Kerckhoff's principle

We made the comment We KNOW that SHIFT was used. More generally we will always use
Kerckhoff's principle:

- *The encryption scheme* is not secret
- Eve knows the encryption scheme
- The only secret is the key
- The key must be chosen at random; kept secret

Arguments For And Against Kerckhoff's Principle

Arguments For:

- Easier to keep *key* secret than *algorithm*
- Easier to change *key* than to change *algorithm*
- Standardization
  - Ease of deployment
  - Public validation
- If prove system secure then very strong proof of security since even if Eve knows scheme she can't crack.

Arguments For And Against Kerckhoff's Principle

Arguments For:

- Easier to keep *key* secret than *algorithm*
- Easier to change *key* than to change *algorithm*
- Standardization
  - Ease of deployment
  - Public validation
- If prove system secure then very strong proof of security since even if Eve knows scheme she can't crack.

Arguments Against:

- There are none.

Byte-wise Shift Cipher

- In ASCII all small letters, cap letters numbers, punctuation, mapped to 7-bit strings.
- Use XOR instead of modular addition. Fast!
- Decode and Encode are both XOR.
- Essential properties still hold

Byte-wise shift cipher

- $\mathcal{M} = \{\text{strings of bytes}\}$
- *Gen*: choose uniform byte $k \in \mathcal{K} = \{0, \ldots, 255\}$
- $Enc_k(m_1 \ldots m_t)$: output $c_1 \ldots c_t$, where $c_i := m_i \oplus k$
- $Dec_k(c_1 \ldots c_t)$: output $m_1 \ldots m_t$, where $m_i := c_i \oplus k$
- Verify that correctness holds...

Example

Key is 11001110.
Alice wants to send 00011010, 11100011, 00000000
She sends

$$00011010 \oplus 11001110, 11100011 \oplus 11001110, 00000000 \oplus 11001110$$

$$= 11010100, 00101101, 11001110$$

Example

Key is 11001110.
Alice wants to send $00011010, 11100011, 00000000$
She sends

$$00011010 \oplus 11001110, 11100011 \oplus 11001110, 00000000 \oplus 11001110$$

$$= 11010100, 00101101, 11001110$$

Question: Should it worry Alice and Bob that the key itself was transmitted? Discuss

Example

Key is 11001110.
Alice wants to send 00011010, 11100011, 00000000
She sends

$$00011010 \oplus 11001110, 11100011 \oplus 11001110, 00000000 \oplus 11001110$$

$$= 11010100, 00101101, 11001110$$

Question: Should it worry Alice and Bob that the key itself was transmitted? Discuss
No. Eve has no way of knowing that.

Is this Cipher Secure?

- No – only 256 possible keys!
- Given a ciphertext, try decrypting with every possible key
- If ciphertext is long enough, only one plaintext will look like English.
- Better than normal shift- more keys.
- Worse than normal shift- punctuation and capitol letters have ore patterns.

Sufficient key space principle

- The key space must be large enough to make exhaustive-search attacks impractical
  - How large do you think that is?

Sufficient key space principle

- The key space must be large enough to make exhaustive-search attacks impractical
  - How large do you think that is? No real answer—depends Eve's technology.
- Note: this makes some assumptions. . .
  - English-language plaintext
  - Ciphertext sufficiently long so only one valid plaintext

Is this cipher secure if we are transmitting numbers?

If Alice sends Bob a Document in English via Byte-Shift then insecure!

What if Alice sends Bob a credit card number? Discuss

Is this cipher secure if we are transmitting numbers?

If Alice sends Bob a Document in English via Byte-Shift then insecure!

What if Alice sends Bob a credit card number? Discuss
Credit Card Numbers also have patterns:

1. Visa cards always begin with 4
2. American Express always begins 34 or 37
3. Mastercard starts with 51 or 52 or 53 or 54.

Upshot: If Eve knows what kind of information is being transmitted (English, Credit Card Numbers, numbers on checks) she can use this to make any cipher with a small key space insecure.

Affine, Quadratic, Cubic, and Polynomial Ciphers

September 8, 2019

Affine Cipher

Recall: Shift cipher with shift s:

1. Encrypt via $x \to x + s \pmod{26}$.
2. Decrypt via $x \to x - s \pmod{26}$.

We replace $x + s$ with more elaborate functions

Definition: The Affine cipher with a, b:

1. Encrypt via $x \to ax + b \pmod{26}$.
2. Decrypt via $x \to a^{-1}(x - b) \pmod{26}$

Affine Cipher

Recall: Shift cipher with shift s:

1. Encrypt via $x \to x + s \pmod{26}$.
2. Decrypt via $x \to x - s \pmod{26}$.

We replace $x + s$ with more elaborate functions

Definition: The Affine cipher with a, b:

1. Encrypt via $x \to ax + b \pmod{26}$.
2. Decrypt via $x \to a^{-1}(x - b) \pmod{26}$

Does this work? Vote YES or NO or OTHER.

Affine Cipher

Recall: Shift cipher with shift s:

1. Encrypt via $x \to x + s \pmod{26}$.
2. Decrypt via $x \to x - s \pmod{26}$.

We replace $x + s$ with more elaborate functions

Definition: The Affine cipher with a, b:

1. Encrypt via $x \to ax + b \pmod{26}$.
2. Decrypt via $x \to a^{-1}(x - b) \pmod{26}$

Does this work? Vote YES or NO or OTHER. Answer: OTHER

Affine Cipher

Recall: Shift cipher with shift s:

1. Encrypt via $x \to x + s \pmod{26}$.
2. Decrypt via $x \to x - s \pmod{26}$.

We replace $x + s$ with more elaborate functions

Definition: The Affine cipher with a, b:

1. Encrypt via $x \to ax + b \pmod{26}$.
2. Decrypt via $x \to a^{-1}(x - b) \pmod{26}$

Does this work? Vote YES or NO or OTHER. Answer: OTHER
$2x + 1$ does not work: 0 and 13 both map to 1.

Affine Cipher

Recall: Shift cipher with shift s:

1. Encrypt via $x \to x + s \pmod{26}$.
2. Decrypt via $x \to x - s \pmod{26}$.

We replace $x + s$ with more elaborate functions

Definition: The Affine cipher with a, b:

1. Encrypt via $x \to ax + b \pmod{26}$.
2. Decrypt via $x \to a^{-1}(x - b) \pmod{26}$

Does this work? Vote YES or NO or OTHER. Answer: OTHER
$2x + 1$ does not work: 0 and 13 both map to 1.
Need the map to be a bijection so it will have a unique inverse.

Affine Cipher

Recall: Shift cipher with shift s:

1. Encrypt via $x \to x + s \pmod{26}$.
2. Decrypt via $x \to x - s \pmod{26}$.

We replace $x + s$ with more elaborate functions

Definition: The Affine cipher with a, b:

1. Encrypt via $x \to ax + b \pmod{26}$.
2. Decrypt via $x \to a^{-1}(x - b) \pmod{26}$

Does this work? Vote YES or NO or OTHER. Answer: OTHER
$2x + 1$ does not work: 0 and 13 both map to 1.
Need the map to be a bijection so it will have a unique inverse.

Condition on a, b so that $x \to ax + b$ is a bij: a rel prime to 26.
Condition on a, b so that a has an inv mod 26: a rel prime to 26.

Shift vs Affine

Shift: Key space is size 26

Affine: Key space is $\{1, 3, 5, 7, 9, 11, 15, 17, 19, 21, 23, 25\} \times \{0, \ldots, 25\}$ which has $12 \times 26 = 312$ elements.

In an Earlier Era Affine would be harder to crack than Shift.

Shift vs Affine

Shift: Key space is size 26

Affine: Key space is
$\{1,3,5,7,9,11,15,17,19,21,23,25\} \times \{0,\ldots,25\}$ which has
$12 \times 26 = 312$ elements.

In an Earlier Era Affine would be harder to crack than Shift.

Today They are both easy to crack.

Both Need: The Is English algorithm. Reading through 312 transcripts to see which one looks like English would take A LOT of time!

The Quadratic Cipher

Definition: The Quadratic cipher with a, b, c: Encrypt via $x \to ax^2 + bx + c \pmod{26}$.

The Quadratic Cipher

Definition: The Quadratic cipher with a, b, c: Encrypt via $x \rightarrow ax^2 + bx + c \pmod{26}$.

Does this work? Vote YES or NO.

The Quadratic Cipher

Definition: The Quadratic cipher with a, b, c: Encrypt via $x \rightarrow ax^2 + bx + c \pmod{26}$.

Does this work? Vote YES or NO. Answer: NO

The Quadratic Cipher

Definition: The Quadratic cipher with a, b, c: Encrypt via $x \rightarrow ax^2 + bx + c \pmod{26}$.

Does this work? Vote YES or NO. Answer: NO

No easy test for Invertibility (depends on def of easy).

How Easy?: Given a quadratic $f(x)$ one could compute $f(0), \ldots, f(25)$ all mod 26 and see if all are different.

- Before computers this would be tedious and much slower than finding (as for Affine) a that is rel prime to 26.
- With computers this cipher is not used since its easily cracked.
- If alphabet is size n then can determine if invertible in $O(n)$ steps. Is this good?

The Quadratic Cipher

Definition: The Quadratic cipher with a, b, c: Encrypt via $x \rightarrow ax^2 + bx + c \pmod{26}$.

Does this work? Vote YES or NO. Answer: NO

No easy test for Invertibility (depends on def of easy).

How Easy?: Given a quadratic $f(x)$ one could compute $f(0), \ldots, f(25)$ all mod 26 and see if all are different.

- Before computers this would be tedious and much slower than finding (as for Affine) a that is rel prime to 26.
- With computers this cipher is not used since its easily cracked.
- If alphabet is size n then can determine if invertible in $O(n)$ steps. Is this good? No!. Input size is n and the poly, in binary so length $O(\log n)$. Time is Exp in length of input.
- Important throughout the course: Alice and Bob need algorithms poly in length of input which is often $O(\log n)$. So $O(n)$ is to much time.

The Polynomial Cipher

Definition: Poly Cipher with poly p (coefficients in $\{0, \ldots, 25\}$).

1. Encrypt via $x \rightarrow p(x) \pmod{26}$.
2. Decrypt via $x \rightarrow p^{-1}(x) \pmod{26}$.

Given a polynomial over mod 26 (or any mod) does it have an inverse? What is the complexity of this problem?

Vote: P, NP-complete, unknown to science.

The Polynomial Cipher

Definition: Poly Cipher with poly p (coefficients in $\{0, \ldots, 25\}$).

1. Encrypt via $x \rightarrow p(x) \pmod{26}$.
2. Decrypt via $x \rightarrow p^{-1}(x) \pmod{26}$.

Given a polynomial over mod 26 (or any mod) does it have an inverse? What is the complexity of this problem?

Vote: P, NP-complete, unknown to science.

Unknown to Science but if over mod a prime then in P. But even then, algorithm is complicated, not used.

The Polynomial Cipher

Definition: Poly Cipher with poly p (coefficients in $\{0, \ldots, 25\}$).

1. Encrypt via $x \to p(x) \pmod{26}$.
2. Decrypt via $x \to p^{-1}(x) \pmod{26}$.

Given a polynomial over mod 26 (or any mod) does it have an inverse? What is the complexity of this problem?
Vote: P, NP-complete, unknown to science.
Unknown to Science but if over mod a prime then in P. But even then, algorithm is complicated, not used.

Course website, Notes, has pointer to blog of mine on this. Some of the comments have theorems and pointers to the literature.

The Polynomial Cipher

Definition: Poly Cipher with poly p (coefficients in $\{0, \ldots, 25\}$).

1. Encrypt via $x \rightarrow p(x) \pmod{26}$.
2. Decrypt via $x \rightarrow p^{-1}(x) \pmod{26}$.

Given a polynomial over mod 26 (or any mod) does it have an inverse? What is the complexity of this problem?
Vote: P, NP-complete, unknown to science.
Unknown to Science but if over mod a prime then in P. But even then, algorithm is complicated, not used.

Course website, Notes, has pointer to blog of mine on this. Some of the comments have theorems and pointers to the literature.

The first place The Polynomial Cipher appeared was

The Polynomial Cipher

Definition: Poly Cipher with poly p (coefficients in $\{0, \ldots, 25\}$).

1. Encrypt via $x \rightarrow p(x) \pmod{26}$.
2. Decrypt via $x \rightarrow p^{-1}(x) \pmod{26}$.

Given a polynomial over mod 26 (or any mod) does it have an inverse? What is the complexity of this problem?
Vote: P, NP-complete, unknown to science.
Unknown to Science but if over mod a prime then in P. But even then, algorithm is complicated, not used.

Course website, Notes, has pointer to blog of mine on this. Some of the comments have theorems and pointers to the literature.

The first place The Polynomial Cipher appeared was

my 3-week summer course on crypto for High School Students.

So, as the kids say, its not a thing.

General Substitution Cipher

Shift and Affine were good for Alice and Bob since

1. Easy to encrypt, Easy to decrypt
2. Short Key: Roughly 5 bits for Shift, 10 bits for Affine.

Definition: Gen Sub Cipher with perm f on $\{0, \ldots, 25\}$.

1. Encrypt via $x \rightarrow f(x)$.
2. Decrypt via $x \rightarrow f^{-1}(x)$

1. Key is now permutation, $\lceil \log_2(26!) \rceil = 89$ bits.
2. Encrypt and Decrypt slightly harder

The Gen Sub Cipher is Uncrackable (informally)

Theorem: The Gen Sub Cipher is Uncrackable in reasonable time (this is an informal statement).

Proof: Eve sees a text T. There are 26! possible permutations that could have been used. Eve has to look at all of them. This takes roughly 26! steps which is unreasonable.

End of Proof

So, if this cipher is uncrackable, why is it not used more? Discuss.

The Gen Sub Cipher is Uncrackable (informally)

Theorem: The Gen Sub Cipher is Uncrackable in reasonable time (this is an informal statement).
Proof: Eve sees a text T. There are 26! possible permutations that could have been used. Eve has to look at all of them. This takes roughly 26! steps which is unreasonable.
End of Proof
So, if this cipher is uncrackable, why is it not used more? Discuss.

Because I lied to you! The proof is not correct. The proof ASSUMES that Eve uses brute force. Our model of what Eve can do is too limited.

Okay, the proof is wrong, but is Gen Sub crackable?

The Gen Sub Cipher is Uncrackable (informally)

Theorem: The Gen Sub Cipher is Uncrackable in reasonable time (this is an informal statement).
Proof: Eve sees a text T. There are 26! possible permutations that could have been used. Eve has to look at all of them. This takes roughly 26! steps which is unreasonable.
End of Proof
So, if this cipher is uncrackable, why is it not used more? Discuss.

Because I lied to you! The proof is not correct. The proof ASSUMES that Eve uses brute force. Our model of what Eve can do is too limited.

Okay, the proof is wrong, but is Gen Sub crackable?

Yes: Eve can use Freq Analysis

Freq Analysis

Alice sends Bob a LONG text encrypted by Gen Sub Cipher.
Eve finds freq of letters, pairs, triples, ...

Text in English.

1. Can use known freq: *e* is most common letter, *th* is most common pair.
2. If Alice is telling Bob about Mid East Politics than may need to adjust: *q* is more common (Iraq, Qatar) and some words more common.

Silly Counter Example – Pangrams

Pangrams: Sentence where each letter occurs at least once.

Short Pangrams ruin Freq analysis. Here are some:

1. The quick brown fox jumps over the lazy dog.
2. Pack my box with five dozen liquor jugs.
3. Amazingly few discotheques provide jukeboxes.
4. Watch Jeopardy! Alex Trebek's fun TV quiz game.

Silly Counter Example – Lipograms

Lipograms: A work that omits one letter

1. Gadsby is a 50,000-word novel with no *e*.
2. Eunoia is a 5-chapter novel, indexed by vowels. Chapter A only use the vowel A, etc.
3. How I met your mother, Season 9, Episode 9: Lily and Robin challenge Barney to get a girl's phone number without using the letter *e*.

We are not going to deal with this sillyness!
We assume long normal texts!

Alternatives to Gen Sub (History)

In the Year 2018 Alice can easily generate a random permutation of $\{a, \ldots, z\}$ and send it to Bob.

In the Year 1018 Alice needs a way to encode a random-looking permutation of $\{a, \ldots, z\}$ and transmit it to Bob. So need SHORT description of random-looking perm.

1. We show two such methods.
2. Foreshadowing the need for a short description of a random-looking string of bits which we will be central later in this course.

Alternative to Gen Sub: Keyword Shift Cipher

$\Sigma = \{a, \ldots, k\}$. Key is a word and a shift s. Key: jack, 4.
Alice then does the following:

1. list out the key word and then the remaining letters:

j	a	c	k	b	d	e	f	g	h	i

2. Now do Shift 4 on this:

j	a	c	k	b	d	e	f	g	h	i
f	g	h	i	j	a	c	k	b	d	e

3. Put the table in order to get how to encode.

a	b	c	d	e	f	g	h	i	j	k
g	j	h	a	c	k	b	d	e	f	i

Point: From a (short) word and a shift you get a Random-Looking permutation of $\{a, \ldots, k\}$. We will later define Random-Looking rigorously.

How Random Does Keyword Shift Cipher Look?

a	b	c	d	e	f	g	h	i	j	k
g	j	h	a	c	k	b	d	e	f	i

Note that h, i, j maps to d, e, f. What is prob that in a random perm of $\{a, \ldots, k\}$ there will be three in a row (we don't count wrap around).

Number of perms with three consecutive: Pick spot where begins, one of 9 ways, then pick starting point one of 9 ways, then permute the remaining $11 - 3 = 8$.

$$9 \times 9 \times 8!$$

So prob is

$$\leq \frac{9 \times 9 \times 8!}{11!} = \frac{81}{9 \times 10 \times 11} \sim 0.08 \ldots$$

We will use this later.

Alternative to Gen Sub: Keyword Mixed Cipher

$\Sigma = \{a, \ldots, k\}$. Key is word w. We take w =jack.

1. Write w and then under it the rest of Σ in blocks of size $|w|$:

$$\begin{array}{|c|c|c|c|} j & a & c & k \\ b & d & e & f \\ g & h & i & \end{array}$$

2. Write down these letters column by column:

$$\begin{array}{|c|c|c|c|c|c|c|c|c|c|c} j & b & g & a & d & h & c & e & i & k & f \end{array}$$

3. Put the letters in order under it:

$$\begin{array}{|c|c|c|c|c|c|c|c|c|c|c} j & b & g & a & d & h & c & e & i & k & f \\ a & b & c & d & e & f & g & h & i & j & k \end{array}$$

4. Put table in order. This is how we encode:

$$\begin{array}{|c|c|c|c|c|c|c|c|c|c|c} a & b & c & d & e & f & g & h & i & k & f \\ d & b & g & e & h & k & c & f & i & j & k \end{array}$$

Point: From a (short) word and a shift you get a Random-Looking permutation of $\{a, \ldots, k\}$. We will later define Random-Looking rigorously.

Keyword-Shift vs Keyword-Mixed

Both Keyword-Shift, Keyword-Mixed both take a short seed and produce a Random Looking permutation. Which one is better?

We won't answer that question, but we will show how to ask it.

We will use a game!

Keyword-Shift vs Keyword-Mixed

Both Keyword-Shift, Keyword-Mixed both take a short seed and produce a Random Looking permutation. Which one is better?

We won't answer that question, but we will show how to ask it.

We will use a game!

My wife say that when mathematicians use the word game, such games are not fun.

Keyword-Shift vs Keyword-Mixed

Both Keyword-Shift, Keyword-Mixed both take a short seed and produce a Random Looking permutation. Which one is better?

We won't answer that question, but we will show how to ask it.

We will use a game!

My wife say that when mathematicians use the word game, such games are not fun.

Today's lecture will support her viewpoint.

Keyword-Shift vs Truly Random

Alice and Eve play the following game.

Game: $\Sigma = \{a, b, \ldots, z\}$. L is length of keyword, $L = 6$.

1. Alice flips a fair coin.
   1.1 If T then Alice gen a rand perm of Σ and sends to Eve.
   1.2 If H then Alice gen a rand word $w \in \Sigma^6$, a rand $s \in \{0, \ldots, 25\}$, creates a perm using Keyword-Shift with w, s, sends to Eve.
2. Eve tries to determine if perm is from H or T. If Eve is right she wins!

Alice has no strategy in this game.

Keyword-Shift vs Truly Random

Alice and Eve play the following game.

Game: $\Sigma = \{a, b, \ldots, z\}$. L is length of keyword, $L = 6$.

1. Alice flips a fair coin.
   1.1 If T then Alice gen a rand perm of Σ and sends to Eve.
   1.2 If H then Alice gen a rand word $w \in \Sigma^6$, a rand $s \in \{0, \ldots, 25\}$, creates a perm using Keyword-Shift with w, s, sends to Eve.
2. Eve tries to determine if perm is from H or T. If Eve is right she wins!

Alice has no strategy in this game.

Eve can have a strategy.

Keyword-Shift vs Truly Random

Alice and Eve play the following game.

Game: $\Sigma = \{a, b, \ldots, z\}$. L is length of keyword, $L = 6$.

1. Alice flips a fair coin.
   1.1 If T then Alice gen a rand perm of Σ and sends to Eve.
   1.2 If H then Alice gen a rand word $w \in \Sigma^6$, a rand $s \in \{0, \ldots, 25\}$, creates a perm using Keyword-Shift with w, s, sends to Eve.
2. Eve tries to determine if perm is from H or T. If Eve is right she wins!

Alice has no strategy in this game.

Eve can have a strategy.

We measure how good the Keyword-Shift is by the probability that an optimal Eve can win.

Keyword-Shift vs Truly Random

Alice and Eve play the following game.
Game: $\Sigma = \{a, b, \ldots, z\}$. L is length of keyword, $L = 6$.

1. Alice flips a fair coin.
   1.1 If T then Alice gen a rand perm of Σ and sends to Eve.
   1.2 If H then Alice gen a rand word $w \in \Sigma^6$, a rand $s \in \{0, \ldots, 25\}$, creates a perm using Keyword-Shift with w, s, sends to Eve.
2. Eve tries to determine if perm is from H or T. If Eve is right she wins!

Alice has no strategy in this game.
Eve can have a strategy.
We measure how good the Keyword-Shift is by the probability that an optimal Eve can win.
How well can Eve do? Discuss

Keyword-Shift vs Truly Random

Alice and Eve play the following game.

Game: $\Sigma = \{a, b, \ldots, z\}$. L is length of keyword, $L = 6$.

1. Alice flips a fair coin.
   1.1 If T then Alice gen a rand perm of Σ and sends to Eve.
   1.2 If H then Alice gen a rand word $w \in \Sigma^6$, a rand $s \in \{0, \ldots, 25\}$, creates a perm using Keyword-Shift with w, s, sends to Eve.
2. Eve tries to determine if perm is from H or T. If Eve is right she wins!

Alice has no strategy in this game.

Eve can have a strategy.

We measure how good the Keyword-Shift is by the probability that an optimal Eve can win.

How well can Eve do? Discuss

As stated Eve can do very well; however, we need to adjust game.

An Issue with the Game

We have not specified how powerful Eve is.

1. If Eve is all powerful then she can list out in her head ALL perms that come from keyword-shift (of length L). Strategy: If its one of them guess that its H (she'll be right most of the time), if not then guess that its T (she will always be right).
2. If Eve is time limited then how well can show do? We won't define limited rigorously here. We note that Eve could see if there are three consecutive letters (e.g., a, b, c or p, q, r or $\ldots$) and if there are then guess H, if not then T. From prior calculation on smaller example we see that this is pretty good.

We have not specified what probability will be *pretty good*. For now do not need to. The important thing is if its smaller than prob for keyword-mixed cipher.

We leave this to the reader.

Definition of Random Looking

We do this informally.
Let C be a crypto-system. Let $|C|$ be the number of perms. (For Shift $|C| = 26$, for keyword-shift with 6-letter words, $|C| = 26^6 \times 26$).

Assume Eve is limited in time by $\log |C|$. (The idea is that Eve REALLY cannot look at anything close to $|C|$ perms.)

C generates perms that look random if when Eve plays the game the prob that she wins is $\leq \frac{1}{2}$.

Note: This is not real. $\log |C|$ is too small. The idea is that an Eve who cannot look at anything close to the all the perms in C can't do well in the game.

Why this is all Silly and Why this is Not all Silly

1. Silly: We can measure how good a cipher C is much more easily by looking at how many different permuations it can generate. For example, Shift leads to 26 perms, Affine to 312 perms.
2. Not Silly: We have restated Keyword-shift and Keyword-mixed as ways to take a short seed and get a Random Looking permutation. This is a small (though silly) example of a psuedo-random generator. We will visit that concept later and use a similar game.

The Vigenère cipher

Key: A word or phrase. Example: *dog* = *(3,14,6)*.
Easy to remember and transmit.
Example using *dog*.
Shift 1st letter by 3
Shift 2nd letter by 14
Shift 3nd letter by 6
Shift 4th letter by 3
Shift 5th letter by 14
Shift 6th letter by 6, etc.

Jacob Prinz is a Physics Major
Jacob Prinz isaPh ysics Major

encrypts to

MOIRP VUWTC WYDDN BGOFG SDXUU

The Vigenère cipher

Key: $k = (k_1, k_2, \ldots, k_n)$.
Encrypt (all arithmetic is mod 26)

$$Enc(m_1, m_2, \ldots, m_N) =$$

$$m_1 + k_1, m_2 + k_2, \ldots, m_n + k_n,$$

$$m_{n+1} + k_1, m_{n+2} + k_2, \ldots, m_{n+n} + k_n,$$

$$\ldots$$

Decrypt Decryption just reverse the process

The Vigenère cipher

- Size of key space?
  - If keys are 14-char then key space size $26^{14} \approx 2^{66}$
  - If variable length keys, even more.
  - Brute-force search infeasible
- Is the Vigenère cipher secure?
- Believed secure for many years. . .
- Might not have even been secure then. . .

Cracking Vig cipher: Step One-find Keylength

Assume T is a text encoded by Vig, key length L unknown.
For $0 \leq i \leq L-1$, letters in pos $\equiv i \pmod{26}$ – same shift.
Look for a sequence of (say) 3-letters to appear (say) 4 times.

Example: aiq appears in the
57-58-59th slot, 87-88-89th slot 102-103-104th slot
162-163-164th slot

Important: Very likely that aiq encrypted the same 3-letter sequence and hence the length of the key is a divisor of
87-57=30 102-87=15 162-102=60
The only possible L's are 1,3,5,15.

Good Enough: We got the key length down to a small finite set.

Important Point about letter Freq

Assume (and its roughly true): In an English text of length N:

e occurs $\sim 13\%$ t occurs $\sim 9\%$ a occurs $\sim 8\%$

Etc- other letters have frequencies that are true for all texts.

Important Point about letter Freq

Assume (and its roughly true): In an English text of length N:

e occurs $\sim 13\%$ t occurs $\sim 9\%$ a occurs $\sim 8\%$

Etc- other letters have frequencies that are true for all texts.

Assume (and its roughly true): In an English text of length N, if $i \ll N$, then if you take every ith letter of T:

e occurs $\sim 13\%$ t occurs $\sim 9\%$ a occurs $\sim 8\%$

Etc- other letters same frequencies as normal texts.

Important Point about letter Freq

Assume (and its roughly true): In an English text of length N:

e occurs $\sim 13\%$ t occurs $\sim 9\%$ a occurs $\sim 8\%$

Etc- other letters have frequencies that are true for all texts.

Assume (and its roughly true): In an English text of length N, if $i \ll N$, then if you take every ith letter of T:

e occurs $\sim 13\%$ t occurs $\sim 9\%$ a occurs $\sim 8\%$

Etc- other letters same frequencies as normal texts.

Relevant to us:

$\vec{q}$ freq of every Lth letter: then $\sum_{i=1}^{26} q_i^2 \approx 0.065$.

$\vec{q}$ is NOT (we won't define that rigorously): $\sum_{i=1}^{26} q_i^2$ MUCH lower.

Cracking Vig cipher: Step One-find Keylength

Let K be the set of possible key lengths. K is small. For every $L \in K$:

- Form a stream of every Lth character.
- Find the frequencies of that stream: $\vec{q}$.
- Compute $Q = \sum q_i^2$
- If $Q \approx 0.065$ then YES L is key length.
- If Q much less than 0.065 then NO L is not key length.
- One of these two will happen
- Just to make sure, check another stream.

Note: Differs from Is English:

Is English wanted to know if the text was actually English

What we do above is see if the text has same dist of English, but okay if diff letters. E.g., if z is 13%, a is 9%, and other letters have roughly same numbers as English then we know the stream is SOME Shift. We later use Is English to see which shift.

A Note on Finding Keylength

We presented Method ONE:

1. Find phrase of length x appearing y times. Differences D.
2. K is set of divisors of all $L \in D$. Correct keylength in K.
3. Test $L \in K$ for key length until find one that works.

Or could try all key lengths up to a certain length, Method TWO:

1. Let $K = \{1, \ldots, 100\}$ (I am assuming key length ≤ 100).
2. Test $L \in K$ for key length until find one that works.

Note: With modern computers use Method TWO. In days of old eyeballing it made Method ONE reasonable.

Cracking the Vig cipher: Step Two-Freq Anal

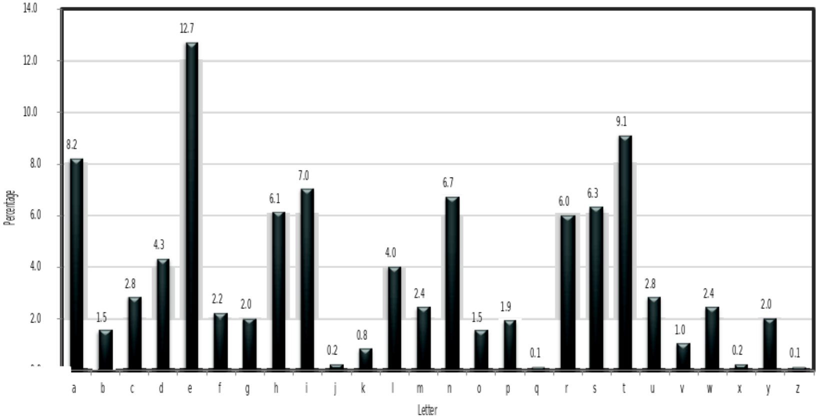
After Step One we have the key length L. Note:

- Every L^{th} character is "encrypted" using the same shift.
- Important: Letter Freq still hold if you look at every Lth letter!

Step Two:

1. Separate text T into L streams depending on position mod L
2. For each steam try every shift and use Is English to determine which shift is correct.
3. You now know all shifts for all positions. Decrypt!

Using plaintext letter frequencies



Byte-wise Vigenère cipher

- The key is a string of bytes
- The plaintext is a string of bytes
- To encrypt, XOR each character in the plaintext with the next character of the key
  - Wrap around in the key as needed
- Decryption just reverses the process.

Note: Decryption and Encryption both use XOR with same key.

Note: Can be cracked as original Vig can be cracked.

REVIEW: Negatives and Inverses Mod n

Say we are in mod 100. Our universe is $\{0, \ldots, 99\}$.

REVIEW: Negatives and Inverses Mod n

Say we are in mod 100. Our universe is $\{0, \ldots, 99\}$.

-12? the number that when you add to 12 you get 0.

Try all $x \in \{0, \ldots, 99\}$ and hope one works? Better way?

REVIEW: Negatives and Inverses Mod n

Say we are in mod 100. Our universe is $\{0, \ldots, 99\}$.

-12? the number that when you add to 12 you get 0.

Try all $x \in \{0, \ldots, 99\}$ and hope one works? Better way?

YES: $100 - 12 = 88$ works: $(100 - 12) + 12 = 100 \equiv 0$.

REVIEW: Negatives and Inverses Mod n

Say we are in mod 100. Our universe is $\{0, \ldots, 99\}$.

-12? the number that when you add to 12 you get 0.

Try all $x \in \{0, \ldots, 99\}$ and hope one works? Better way?

YES: $100 - 12 = 88$ works: $(100 - 12) + 12 = 100 \equiv 0$.

More general: $-a \pmod{n}$ is $n - a$.

REVIEW: Negatives and Inverses Mod n

Say we are in mod 100. Our universe is $\{0, \ldots, 99\}$.

-12? the number that when you add to 12 you get 0.

Try all $x \in \{0, \ldots, 99\}$ and hope one works? Better way?

YES: $100 - 12 = 88$ works: $(100 - 12) + 12 = 100 \equiv 0$.

More general: $-a \pmod{n}$ is $n - a$.

$\frac{1}{3}$? $\frac{1}{3}$ the number that when you mult by 3 you get 1.

Try all $x \in \{0, \ldots, 99\}$ and hope one works? Better way?

REVIEW: Negatives and Inverses Mod n

Say we are in mod 100. Our universe is $\{0, \ldots, 99\}$.

-12? the number that when you add to 12 you get 0.

Try all $x \in \{0, \ldots, 99\}$ and hope one works? Better way?

YES: $100 - 12 = 88$ works: $(100 - 12) + 12 = 100 \equiv 0$.

More general: $-a \pmod{n}$ is $n - a$.

$\frac{1}{3}$? $\frac{1}{3}$ the number that when you mult by 3 you get 1.

Try all $x \in \{0, \ldots, 99\}$ and hope one works? Better way?

YES: Later. Now want $3x \in \{101, 201, 301, \ldots\}$. Note 3 div 201
$3x = 201$, so $x = 67$.

REVIEW: Negatives and Inverses Mod n

Say we are in mod 100. Our universe is $\{0, \ldots, 99\}$.

-12? the number that when you add to 12 you get 0.

Try all $x \in \{0, \ldots, 99\}$ and hope one works? Better way?

YES: $100 - 12 = 88$ works: $(100 - 12) + 12 = 100 \equiv 0$.

More general: $-a \pmod{n}$ is $n - a$.

$\frac{1}{3}$? $\frac{1}{3}$ the number that when you mult by 3 you get 1.

Try all $x \in \{0, \ldots, 99\}$ and hope one works? Better way?

YES: Later. Now want $3x \in \{101, 201, 301, \ldots\}$. Note 3 div 201
$3x = 201$, so $x = 67$.
Note: $-12, \frac{1}{3}$ are intermediaries. Want result in $\{0, \ldots, n - 1\}$.