An Invitation to Cryptography:
Classical Ciphers

CMSC 28400, Autumn 2021, Lecture 1

David Cash

University of Chicago

Slides only rarely after today!

I think Cryptography is

Fun and Interesting

Beautiful and unique math
Clever attacks
Philosophy, precisely
The drama!

&

Important

For your daily life
For businesses
For liberty and democracy
Life or death for some

This class: CSMC 28400 “Cryptography”…

… Counts for the theory sequence (BS/BA in CS).
You will work with definitions, theorems, and proofs.

… Is a Computer Science class.
You will write programs building and breaking crypto.

Will assume knowledge in…

Math

Algorithms analysis (Big-Oh)
Discrete probability (a little)
Modular arithmetic

Programming

Write short programs
Understand binary/hex
Learn some python

Not assumed: Computer security
Any crypto knowledge

Outline of Topics

Part 1: Classical Crypto (Week 1)

1. Classical ciphers and how to break them
2. Enigma and the Polish Attack

Part 2: Probability Theory Background (Week 2)

1. Discrete probability spaces and events
2. Random variables

Part 3: Modern Symmetric Crypto (Weeks 3-6)

1. One-time pad and perfect secrecy
2. Blockciphers: DES and/or AES
3. Modes of operation
4. Message authentication
5. Hash functions

Part 4: Public-Key Crypto (Weeks 7-9)

1. Number theory refresh
2. Group theory
3. Discrete logarithms, factoring, RSA problems
4. Public-key encryption
5. Digital signatures

Themes:

1. Attacks!
2. Math AND CS
3. Definitions
4. Proofs

After finishing this class, you should able to…

… Understand design rationale for lots of modern crypto.

… Evaluate the security of many crypto constructions.

… Implement attacks against crypto.

This class won't cover everything, including…

… Lots of relevant crypto. 284 is just a start.

… How to securely implement crypto (!).

… How to design a secure system (website, app, …)

Assessment

1. Problem Sets (1 per week, due Thursdays)
2. Programing Projects (3, spaced during quarter)
3. One midterm exam: Week 6
4. Final exam at end of term

No participation / attendance grade

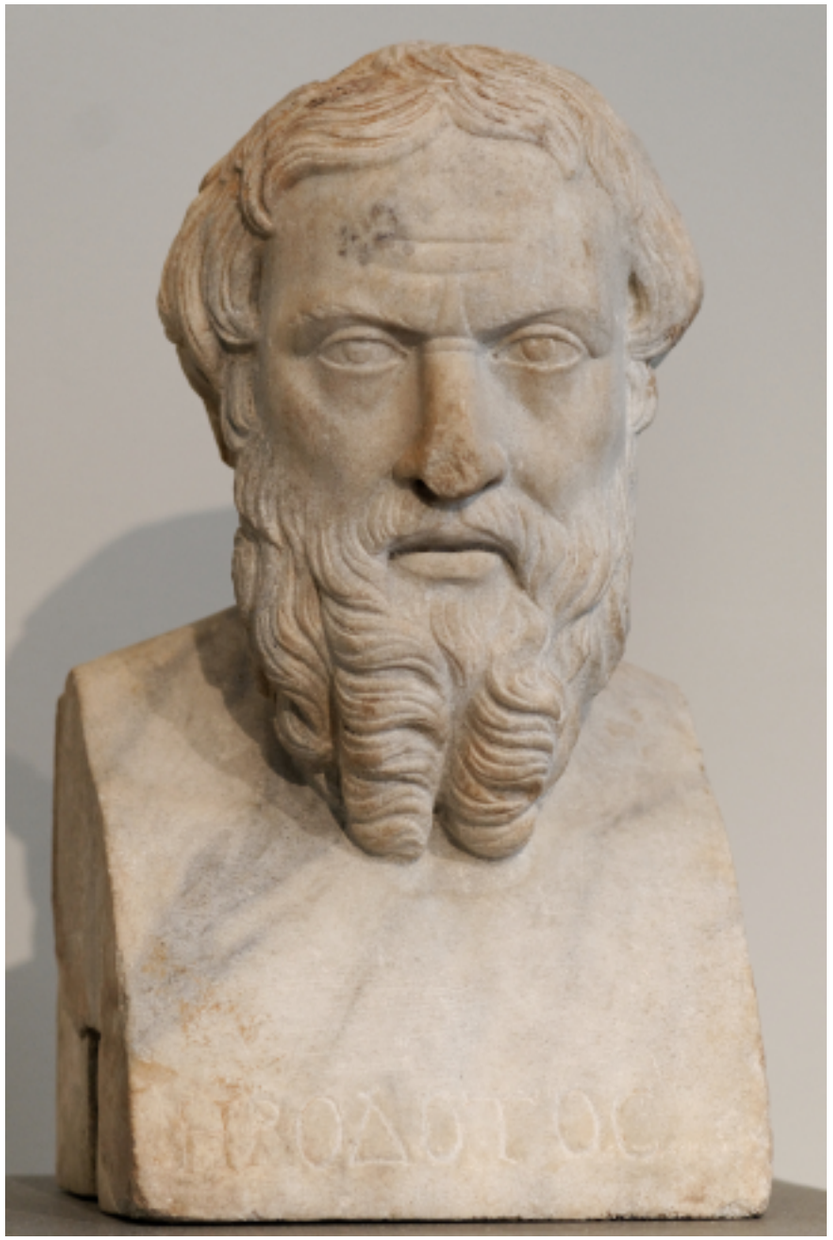
First problem set and project out this week.

Please read the syllabus (including grading policy) at:
https://www.cs.uchicago.edu/~davidcash/284-autumn-21/

This Lecture: History and Classical Ciphers

1. A very quick history of cryptography
2. Classical ciphers and how they are broken
   - Shift cipher
   - Substitution cipher
   - Vigenère cipher
   - Homophonic cipher
   - One-time pad

Grecco-Persian Wars
c. 500 BC



Herodotus (499-449BC)

Goal: Private Communication

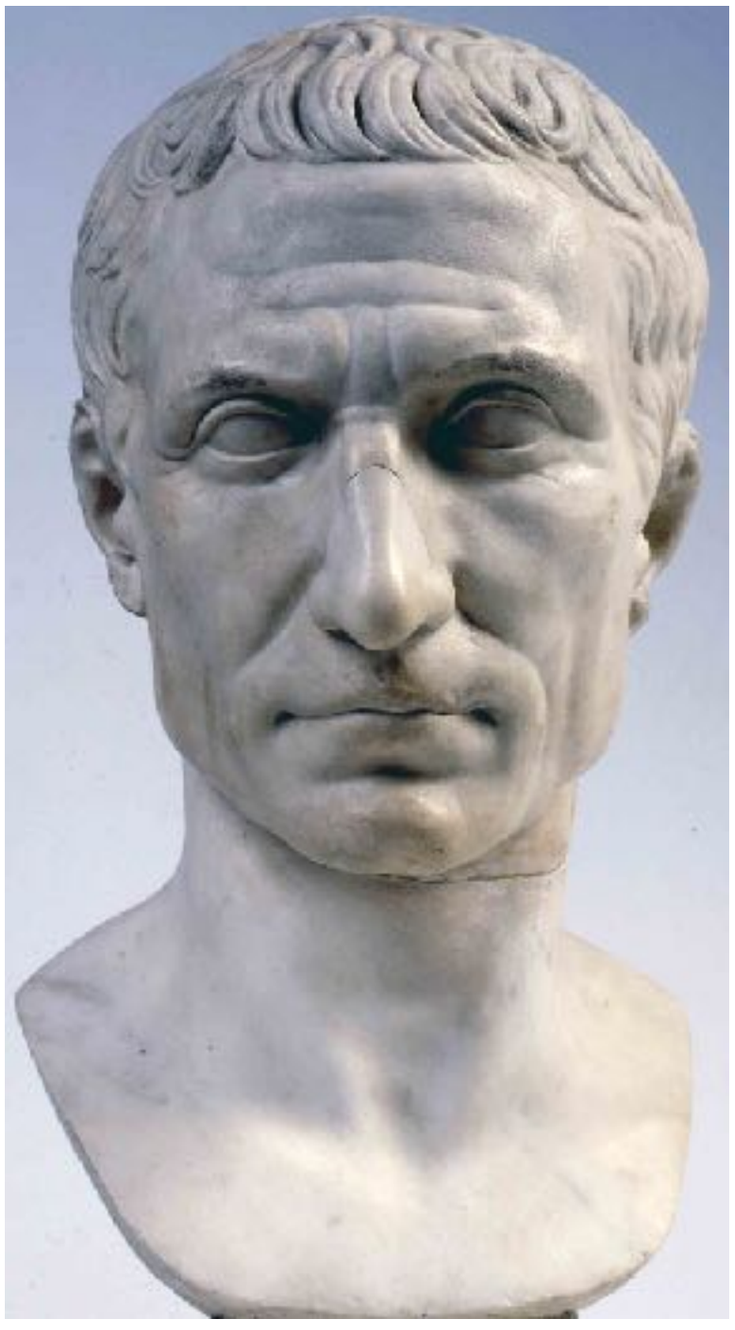
HELP IS COMING

Communication channel
(insecure)

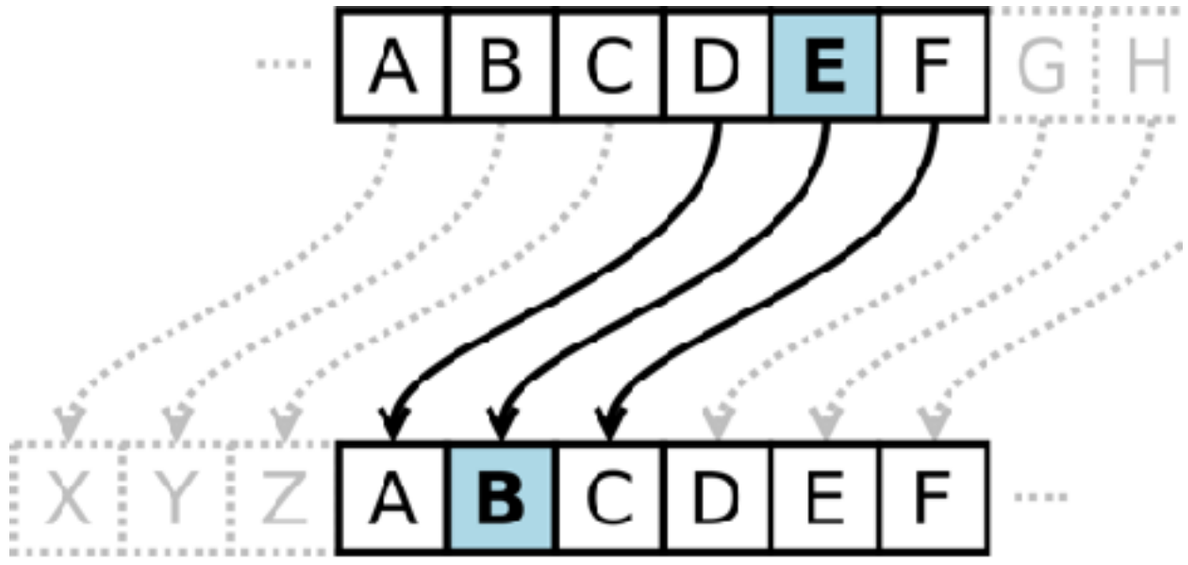
The Beginnings of Encryption, ~400BC

Scytale

Substitution Ciphers, c. 50BC + earlier

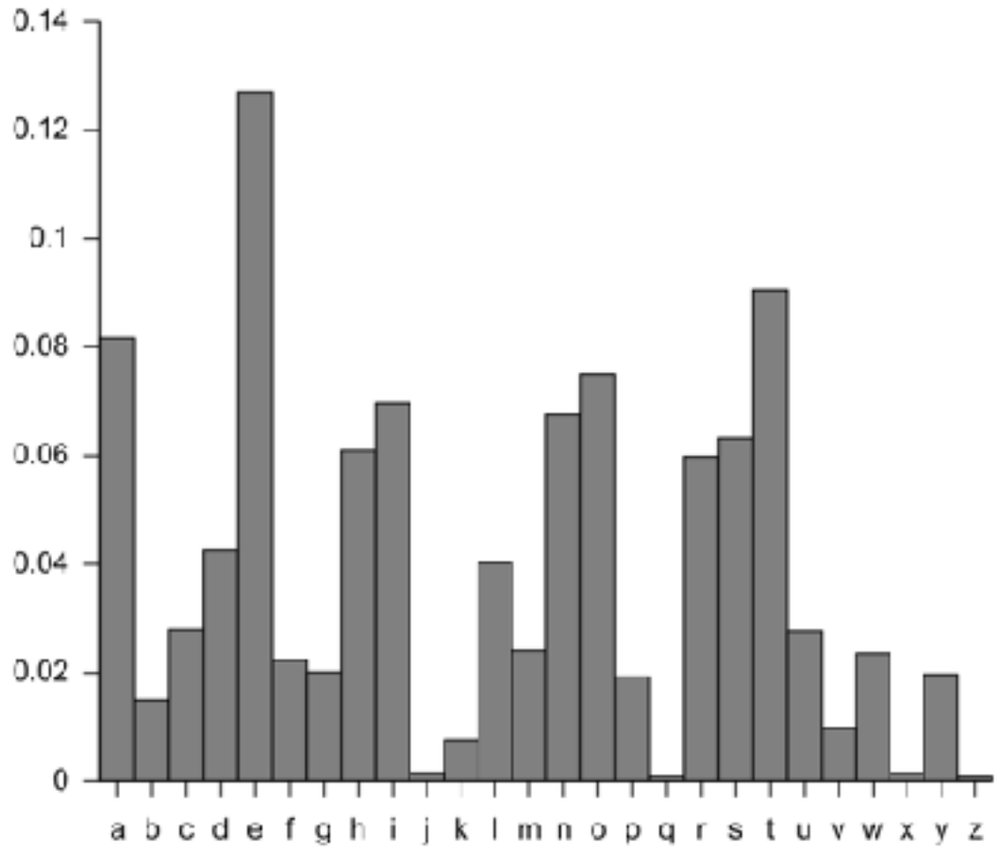


Julius Caesar (100-44BC)

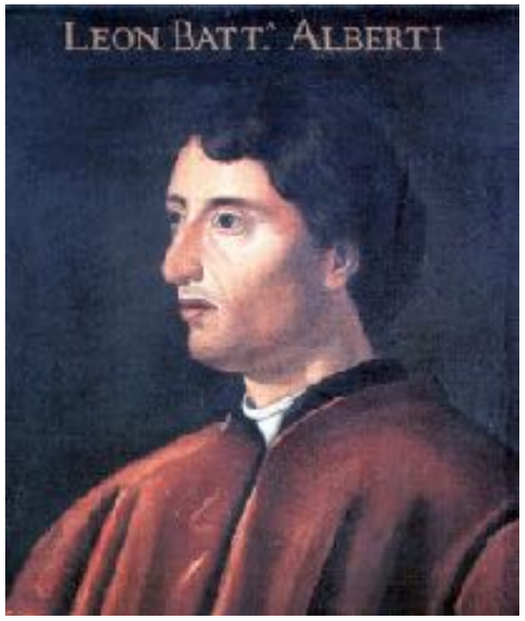


Frequency Analysis and Al-Kindi (801-873 AD)

The first page of al-Kindi's manuscript "On Deciphering Cryptographic Messages"

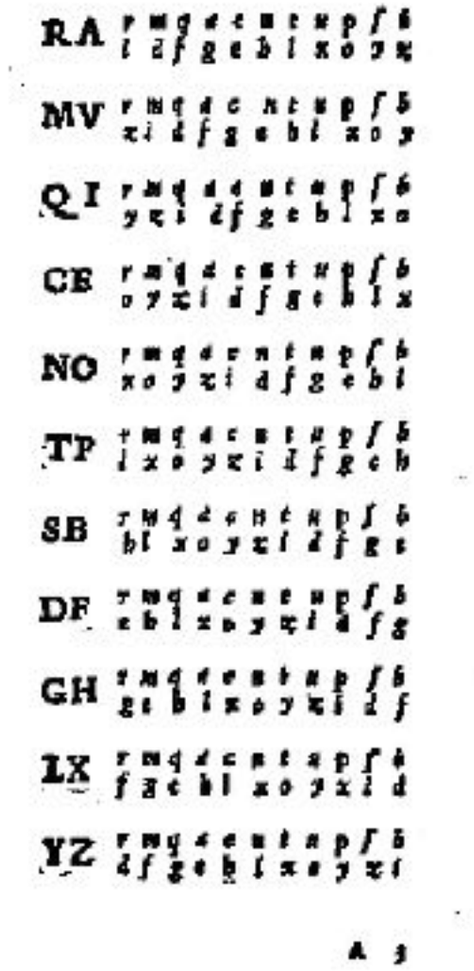


Polyalphabetic Ciphers: *Le Chiffre Indéchiffrable*



Leon Battista Alberti (1404-1472)

Blaise de Vigenère (1523-1596)
(not the inventor)



Homophonic Ciphers: *Le Grand Chiffre* (c. 1626-1811)

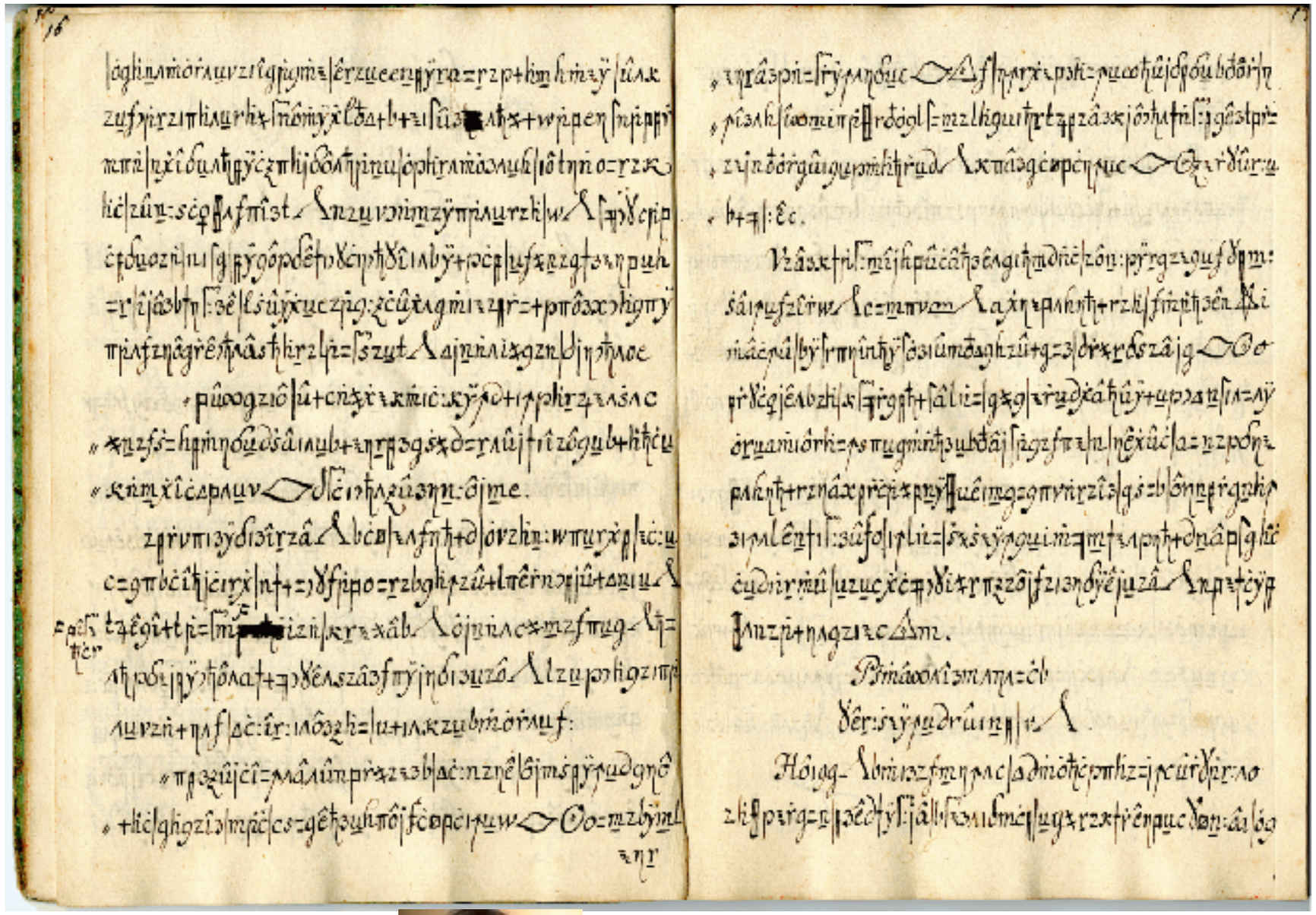
N	O	P	Q	R	S	T	V	X	Y	Z	&
811	117 238	219	407	511	355	360	141 163	205	518	820	277 448
702	359 500	338	595	733	527	618	284 164	436	639		615 827

Louis XIV (1638-1715)

Étienne Bazeries (1846-1931) broke it in 1890s(!)

Guessed that “124-22-125-46-345” stood for “*les en-ne-mie-s*”

Homophonic Ciphers: Copiale Cipher (1760)



Kevin Knight

Beáta Megyesi

+ Christiane Schaefer

Broken in 2011 using machine learning!

Mechanical-ciphers: c. 1900-1980s

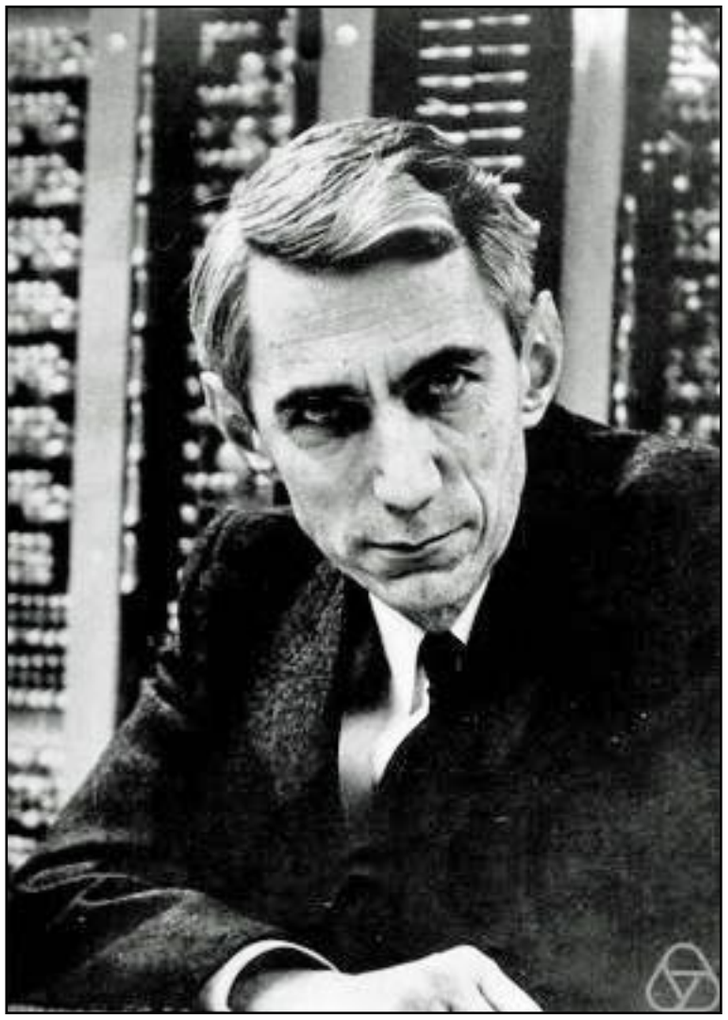
Cracking Enigma (early 30s — end of WWII)

Marian Rejewski (1905-1980)

Alan Turing (1912-1954)

Details Thursday!

Postwar Cryptography: Moving from Art to Science



Claude Shannon (1916-2001)

Communication Theory of Secrecy Systems*

By C. E. SHANNON

1. INTRODUCTION AND SUMMARY

THE problems of cryptography and secrecy systems furnish an interesting application of communication theory.[1] In this paper a theory of secrecy systems is developed. The approach is on a theoretical level and is intended to complement the treatment found in standard works on cryptography.[2] There, a detailed study is made of the many standard types of codes and ciphers, and of the ways of breaking them. We will be more concerned with the general mathematical structure and properties of secrecy systems.

The treatment is limited in certain ways. First, there are three general types of secrecy system: (1) concealment systems, including such methods as invisible ink, concealing a message in an innocent text, or in a fake covering cryptogram, or other methods in which the existence of the message is concealed from the enemy; (2) privacy systems, for example speech inversion, in which special equipment is required to recover the message; (3) "true" secrecy systems where the meaning of the message is concealed by cipher, code, etc., although its existence is not hidden, and the enemy is assumed to have any special equipment necessary to intercept and record the transmitted signal. We consider only the third type—concealment systems are primarily a psychological problem, and privacy systems a technological one.

Secondly, the treatment is limited to the case of discrete information, where the message to be enciphered consists of a sequence of discrete symbols, each chosen from a finite set. These symbols may be letters in a language, words of a language, amplitude levels of a "quantized" speech or video signal, etc., but the main emphasis and thinking has been concerned with the case of letters.

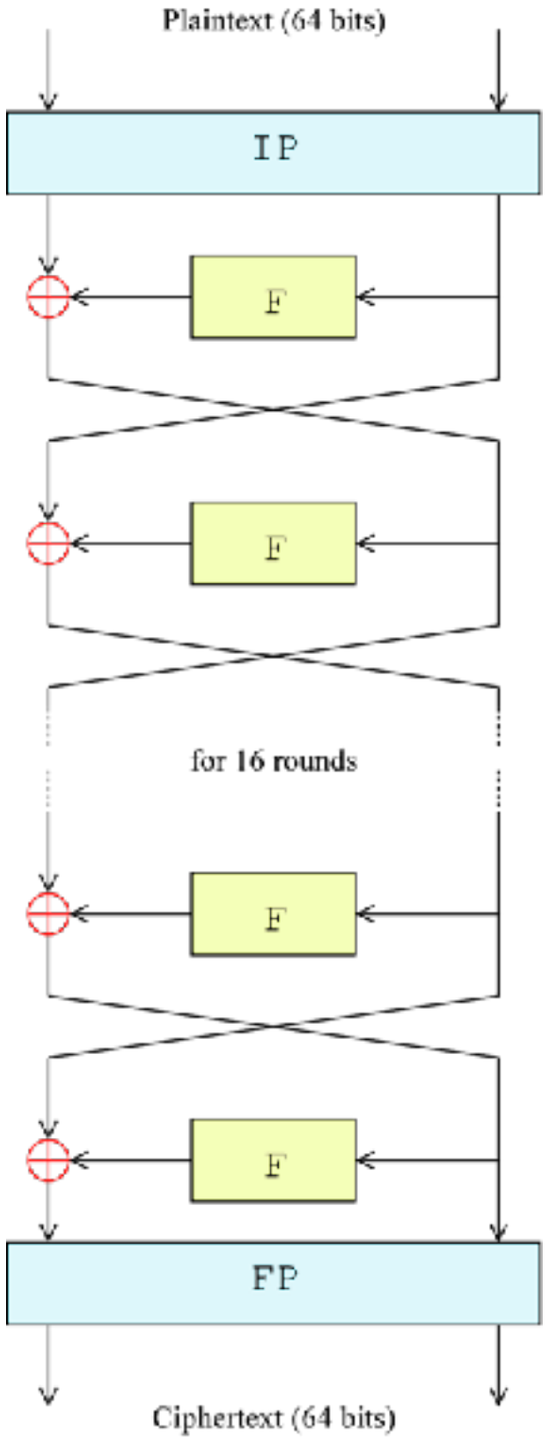
The paper is divided into three parts. The main results will now be briefly summarized. The first part deals with the basic mathematical structure of secrecy systems. As in communication theory a language is considered to

* The material in this paper appeared originally in a confidential report "A Mathematical Theory of Cryptography" dated Sept. 1, 1945, which has now been declassified.

[1] Shannon, C. E., "A Mathematical Theory of Communication," *Bell System Technical Journal*, July 1948, p. 379; Oct. 1948, p. 623.

[2] See, for example, H. F. Gaines, "Elementary Cryptanalysis," or M. Givierge, "Cours de Cryptographie."

The Modern Cryptography Era Begins: DES,1970s



IBM + NATIONAL SECURITY AGENCY * UNITED STATES OF AMERICA *

Horst Feistel (1915-1990)

Key Distribution Problem

My CC num =
4417 4001 7234 1189

amazon.com

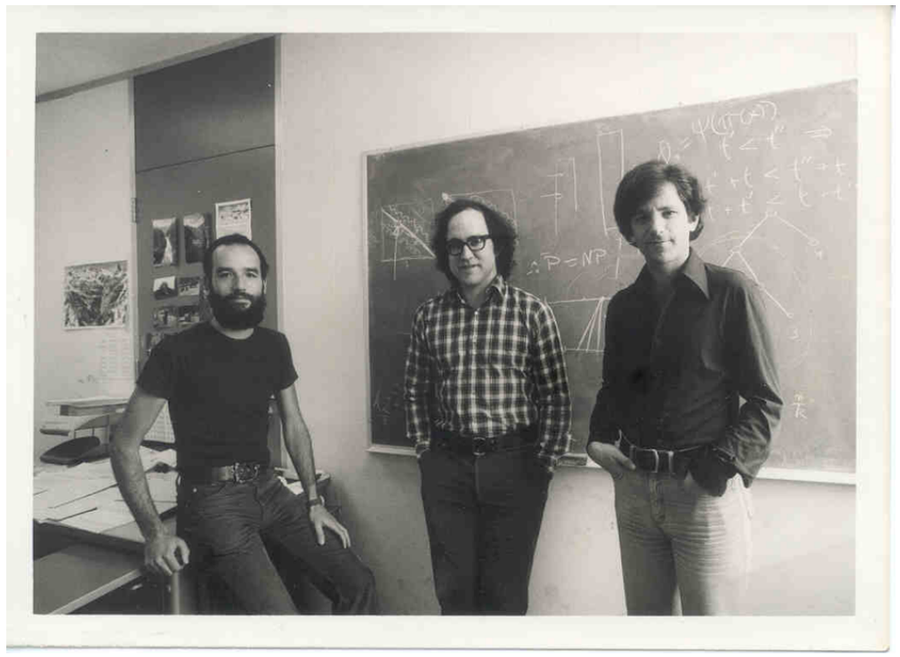
The Internet

The Public-Key Revolution (1978)

Basic question: If two people are talking in the presence of an eavesdropper, and they don't have pre-shared a key, is there any way they can send private messages?

Diffie and Hellman
in 1976: **Yes!**

Turing Award, 2015,
+ Million Dollars



Rivest, Shamir, Adleman
in 1978: **Yes, differently!**

Turing Award, 2002,
+ no money

Cocks, Ellis, Williamson
in 1969, at GCHQ:
Yes, we know about both…

Pat on the back?

Provable Security (1980s — present)

Shafi Goldwasser

Silvio Micali

Turing Award, 2012, + 250k Dollars

Reprinted from JOURNAL OF COMPUTER AND SYSTEM SCIENCES
All Rights Reserved by Academic Press, New York and London

Vol. 28, No. 2, April 1984
Printed in Belgium

Probabilistic Encryption*

SHAFI GOLDWASSER AND SILVIO MICALI

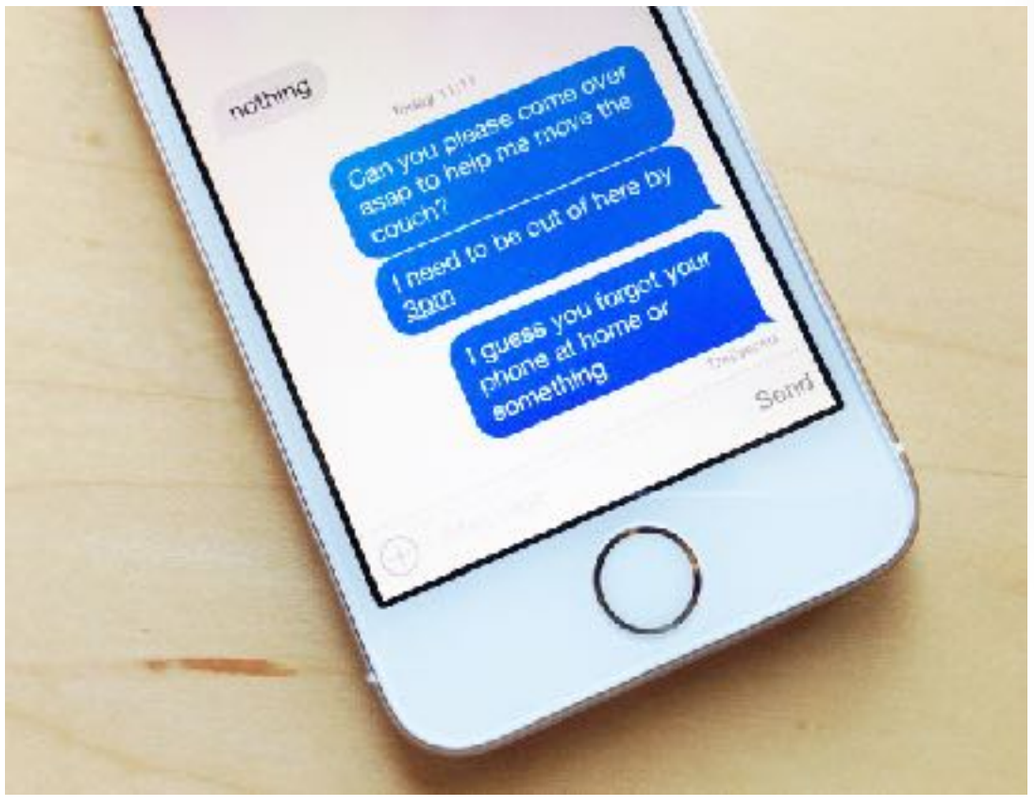
Laboratory of Computer Science, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139

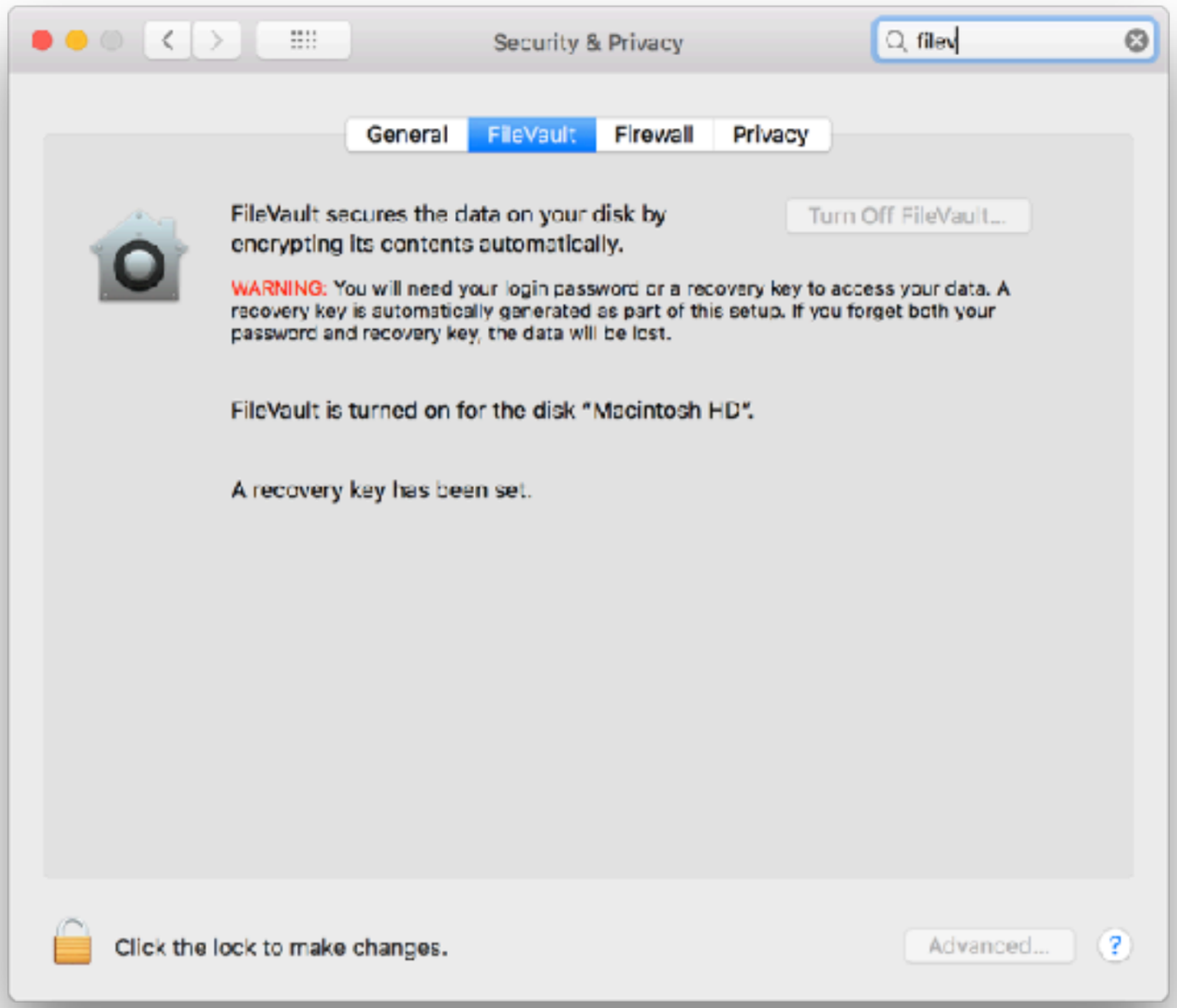
Received February 3, 1983; revised November 8, 1983

A new probabilistic model of data encryption is introduced. For this model, under suitable complexity assumptions, it is proved that extracting *any information* about the cleartext from the cyphertext is hard on the average for an adversary with polynomially bounded computational resources. The proof holds for any message space with any probability distribution. The first implementation of this model is presented. The security of this implementation is proved under the intractability assumption of deciding Quadratic Residuosity modulo composite numbers whose factorization is unknown.

Cryptowars of the 1990s

Crypto Today





Amazon.com: Online Shopp ×

https://www.amazon.com

www.amazon.com

Your connection to this site is private.

Details

Permissions | Connection

Chrome verified that Symantec Class 3 Secure Server CA - G4 issued this website's certificate. The server did not supply any Certificate Transparency information.

Certificate Information

Your connection to www.amazon.com is encrypted using a modern cipher suite.

The connection uses TLS 1.2.

The connection is encrypted and authenticated using AES_128_GCM and uses ECDHE_RSA as the key exchange mechanism.

What do these mean?

amazo

Departments

ON UPDATED DAILY

EXPLORE

zon.com Today's Deals Gift Cards

DESTINATIO

ENTERTAINME

fire $49

WIRED SUBSCRIBE

KIM ZETTER SECURITY 09.24.13 06:30 AM

How a Crypto 'Backdoor' Pitted the Tech World Against the NSA

ars TECHNICA

BIZ & IT —

NSA official: Support of backdoored Dual_EC_DRBG was “regrettable”

Agency supported crypto function for years after "trap door" was disclosed.

DAN GOODIN - 1/14/2015, 12:43 PM

Cryptowars of the 2010-2020s

MIT Technology Review

Computing / Cybersecurity

Barr's call for encryption backdoors has reawakened a years-old debate

Attorney General William Barr's speech on Tuesday reignited a dispute that's more relevant than ever.

by **Patrick Howell O'Neill** Jul 24, 2019

Attorney General William Barr

PHOTO: DEPARTMENT OF JUSTICE

CASEY CHIN; GETTY IMAGES

LEANDER KAHNEY BACKCHANNEL 04.16.2019 12:43 PM

The FBI Wanted a Backdoor to the iPhone. Tim Cook Said No

The agency wanted to crack the iPhone of Syed Farook, a suspect in the 2015 San Bernardino shooting. The Apple CEO took a stand.

Cryptography Today: An International Community











This Lecture: History and Classical Ciphers

1. A very quick history of cryptography
2. **Classical ciphers and how they are broken**
   - Shift cipher
   - Substitution cipher
   - Vigenère cipher
   - Homophonic cipher
   - One-time pad

Why Classical Ciphers?

All of these ciphers are broken. Why study them?

Definition: Cipher

Let **Keys**, **Msgs**, and **Ctxts** be sets. We call a function

$$E : \mathbf{Keys} \times \mathbf{Msgs} \to \mathbf{Ctxts}$$

a *cipher* if for all $K \in \mathbf{Keys}$, $\mathbf{E}(K,\cdot)$ is one-to-one.

We write $\mathbf{E}^{-1}(K,C)$ for inverse (note: not quite function inverse).

Letters as Numbers

Convenient to identify letters with numbers:

A	0
B	1
C	2
D	3
E	4
F	5
…	…
Z	25

Classical Cipher 1: Shift Cipher

Key: Number **K** between 0 and 25

E(K,M) shifts each letter of **M** forward **K** spots, wrapping if needed.

Example: Key=6

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
g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	a	b	c	d	e	f

Breaking the Shift Cipher: Exhaustive Key Search

Ciphertext: **`zkjxvkkzexcrkv`**

```
zkjxvkkzexcrkv
alkywllafydslw
bmlzxmmbgzetmx
cnmaynnchafuny
donbzoodibgvoz
epocappejchwpa
fqpdbqqfkdixqb
grqecrrglejyrc
hsrfdsshmfkzsd
itsgettinglate
juthfuujohmbuf
kvuigvvkpincvg
lwvjhwwlqjodwh
mxwkixxmrkpexi
nyxljyynslqfyj
ozymkzzotmrgzk
paznlaapunshal
qbaombbqvotibm
rcbpnccrwpujcn
sdcqoddsxqvkdo
tedrpeetyrwlep
ufesqffuzsxmfq
vgftrggvatyngr
whgushhwbuzohs
xihvtiixcvapit
yjiwujjydwbqju
```

Classical Cipher 2: Substitution Cipher

Key: Any permutation **π** of **{A,B,…,Z}**

E(π,M) applies **π** to each letter of **M**

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

E(π,david) = GFDTG

keys = 26! > 2^{88}, i.e. very large

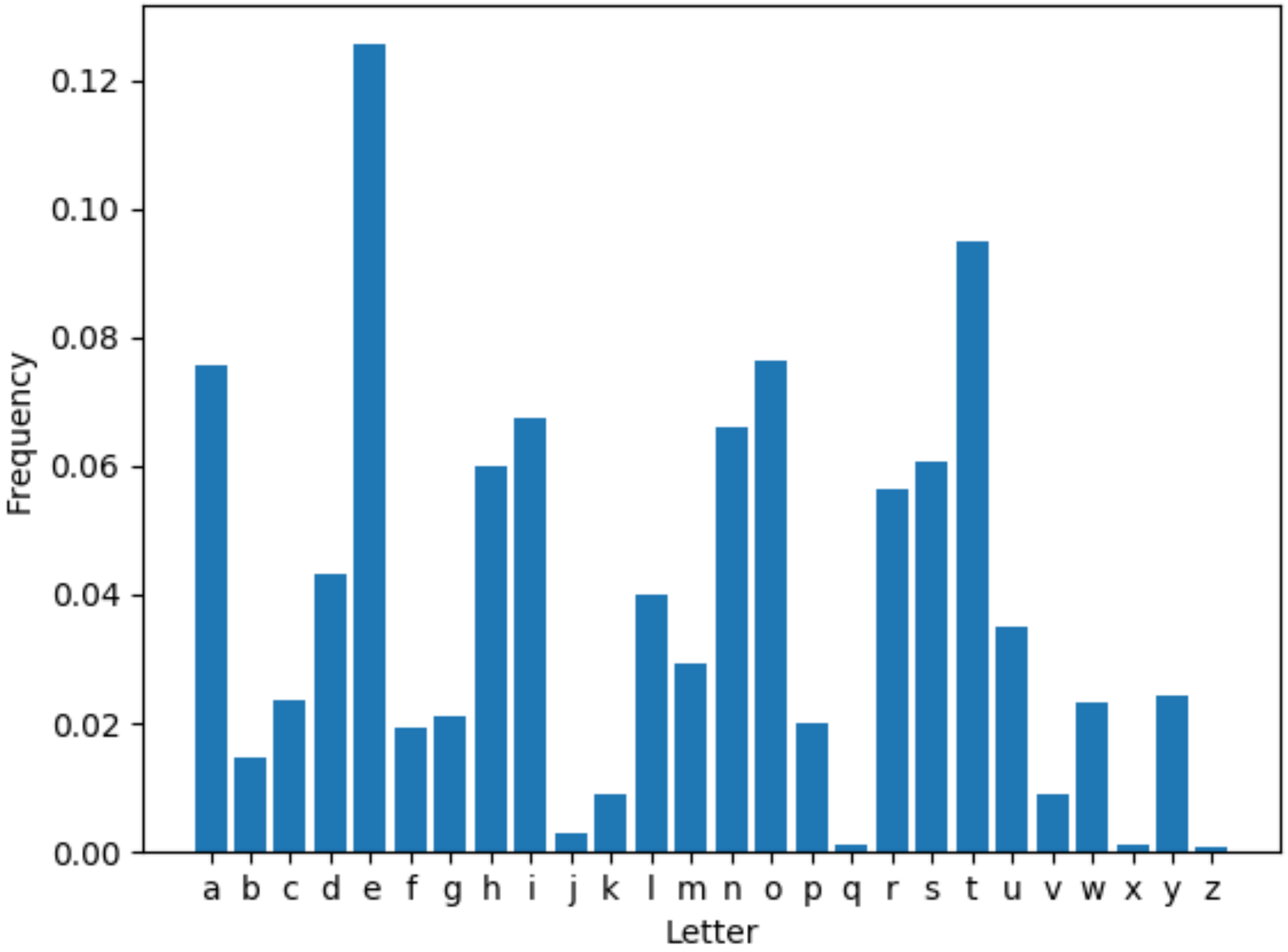
Breaking the Substitution Cipher

COXBX TBX CVK CDGXR DI T GTI'R ADHX VOXI OX ROKQAU IKC RNXPQATCX: VOXI OX PTI'C THHKBU DC, TIU VOXI OX PTI.

Exploit structure of English text:

- E is most common letter in English
- One-letter words must be "I" or "A"
- etc.

English Letter Frequencies



Step One: Letter Counts

COXBX TBX CVK CDGXR DI T GTI'R ADHX VOXI OX ROKQAU IKC RNXPQATCX: VOXI OX PTI'C THHKBU DC, TIU VOXI OX PTI.

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

Step One: Letter Counts

COXBX TBX CVK CDGXR DI T GTI'R ADHX VOXI OX ROKQAU IKC RNXPQATCX: VOXI OX PTI'C THHKBU DC, TIU VOXI OX PTI.

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

Step One: Letter Counts

COXBX TBX CVK CDGXR DI T GTI'R ADHX VOXI OX ROKQAU IKC RNXPQATCX: VOXI OX PTI'C THHKBU DC, TIU VOXI OX PTI.

A	B	C	D	E	F	G	H	I	J	K	L	M
3												

N	O	P	Q	R	S	T	U	V	W	X	Y	Z

Step One: Letter Counts

COXBX TBX CVK CDGXR DI T GTI'R ADHX VOXI OX ROKQAU IKC RNXPQATCX: VOXI OX PTI'C THHKBU DC, TIU VOXI OX PTI.

A	B	C	D	E	F	G	H	I	J	K	L	M
3	3											

N	O	P	Q	R	S	T	U	V	W	X	Y	Z

Step One: Letter Counts

COXBX TBX CVK CDGXR DI T GTI'R ADHX VOXI OX ROKQAU IKC RNXPQATCX: VOXI OX PTI'C THHKBU DC, TIU VOXI OX PTI.

A	B	C	D	E	F	G	H	I	J	K	L	M
3	3	7	4	0	0	2	3	9	0	4	0	0

N	O	P	Q	R	S	T	U	V	W	X	Y	Z
1	8	3	2	4	0	8	3	4	0	13	0	0

E E E E E E

COXBX TBX CVK CDGXR DI T GTI'R ADHX VOXI

E E E E E

OX ROKQAU IKC RNXPQATCX: VOXI OX PTI'C

E E

THHKBU DC, TIU VOXI OX PTI.

A	B	C	D	E	F	G	H	I	J	K	L	M
3	3	7	4	0	0	2	3	9	0	4	0	0

N	O	P	Q	R	S	T	U	V	W	X	Y	Z
I	8	3	2	4	0	8	3	4	0	I3	0	0

Input:	A	B	C	D	E	F	G	H	I	J	K	L	M
Maps to:													

Input:	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
Maps to:													

```
  E E   E        E                E   E
COXBX TBX CVK CDGXR DI T GTI’R ADHX VOXI

 E              E     E    E   E
OX ROKQAU IKC RNXPQATCX: VOXI OX PTI’C

                 E   E
THHKBU DC, TIU VOXI OX PTI.
```

A	B	C	D	E	F	G	H	I	J	K	L	M
3	3	7	4	0	0	2	3	9	0	4	0	0

N	O	P	Q	R	S	T	U	V	W	X	Y	Z
1	8	3	2	4	0	8	3	4	0	13	0	0

Input:	A	B	C	D	E	F	G	H	I	J	K	L	M
Maps to:													

Input:	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
Maps to:											E		

E E E E E E

COXBX TBX CVK CDGXR DI T GTI'R ADHX VOXI

E E E E E

OX ROKQAU IKC RNXPQATCX: VOXI OX PTI'C

E E

THHKBU DC, TIU VOXI OX PTI.

Input:	A	B	C	D	E	F	G	H	I	J	K	L	M
Maps to:													

Input:	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
Maps to:											E		

OX appears in ciphertext, so **O** represents **B**,**H**,**M**, or **W**

⇒There are lots of **O**s, so guess it represents **H**

```
  HE E   E          E                          E   HE
COXBX TBX CVK CDGXR DI T GTI'R ADHX VOXI

HE  H             E       E   HE  HE
OX ROKQAU IKC RNXPQATCX: VOXI OX PTI'C

                  HE  HE
THHKBU DC, TIU VOXI OX PTI.
```

Input:	A	B	C	D	E	F	G	H	I	J	K	L	M
Maps to:													

Input:	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
Maps to:		H									E		

OX appears in ciphertext, so **O** represents **B**,**H**,**M**, or **W**

⇒There are lots of **O**s, so guess it represents **H**

```
  HE  E    E          E                                  E   HE
COXBX TBX CVK CDGXR DI T GTI'R ADHX VOXI

HE  H             E       E   HE  HE
OX ROKQAU IKC RNXPQATCX: VOXI OX PTI'C

                  HE  HE
THHKBU DC, TIU VOXI OX PTI.
```

Input:	A	B	C	D	E	F	G	H	I	J	K	L	M
Maps to:													

Input:	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
Maps to:		H									E		

***HE*E**

COXBX Could be: **THERE**, **THESE**, **WHERE** ...

⇒Guess **C** represents **T** because no "?" was used

THE E E T T E E HE
COXBX TBX CVK CDGXR DI T GTI'R ADHX VOXI

HE H E E HE HE T
OX ROKQAU IKC RNXPQATCX: VOXI OX PTI'C

T HE HE
THHKBU DC, TIU VOXI OX PTI.

Input:	A	B	C	D	E	F	G	H	I	J	K	L	M
Maps to:			T										

Input:	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
Maps to:		H									E		

*HE*E

COXBX Could be: **THERE**, **THESE**, **WHERE** ...

⇒Guess **C** represents **T** because no "?" was used

THE E E T T E E HE
COXBX TBX CVK CDGXR DI T GTI'R ADHX VOXI

HE H E E HE HE T
OX ROKQAU IKC RNXPQATCX: VOXI OX PTI'C

T HE HE
THHKBU DC, TIU VOXI OX PTI.

Input:	A	B	C	D	E	F	G	H	I	J	K	L	M
Maps to:			T										

Input:	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
Maps to:		H									E		

Now **B** represents either **R** or **S**

So **TBX** is ***RE** or ***SE**

⇒Guess **T** represents **A** and **B** represents **R**

THERE ARE T T E A A E HE
COXBX TBX CVK CDGXR DI T GTI'R ADHX VOXI

HE H T E A E HE HE A T
OX ROKQAU IKC RNXPQATCX: VOXI OX PTI'C

A R T A HE HE A
THHKBU DC, TIU VOXI OX PTI.

Input:	A	B	C	D	E	F	G	H	I	J	K	L	M
Maps to:		R	T										

Input:	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
Maps to:		H					A				E		

Now **B** represents either **R** or **S**

So **TBX** is ***RE** or ***SE**

⇒Guess **T** represents **A** and **B** represents **R**

THERE ARE T T E A A E HE
COXBX TBX CVK CDGXR DI T GTI'R ADHX VOXI

HE H T E A E HE HE A T
OX ROKQAU IKC RNXPQATCX: VOXI OX PTI'C

A R T A HE HE A
THHKBU DC, TIU VOXI OX PTI.

Input:	A	B	C	D	E	F	G	H	I	J	K	L	M
Maps to:		R	T										

Input:	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
Maps to:		H					A				E		

Next **DC** = ***T**

⇒ **D** can't be **A**, so it must be **I**

THERE ARE T TI E I A A I E HE
COXBX TBX CVK CDGXR DI T GTI'R ADHX VOXI

HE H T E A E HE HE A T
OX ROKQAU IKC RNXPQATCX: VOXI OX PTI'C

A R IT A HE HE A
THHKBU DC, TIU VOXI OX PTI.

Input:	A	B	C	D	E	F	G	H	I	J	K	L	M
Maps to:		R	T	I									

Input:	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
Maps to:		H					A				E		

Next **DC** = ***T**

⇒ **D** can't be **A**, so it must be **I**

THERE ARE T TI E I A A I E HE
COXBX TBX CVK CDGXR DI T GTI'R ADHX VOXI

HE H T E A E HE HE A T
OX ROKQAU IKC RNXPQATCX: VOXI OX PTI'C

A R IT A HE HE A
THHKBU DC, TIU VOXI OX PTI.

Input:	A	B	C	D	E	F	G	H	I	J	K	L	M
Maps to:		R	T	I									

Input:	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
Maps to:		H					A				E		

Etc etc...

THERE ARE TWO TIMES IN A MAN'S LIFE WHEN
COXBX TBX CVK CDGXR DI T GTI'R ADHX VOXI

HE SHOULD NOT SPECULATE: WHEN HE CAN'T
OX ROKQAU IKC RNXPQATCX: VOXI OX PTI'C

AFFORD IT, AND WHEN HE CAN
THHKBU DC, TIU VOXI OX PTI.

Input:	A	B	C	D	E	F	G	H	I	J	K	L	M
Maps to:	L	R	T	I			M	F	N		O		

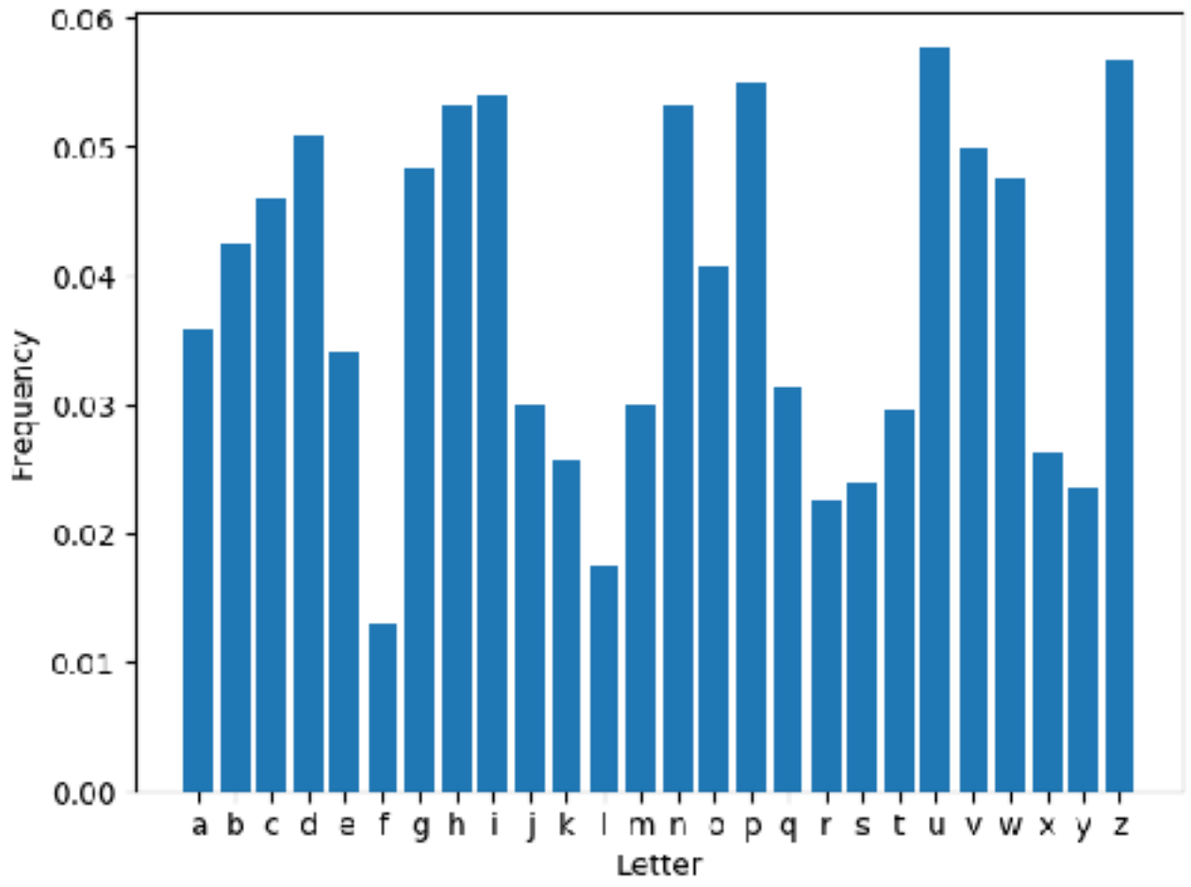
Input:	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
Maps to:	P	H	C	U	S		A	D	W		E		

Classical Cipher 3: Vigenère Cipher

Key: Secret word of some length, ex: **CRYPTO**

E(K,M) shifts letters of M using secret word as follows:

```
Key: CRYPTOCRYPTOCRYPTOCRYPTOCRYPTOCRYPTOCRYPTO
Msg: acannercancanasmanycansasacannercancancans
Ctxt: CTYCGSTTYCVOPRQBTBATYCLOURAPGBGIAPGQCEAPGG
```



Finding the Keylength of Vigenère: First Way

```
Key:CRYPTOCRYPTOCRYPTOCRYPTOCRYPTOCRYPTOCRYPTO
Msg:acannercancanasmanycansasacannercancancans
Ctxt:CTYCGSTTYCVOPRQBTBATYCLOURAPGBGIAPGQCEAPGG
```

```
     CTYCGSTTYCVOPRQBTBATYCLOURAPGBGIAPGQCEAPGG
```

Matching chunks start multiples of 6 apart.

Key length probably divides 6, i.e. is 1,2,3 or 6.

Finding the Keylength of Vigenère: Second Way

	#matches
CTYCGSTTYCVOPRQBTBATYCLOURAPGBGIAPGQCEAPGG	
CTYCGSTTYCVOPRQBTBATYCLOURAPGBGIAPGQCEAPGG CTYCGSTTYCVOPRQBTBATYCLOURAPGBGIAPGQCEAPGG	2
CTYCGSTTYCVOPRQBTBATYCLOURAPGBGIAPGQCEAPGG CTYCGSTTYCVOPRQBTBATYCLOURAPGBGIAPGQCEAPGG	0
CTYCGSTTYCVOPRQBTBATYCLOURAPGBGIAPGQCEAPGG CTYCGSTTYCVOPRQBTBATYCLOURAPGBGIAPGQCEAPGG	2
CTYCGSTTYCVOPRQBTBATYCLOURAPGBGIAPGQCEAPGG CTYCGSTTYCVOPRQBTBATYCLOURAPGBGIAPGQCEAPGG	1
CTYCGSTTYCVOPRQBTBATYCLOURAPGBGIAPGQCEAPGG CTYCGSTTYCVOPRQBTBATYCLOURAPGBGIAPGQCEAPGG	1
CTYCGSTTYCVOPRQBTBATYCLOURAPGBGIAPGQCEAPGG CTYCGSTTYCVOPRQBTBATYCLOURAPGBGIAPGQCEAPGG	9 ←

Classical Cipher 4: Homophonic Substitution Cipher

Key: A table **K** giving possible substitutions for each letter

E(K,M) replaces each letter of **M** with an option from table

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	0	M	1	A	S	N	U	Q	G	7	T	V	I	6	P	Y	9	E	Z	K	4	X	F	W	L
R				H			B	8					2	C			J	O	5						
3																									

Can use multiple letters/digits per message letter!

Breaking homophonic ciphers: Use bi-gram and tri-gram frequencies. Usually very hard without a lot of text.

Classical Cipher 5: One-Time Pad Cipher

Just Vigenère, under a different usage.

Key: Random letter string *as long as the intended message*

E(K,M) works just like Vigenère

One-Time Pad Cipher: Binary Version

Let x,y be bit strings of length n.

Define x⊕y to be their bit-wise XOR

ex:

	01011	01011	01011
⊕	10010	01011	00000
	11001	00000	01011

Message: Bit string **M**
Key: Random bit string **K** *as long as the intended message*

$E(K,M) = K \oplus M$

$E^{-1}(K,C) = K \oplus C$ ⟵ Same as encryption!

$E^{-1}(K,C) = K \oplus C = K \oplus K \oplus M = 0^n \oplus M = M$

Theorem: (Week 3) The one-time pad cipher is “perfectly secret”.

ASCII/UTF-8 Encoding

Represents printable and non-printable characters as 8-bit bytes

Dec	Bin	Hex	Char	Dec	Bin	Hex	Char	Dec	Bin	Hex	Char	Dec	Bin	Hex	Char
0	0000 0000	00	[NUL]	32	0010 0000	20	space	64	0100 0000	40	@	96	0110 0000	60	`
1	0000 0001	01	[SOH]	33	0010 0001	21	!	65	0100 0001	41	A	97	0110 0001	61	a
2	0000 0010	02	[STX]	34	0010 0010	22	"	66	0100 0010	42	B	98	0110 0010	62	b
3	0000 0011	03	[ETX]	35	0010 0011	23	#	67	0100 0011	43	C	99	0110 0011	63	c
4	0000 0100	04	[EOT]	36	0010 0100	24	$	68	0100 0100	44	D	100	0110 0100	64	d
5	0000 0101	05	[ENQ]	37	0010 0101	25	%	69	0100 0101	45	E	101	0110 0101	65	e
6	0000 0110	06	[ACK]	38	0010 0110	26	&	70	0100 0110	46	F	102	0110 0110	66	f
7	0000 0111	07	[BEL]	39	0010 0111	27	'	71	0100 0111	47	G	103	0110 0111	67	g
8	0000 1000	08	[BS]	40	0010 1000	28	(	72	0100 1000	48	H	104	0110 1000	68	h
9	0000 1001	09	[TAB]	41	0010 1001	29	)	73	0100 1001	49	I	105	0110 1001	69	i
10	0000 1010	0A	[LF]	42	0010 1010	2A	*	74	0100 1010	4A	J	106	0110 1010	6A	j
11	0000 1011	0B	[VT]	43	0010 1011	2B	+	75	0100 1011	4B	K	107	0110 1011	6B	k
12	0000 1100	0C	[FF]	44	0010 1100	2C	,	76	0100 1100	4C	L	108	0110 1100	6C	l
13	0000 1101	0D	[CR]	45	0010 1101	2D	-	77	0100 1101	4D	M	109	0110 1101	6D	m
14	0000 1110	0E	[SO]	46	0010 1110	2E	.	78	0100 1110	4E	N	110	0110 1110	6E	n
15	0000 1111	0F	[SI]	47	0010 1111	2F	/	79	0100 1111	4F	O	111	0110 1111	6F	o
16	0001 0000	10	[DLE]	48	0011 0000	30	0	80	0101 0000	50	P	112	0111 0000	70	p
17	0001 0001	11	[DC1]	49	0011 0001	31	1	81	0101 0001	51	Q	113	0111 0001	71	q
18	0001 0010	12	[DC2]	50	0011 0010	32	2	82	0101 0010	52	R	114	0111 0010	72	r
19	0001 0011	13	[DC3]	51	0011 0011	33	3	83	0101 0011	53	S	115	0111 0011	73	s
20	0001 0100	14	[DC4]	52	0011 0100	34	4	84	0101 0100	54	T	116	0111 0100	74	t
21	0001 0101	15	[NAK]	53	0011 0101	35	5	85	0101 0101	55	U	117	0111 0101	75	u
22	0001 0110	16	[SYN]	54	0011 0110	36	6	86	0101 0110	56	V	118	0111 0110	76	v
23	0001 0111	17	[ETB]	55	0011 0111	37	7	87	0101 0111	57	W	119	0111 0111	77	w
24	0001 1000	18	[CAN]	56	0011 1000	38	8	88	0101 1000	58	X	120	0111 1000	78	x
25	0001 1001	19	[EM]	57	0011 1001	39	9	89	0101 1001	59	Y	121	0111 1001	79	y
26	0001 1010	1A	[SUB]	58	0011 1010	3A	:	90	0101 1010	5A	Z	122	0111 1010	7A	z
27	0001 1011	1B	[ESC]	59	0011 1011	3B	;	91	0101 1011	5B	[	123	0111 1011	7B	{
28	0001 1100	1C	[FS]	60	0011 1100	3C	<	92	0101 1100	5C	\	124	0111 1100	7C	|
29	0001 1101	1D	[GS]	61	0011 1101	3D	=	93	0101 1101	5D	]	125	0111 1101	7D	}
30	0001 1110	1E	[RS]	62	0011 1110	3E	>	94	0101 1110	5E	^	126	0111 1110	7E	~
31	0001 1111	1F	[US]	63	0011 1111	3F	?	95	0101 1111	5F	_	127	0111 1111	7F	[DEL]

Big No-No: “Reusing” The One-Time Pad

```
Msg1: There is a theory which states …
Msg2: You cant trust code that you …
```

```
Key: af9591e1cbda5f5225b9a0508640846f40876078d0df874…
(hex)
```

```
Ctxt1: 350e5c475c110c42430344155d035 …
Ctxt2: 38094c155a500b16174210134 0154 …
(hex)
```

Observation: C1⊕C2=(K⊕M1)⊕(K⊕M2)=M1⊕M2 (!)

Breaking the "Reused" One-Time Pad: Crib-Dragging

Idea:

1. Guess that some common English fragment (a *crib*) appears at a point in **`Msg1`**.
2. Under this assumption, we get a guess for that part of the key
3. Try decrypting **`Ctxt2`** with partial key guess.
4. If we get intelligible, printable characters, probably correct!

If text follows a format (like html), then try guessing cribs like
 and .

Crib-Dragging Example

```
Msg1: There is a theory which states …
Msg2: You cant trust code that you …

 Key: af9591e1cbda5f5225b9a0508640846f40876
(hex) 078d0df874680ab8316dfe05cea795 …

Ctxt1: 350e5c475c110c42430344155d035 …
Ctxt2: 38094c155a500b16174210134 0154 …
(hex)
```

1. Guess this part decrypts to "**The**"
2. Infer this part of key would be **0xaf9591**
3. Try decrypting this part with **0xaf9591**
4. Discover (ASCII encoding of) "**You**"
5. Conclude correct, move on to other parts

Classical Ciphers: Conclusions

In order to build secure ciphers, we…

1. Need a large number of possible keys
2. Need precise specifications for how they should be used
    - Ex: Only use a one-time pad once!
3. Should assume adversary “knows everyone but the key”
    - Known as “Kerkchoffs’s Principle”

Auguste Kerckhoffs
(1835-1903)

The End