cipher - Keyspace

- Key = some permutation of the 26 letters
- 26! = 403,291,461,126,605,635,584,000,000 > 2^{88}
- Search at one trillion keys per second
  - 400 trillion seconds
  - More than 12 million years
- How to cryptanalyze this cipher?

MR Cipher - Weak Keys

- Some keys better disguise ciphertext
  - JQPLMZKOWHANXIEURYTGSFDVCB as a key gives "W uymzmy zymmlex dwgo ljikmy ge tnjfmyc dwgo mjtm."
  - ABCDEFGHIJKLMNOPQRSTUVWXYZ as a key gives "I prefer freedom with danger to slavery with ease."
  - ABCDEFGHIJKLMNOPQRSTUVWXZY as a key gives "I prefer freedom with danger to slaverz with ease."
- **Weak** keys do not disguise the ciphertext
- Weak keys not a problem if the chance of selecting one at random is small

One-Time Pads

- Provably secure encryption scheme
- Sender and receiver generate a large, truly random key letters such as
  - IPKLPSFHGQYPWKQMSVCX…
- Sender uses each key letter to encrypt one letter of plaintext
  - $C_i = (P_i + K_i) \mod 26$
- Receiver uses each key letter to decrypt one letter of ciphertext
  - $P_i = (C_i - K_i) \mod 26$

One-Time Pad - Encryption

- One time pad: IPKLPSFHGQYPWKQMSVCX…
- Plaintext: “ATTACKATDAWN”
- Ciphertext: “JJEMSDGBKRVD”

A (1) + I (9) mod 26 = J (10)

T (20) + P (16) mod 26 = J (10)

T (20) + K (11) mod 26 = E (5)

A (1) + L (12) mod 26 = M (13)

C (3) + P (16) mod 26 = S (19)

K (11) + S (19) mod 26 = D (4)

A (1) + F (6) mod 26 = G (7)

T (20) + H (8) mod 26 = B (2)

D (4) + G (7) mod 26 = K (11)

A (1) + Q (17) mod 26 = R (18)

W (23) + Y (25) mod 26 = V (22)

N (14) + P (16) mod 26 = D (4)

One-Time Pad - Decryption

- One time pad: IPKLPSFHGQYPWKQMSVCX
- Ciphertext: “JJEMSDGBKRVD”
- Plaintext:

  “ATTACKATDAWN”

  J (10) - I (9) mod 26 = A (1)

  J (10) - P (16) mod 26 = T (20)

  E (5) - K (11) mod 26 = T (20)

  .

  .

One-Time Pad - Security

- Why is one-time pad secure?
  - Attacker doesn't know any of the one-time pad
  - The pad is random so all key letters are equally likely
  - When the attacker sees ciphertext: JJEMSDGBKRVD
  - All plaintexts are equally probable

    JJEMSDGBKRVD = ATTACKATDAWN
    for key IPKLPSFHGQYP
    JJEMSDGBKRVD = ELVISISALIVE
    for key EXIDZUNAYIZY
    Etc.

One-Time Pad (cont)

- Every plaintext message is equally possible
- No way for adversary to know which is correct
- A random key sequence added to nonrandom plaintext produces a random ciphertext
- All messages of correct length are equally likely

One-Time Pads - Drawbacks

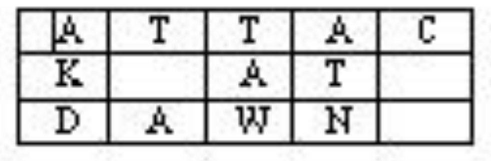
- Key must be as long as the message
- Security depends on adversary never obtaining a copy of the pad
  - Pad must be distributed securely to sender and receiver
  - Pad must be destroyed immediately after use
  - Must use the system properly
  - Pad must be random (pseudo-random not good enough)
  - Cannot reuse the pad

Types of Cryptosystems

- Codebook, cipher or a combination
- Ciphers (e.g., the Caesar cipher)
  - Transform each block of plaintext into a block of ciphertext
  - A **block** is a fixed-size unit
    - Single character (or bit)
    - Multiple characters

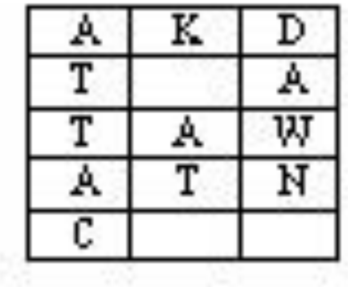
Ciphers

- **Substitution**: Apply some function to plaintext block and key to produce a block of ciphertext which replaces the plaintext (Caesar cipher)
- **Transposition**: Shuffle the blocks into a new order that depends on plaintext block key



A	T	T	A	C
K		A	T	
D	A	W	N	

= "AKDT ATAWATNC"



A	K	D
T		A
T	A	W
A	T	N
C		

= "ATTACK AT DAWN"

Codebook

- Sender and receiver each have a **codebook** that specifies one or more **codeword** for each plaintext

Word	**Codeword**
AT	September
ATTACK	March
ATTACK	December
DAWN	April
DAWN	October
(null)	July
(null)	January

Codebook Encryption/Decryption

- Plaintext:
  - "ATTACK AT DAWN"
- Ciphertext:
  - "March September October" or
  - "March September April" or
  - "July December January September April July" or …
- Codewords can be random numbers, strings of characters, or other symbols

Types of Cryptosystems

- Symmetric-key
  - Same key used for encryption and decryption
  - Typically used for bulk encryption
- Asymmetric-key (or public-key)
  - Different key used for encryption and decryption
  - Usually not used for bulk encryption

Symmetric-key Crypto

- Use of a symmetric-key cryptosystem
  - Sender and receiver agree on a secret key
    - Must be done securely
  - Messages encrypted by sender with shared key and decrypted by the receiver with same key
  - Users need to establish shared secret key beforehand

Public-Key Cryptosystems

- Standard use of a public-key cryptosystem
  - Generate a public-key/private-key pair
    - Disseminate public key, keep private key secret
  - Anyone can encrypt a message to you with your public key
  - Only you can decrypt the message using your private key
  - Users do <u>not</u> need to have a established shared secret beforehand

Public-Key Crypto (cont)

- Another use of a public-key cryptosystem
  - Digital signatures - like nondigital (and then some)
  - User encrypts a document with his private key
  - Anybody can verify the digital signature with the signer's public key
    - Only the private key can generate the signature (nonrepudiation)
  - Nothing comparable in symmetric key crypto

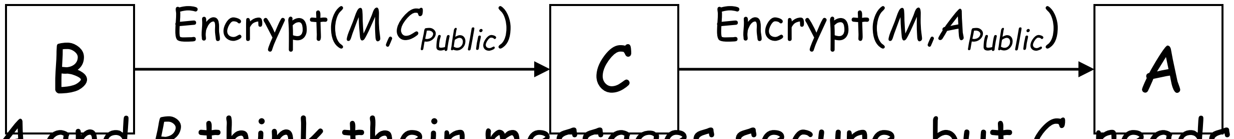
Public-Key Crypto (cont 2)

- For public-key cryptosystem to work
  - For every message, M,

    Decrypt(Encrypt(M, A_{Public}), $A_{Private}$) = M
  - For every pair of users, A and B, (A_{Public}, $A_{Private}$) and (B_{Public}, $B_{Private}$) must be distinct
  - Deriving $A_{private}$ from A_{Public} or the plaintext from the ciphertext is difficult
  - Key generation, encryption, and decryption routines must be resonably fast

Public-Key Crypto - Problems

- Problem #1 - Man in the Middle (MiM)
  - Everybody knows *A*'s public key
  - So if *B* wants to send *M* to *A*, encrypts *M* with A_{Public}
  - What if an adversary, *C*, is able to trick *B* into thinking that C_{Public} is A_{Public}?



B

Encrypt(M,C_{Public})

C

Encrypt(M,A_{Public})

A

  - *A* and *B* think their messages secure, but *C* reads them
  - Public-key cryptography depends on knowing to whom a public key belongs

Public-Key Crypto - Problems (2)

- Problem #2 - Known ciphertext (*forward search*)
  - Everybody knows *A*'s public key
  - If *C* sees Encrypt(*M*, A_{Public}) from *B* to *A*
    - *C* can choose a message, *M'*
    - Encrypt(*M'*, A_{Public})
    - Compare Encrypt(*M'*, A_{Public}) with Encrypt(*M*, A_{Public})
  - This is a serious problem if the number of possible plaintext messages is "too small"

Hybrid Cryptosystems

- Symmetric-key cryptosystems
  - Good for bulk data since fast, but require shared secrets
- Public-key cryptosystems
  - Do not require any shared secrets, but slow
- Hybrid cryptosystems
  - Given a message *M*
  - Choose a symmetric key, *K*, send K using public key crypto
  - Encrypt *M* with *K*

Summary

- **Cryptology** is the art and science of making and breaking "secret codes"
- **Cryptography** is the making
- **Cryptanalysis** is the breaking
- Classic cryptosystems include the **Caesar cipher**, **monoalphabetic replacement cipher**, **one-time pad** and many others

Summary (cont)

- **Symetric-key** cryptosystems are useful for bulk data encryption but require a shared secret
- **Public-key** cryptosystems are much slower but do not require shared secrets and support digital signatures