

CSE331: Introduction to Networks and Security

Lecture 20

Fall 2006

Announcements

- Homework 2 has been assigned:
 - ****NEW DUE DATE****
 - It's now due on *Friday, November 3rd*.

- Midterm 2 is Friday, November 10th
 - ****NEW DATE****
 - It covers just the material since Midterm 1

Broader View of Defenses

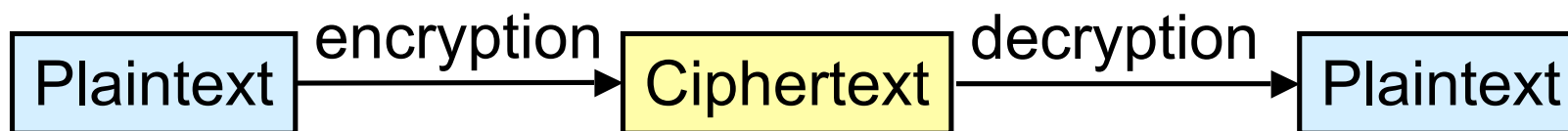
- Prevention -- *make the monoculture hardier*
 - Get the code right in the first place ...
 - ... or figure out what's wrong with it and fix it
 - Lots of active research (static & dynamic methods)
 - Security reviews now taken seriously by industry
 - E.g., ~\$200M just to *review* Windows Server 2003
 - But very expensive...and very large Installed Base problem
- Prevention -- *diversify the monoculture*
 - Via exploiting existing heterogeneity
 - Via creating artificial heterogeneity
- Prevention -- *keep vulnerabilities inaccessible*
 - Cisco's *Network Admission Control*
 - Examine hosts that try to connect, block if vulnerable
 - Microsoft's *Shield*
 - Shim-layer blocks network traffic that fits known *vulnerability* (rather than known *exploit*)



κρυπτο γραφη (Cryptography)

- Greek for “secret writing”
- Confidentiality
 - Obscure a message from eaves-droppers
- Integrity
 - Assure recipient that the message was not altered
- Authentication
 - Verify the identity of the source of a message
- Non-repudiation
 - Convince a 3rd party that what was said is accurate

Terminology



- Cryptographer
 - Invents cryptosystems
- Cryptanalyst
 - Breaks cryptosystems
- Cryptology
 - Study of crypto systems
- Cipher
 - Mechanical way of encrypting text
- Code
 - Semantic translation: “eat breakfast tomorrow” = “attack on Thursday” (or use Navajo!)



Kinds of Cryptographic Analysis

- Goal is to recover the key (& algorithm)
- Ciphertext only attacks
 - No information about content or algorithm
 - Very hard
- Known Plaintext attacks
 - Full or partial plaintext available in addition to ciphertext
- Chosen Plaintext attacks
 - Know which plaintext has been encrypted
- Algorithm & Ciphertext attacks
 - Known algorithm, known ciphertext, recover key

The Caesar Cipher

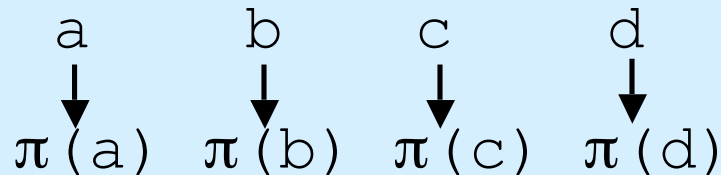
- Purportedly used by Julius Caesar (c. 75 B.C.)
 - Add 3 mod 26

a	b	c	...	x	y	z
↓	↓	↓		↓	↓	↓
d	e	f	...	a	b	c

- Advantages
 - Simple
 - Intended to be performed in the field
 - Most people couldn't read anyway
- Disadvantages
 - Violates “no security through obscurity”
 - Easy to break (why?)

Monoalphabetic Ciphers

- Also called *substitution* ciphers
- Separate *algorithm* from the *key*
 - Add $N \bmod 26$
 - rot13 = Add 13 mod 26
- General monoalphabetic cipher
 - Arbitrary permutation π of the alphabet
 - Key is the permutation



Example Cipher

	a	b	c	d	e	f	g	h	i	j	k	l	...
π	z	d	a	n	c	e	w	i	b	f	g	h	...

Plaintext: **he lied**

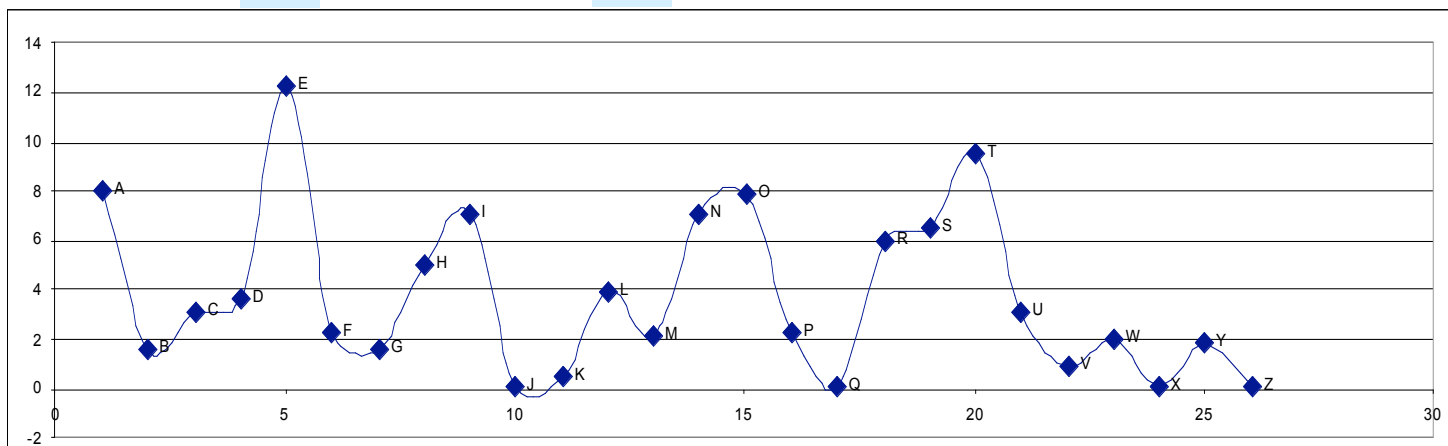
Ciphertext: **ic hbcn**

Cryptanalysis of Monoalphabetic Ciphers

- Brute force attack: try every key
 - $N!$ Possible keys for N -letter alphabet
 - $26! \approx 4 \times 10^{26}$ possible keys
 - Try 1 key per μsec ... 10 trillion years
- ...but (!) monoalphabetic ciphers are *easy* to solve
- One-to-one mapping of letters is bad
- Frequency distributions of common letters

Order & Frequency of Single Letters

E	12.31%	L	4.03%	B	1.62%
T	9.59	D	3.65	G	1.61
A	8.05	C	3.20	V	0.93
O	7.94	U	3.10	K	0.52
N	7.19	P	2.29	Q	0.20
I	7.18	F	2.28	X	0.20
S	6.59	M	2.25	J	0.10
R	6.03	W	2.03	Z	0.09
H	5.14	Y	1.88		





Monoalphabetic Cryptanalysis

- Count the occurrences of each letter in the cipher text
- Match against the statistics of English
- Most frequent letter likely to be “e”
- 2nd most frequent likely to be “t”
- etc.
- Longer ciphertext makes statistical analysis more likely to work...

Digrams and Trigrams

- Digrams in frequency order

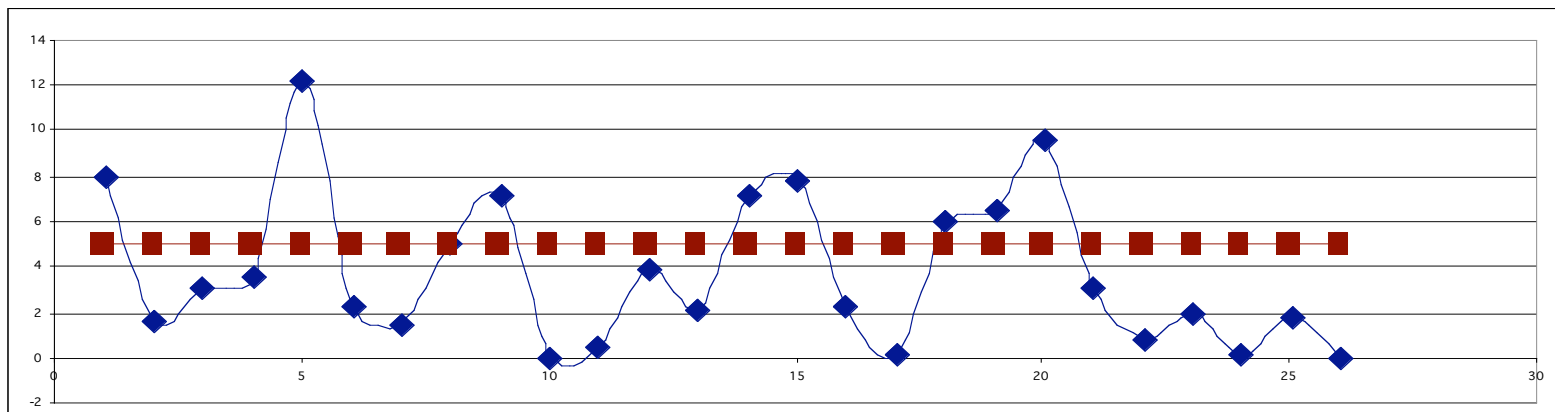
TH HE AN IN ER RE ES ON EA TI AT ST
EN
ND OR

- Trigrams in frequency order

THE AND THA ENT ION TIO FOR NDE
HAS NCE EDT TIS OFT STH MEN

Desired Statistics

- Problems with monoalphabetic ciphers
 - Frequency of letters in ciphertext reflects frequency of plaintext
- Want a single plaintext letter to map to multiple ciphertext letters
 - “e” → “x”, “c”, “w”
- Ideally, ciphertext frequencies should be flat



Polyalphabetic Substitutions

- Pick k substitution ciphers
 - $\pi_1 \pi_2 \pi_3 \dots \pi_k$
 - Encrypt the message by rotating through the k substitutions

m	e	s	s	a	g	e
$\pi_1(\mathbf{m})$	$\pi_2(\mathbf{e})$	$\pi_3(\mathbf{s})$	$\pi_4(\mathbf{s})$	$\pi_1(\mathbf{a})$	$\pi_2(\mathbf{g})$	$\pi_3(\mathbf{e})$
q	a	x	o	a	u	v

- Same letter can be mapped to multiple different ciphertexts
 - Helps smooth out the frequency distributions
 - *Diffusion*

Vigenère Tableau

- Multiple substitutions
 - Can choose “complimentary” ciphers so that the frequency distribution flattens out
 - More generally: more substitutions means flatter distribution
- Vigenère Tableau
 - Invented by Blaise de Vigenère for the court of Henry III of France (c. 1500's)
 - Collection of 26 permutations
 - Usually thought of as a 26 x 26 grid
 - Key is a word

Vigenère Tableau

	a	b	c	d	e	f	g	.	.	.
A	a	b	c	d	e	f	g	.	.	.
B	b	c	d	e	f	g	h	.	.	.
C	c	d	e	f	g	h	i	.	.	.
D	d	e	f	g	h	i	j	.	.	.
E	e	f	g	h	i	j	k	.	.	.
.
.

Plaintext: a bad deed
Key "bed": B EDB EDBE
Ciphertext: b fde hgfh

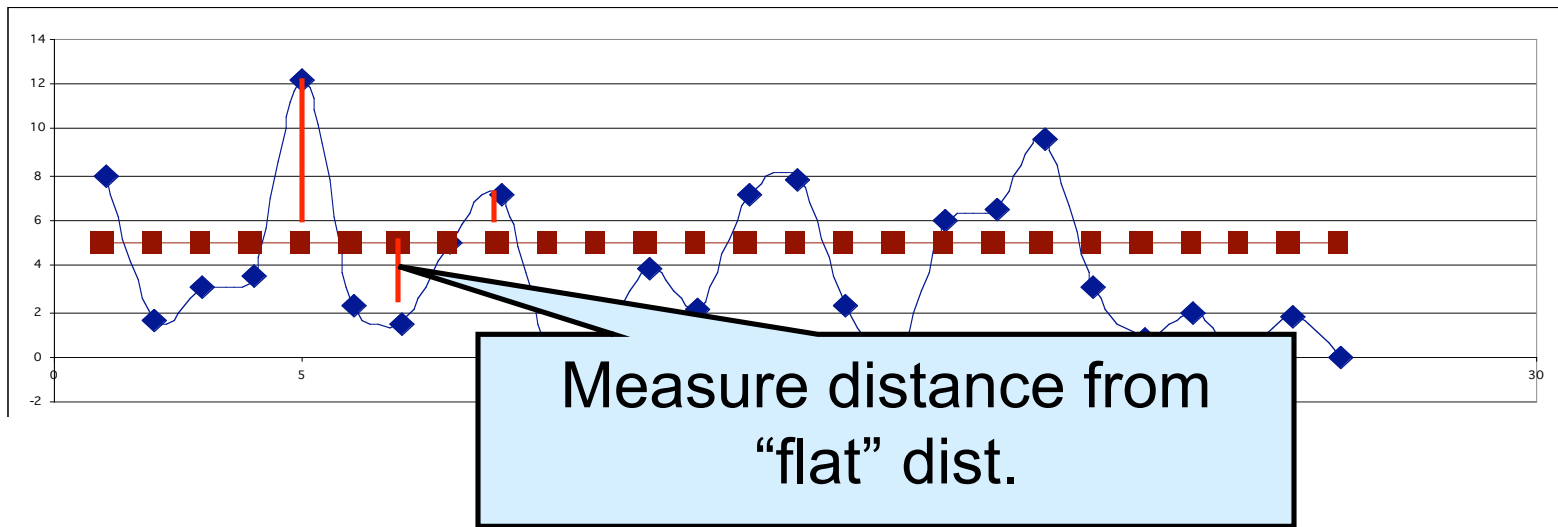
Kasiski Method

- Identify key length of polyalphabetic ciphers
 - If pattern appears k times and key length is n then it will be encoded k/n times by the same key
- 1. Identify repeated patterns of ≥ 3 chars.
- 2. For each pattern
 - Compute the differences between starting points of successive instances
 - Determine the factors of those differences
- 3. Key length is likely to be one of the frequently occurring factors

Cryptanalysis Continued

- Once key length is guessed to be k ...
- Split ciphertext into k slices
 - Single letter frequency distribution for each slice should resemble English distribution
- How do we tell whether a particular distribution is a good match for another?
 - Let $\text{prob}(\alpha)$ be the probability for letter α
 - In a perfectly flat distribution
$$\text{prob}(\alpha) = 1/26 \approx 0.0384$$

Variance: Measure of “roughness”



$$\begin{aligned}\text{Var} &= \sum_{\alpha = a}^{\alpha = z} (\text{prob}(\alpha) - 1/26)^2 \\ &= \dots \\ &= \left(\sum_{\alpha = a}^{\alpha = z} \text{prob}(\alpha)^2 \right) - 1/26\end{aligned}$$

Estimate Variance From Frequency

- $\text{prob}(\alpha)^2$ is probability that any two characters drawn from the text will be α
- Suppose there are n ciphertext letters total
- Suppose $\text{freq}(\alpha)$ is the frequency of α
- What is likelihood of picking α twice at random?
 - $\text{freq}(\alpha)$ ways of picking the first α
 - $(\text{freq}(\alpha) - 1)$ ways of picking the second α
 - But this counts twice because $(\alpha, \beta) = (\beta, \alpha)$
 - So
$$\frac{\text{freq}(\alpha) \times (\text{freq}(\alpha) - 1)}{2}$$

Index of Coincidence

- But there are $\frac{n \times (n-1)}{2}$ pairs of letters
- ...so $\text{prob}(\alpha)$ is roughly $\frac{\text{freq}(\alpha) \times (\text{freq}(\alpha) - 1)}{n \times (n-1)}$
- Index of coincidence: approximates variance from frequencies

$$\text{IC} = \sum_{\alpha = a}^z \frac{\text{freq}(\alpha) \times (\text{freq}(\alpha) - 1)}{n \times (n-1)}$$

What's it good for?

- If the distribution is flat, then $IC \approx 0.0384$
- If the distribution is like English, then $IC \approx 0.068$
- Can verify key length:

keylen	1	2	3	4	5	many
IC	0.068	0.052	0.047	0.044	0.044	... 0.038



Summary: Cracking Polyalphabetic

- Use Kasiski method to guess likely key lengths
- Compute the Index of Coincidence to verify key length k
- k -Slices should have similar IC to English
- Note: digram information harder to use for polyalphabetic ciphers...
 - May want to consider “split digrams”
 - Example: if `tion` is a common sequence $k=2$ then “`t?o`” and “`i?n`” are likely “split digrams”