

Cryptography

Lecture 2

Announcements

- HW1 due on Monday, 2/10 at beginning of class
- Discrete Math Readings/Quizzes on Canvas due on Thursday, 2/13 at 11:59pm
- Survey results: post-quantum cryptography and bitcoin/cryptocurrency

Agenda

- Last time:
 - Historical ciphers and their cryptanalysis (K/L 1.3)
- This time:
 - More cryptanalysis (K/L 1.3)
 - Discussion on defining security
 - Basic terminology
 - Formal definition of symmetric key encryption (K/L 2.1)
 - Information-theoretic security (K/L 2.1)

Shift Cipher

- For $0 \leq i \leq 25$, the i th plaintext character is shifted by some value $0 \leq k \leq 25 \pmod{26}$.
 - E.g. $k = 3$

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
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

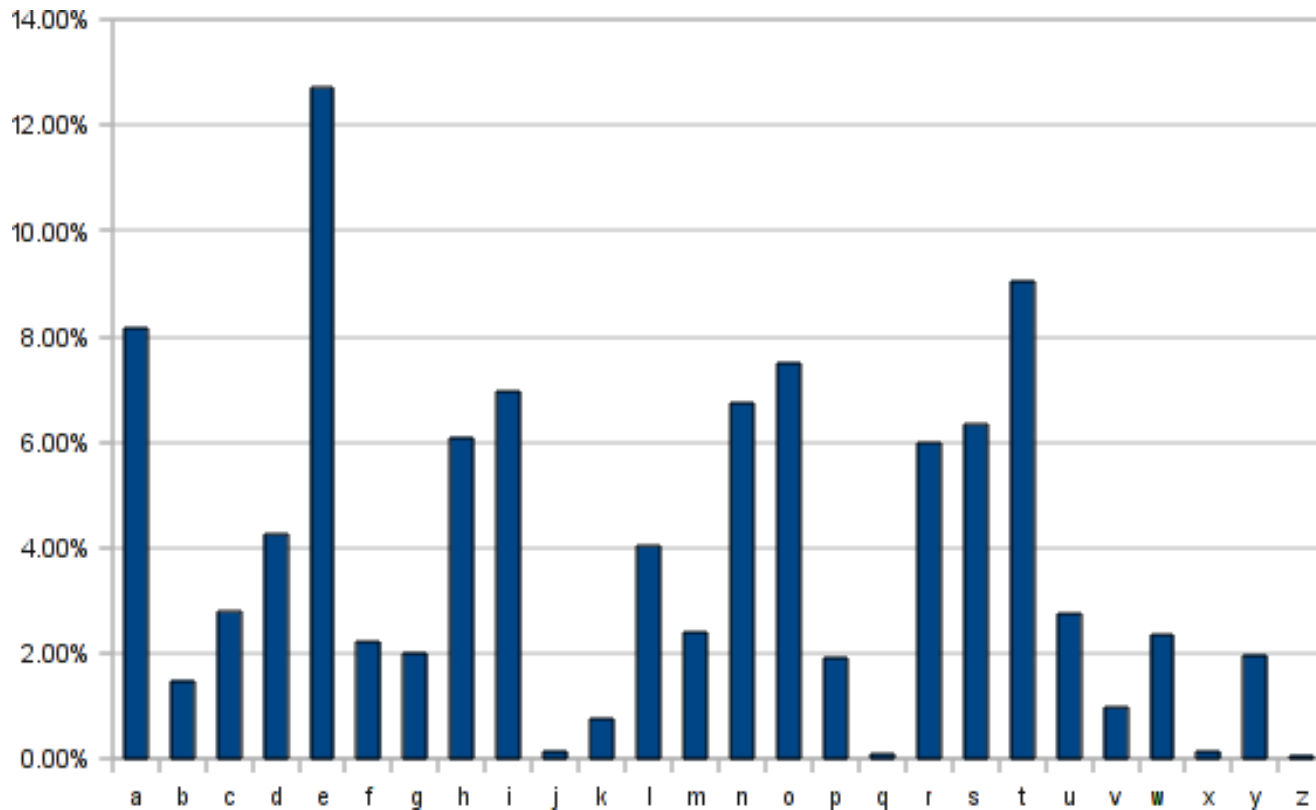
goodmorning



JRRGPRUQLQJ

Frequency Analysis

If plaintext is known to be grammatically correct English, can use frequency analysis to break monoalphabetic substitution ciphers:



An Improved Attack on Shift/Caesar Cipher using Frequency Analysis

- Associate letters of English alphabet with numbers 0...25
- Let p_i denote the probability of the i -th letter in English text.

- Using the frequency table:

$$\sum_{i=0}^{25} p_i^2 \approx 0.065$$

- Let q_i denote the probability of the i -th letter in this ciphertext: # of occurrences/length of ciphertext
- Compute $I_j = \sum_{i=0}^{25} p_i \cdot q_{i+j}$ for each possible shift value j
- Output the value k for which I_k is closest to 0.065.

Vigenere Cipher (1500 A.D.)

- Poly-alphabetic shift cipher: Maps the same plaintext character to different ciphertext characters.
- Vigenere Cipher applies multiple shift ciphers in sequence.
- Example:

Plaintext:	t	e	l	l	h	i	m	a	b	o	u	t	m	e
Key:	c	a	f	e	c	a	f	e	c	a	f	e	c	a
Ciphertext:	V	E	Q	P	J	I	R	E	D	O	Z	X	O	E

Breaking the Vigenere cipher

- Assume length of key t is known.
- Ciphertext $C = c_1, c_2, c_3, \dots$
- Consider sequences
 - $c_1, c_{1+t}, c_{1+2t}, \dots$
 - $c_2, c_{2+t}, c_{2+2t}, \dots$
 - \dots
- For each one, run the analysis from before to determine the shift k_j for each sequence j .

Index of Coincidence Method

- How to determine the key length?
- Consider the sequence: $c_1, c_{1+t}, c_{1+2t}, \dots$ where t is the true key length
- We expect $\sum_{i=0}^{25} q_i^2 \approx \sum_{i=0}^{25} p_i^2 \approx 0.065$
- To determine the key length, try different values of τ and compute $S_\tau = \sum_{i=0}^{25} q_i^2$ for subsequence $c_1, c_{1+\tau}, c_{1+2\tau}, \dots$
- When $\tau = t$, we expect S_τ to be ≈ 0.065
- When $\tau \neq t$, we expect that all characters will occur with roughly the same probability so we expect S_τ to be $\approx \frac{1}{26} \approx 0.038$.

What have we learned?

- Sufficient key space principle:
 - A secure encryption scheme must have a key space that cannot be searched exhaustively in a reasonable amount of time.
- Designing secure ciphers is a hard task!!
 - All historical ciphers can be completely broken.
- First problem: What does it mean for an encryption scheme to be secure?

Recall our setting



Sender

k

$$c \leftarrow \text{Enc}_k(m)$$



c



Receiver

k

$$m = \text{Dec}_k(c)$$

Coming up with the right definition

After seeing various encryption schemes that are clearly not secure, can we formalize what it means to for a private key encryption scheme to be secure?

Coming up with the right definition

First Attempt:

“An encryption scheme is secure if no adversary can find the secret key when given a ciphertext”

Problem: The aim of encryption is to protect the message, not the secret key.

Ex: Consider an encryption scheme that ignores the secret key and outputs the message.

Coming up with the right definition

Second Attempt:

“An encryption scheme is secure if no adversary can find the plaintext that corresponds to the ciphertext”

Problem: An encryption scheme that reveals 90% of the plaintext would still be considered secure as long as it is hard to find the remaining 10%.

Coming up with the right definition

Third Attempt:

“An encryption scheme is secure if no adversary **learns meaningful information** about the plaintext after seeing the ciphertext”

How do you formalize **learns meaningful information**?

Coming Up With The Right Definition

How do you formalize **learns** meaningful **information**?

Two ways:

- An information-theoretic approach of Shannon (next couple of lectures)
- A computational approach (the approach of modern cryptography)

New Topic: Information-Theoretic Security

Probability Background

Terminology

- Discrete Random Variable: A discrete random variable is a variable that can take on a value from a finite set of possible different values each with an associated probability.
- Example: Bag with red, blue, yellow marbles. Random variable X describes the outcome of a random draw from the bag. The value of X can be either red, blue or yellow, each with some probability.

More Terminology

- A **discrete probability distribution** assigns a probability to each possible outcomes of a discrete random variable.
 - Ex: Bag with red, blue, yellow marbles.
- An **experiment** or **trial** (see below) is any procedure that can be infinitely repeated and has a well-defined set of possible outcomes, known as the sample space.
 - Ex: Drawing a marble at random from the bag.
- An **event** is a set of outcomes of an experiment (a subset of the sample space) to which a probability is assigned
 - Ex: A red marble is drawn.
 - Ex: A red or yellow marble is drawn.

Formally Defining a Symmetric Key Encryption Scheme

Syntax

- An encryption scheme is defined by three algorithms
 - Gen, Enc, Dec
- Specification of message space \mathbf{M} with $|\mathbf{M}| > 1$.
- Key-generation algorithm Gen :
 - Probabilistic algorithm
 - Outputs a key k according to some distribution.
 - Keyspace \mathbf{K} is the set of all possible keys
- Encryption algorithm Enc :
 - Takes as input key $k \in \mathbf{K}$, message $m \in \mathbf{M}$
 - Encryption algorithm may be probabilistic
 - Outputs ciphertext $c \leftarrow Enc_k(m)$
 - Ciphertext space \mathbf{C} is the set of all possible ciphertexts
- Decryption algorithm Dec :
 - Takes as input key $k \in \mathbf{K}$, ciphertext $c \in \mathbf{C}$
 - Decryption is deterministic
 - Outputs message $m := Dec_k(c)$

Distributions over K, M, C

- Distribution over K is defined by running Gen and taking the output.
 - For $k \in K$, $\Pr[K = k]$ denotes the prob that the key output by Gen is equal to k .
- For $m \in M$, $\Pr[M = m]$ denotes the prob. That the message is equal to m .
 - Models a priori knowledge of adversary about the message.
 - E.g. Message is English text.
- Distributions over K and M are independent.
- For $c \in C$, $\Pr[C = c]$ denotes the probability that the ciphertext is c .
 - Given Enc , distribution over C is fully determined by the distributions over K and M .