



# History of Cryptography

Levente Buttyán

CrySyS Lab, BME

[buttyan@crysys.hu](mailto:buttyan@crysys.hu)

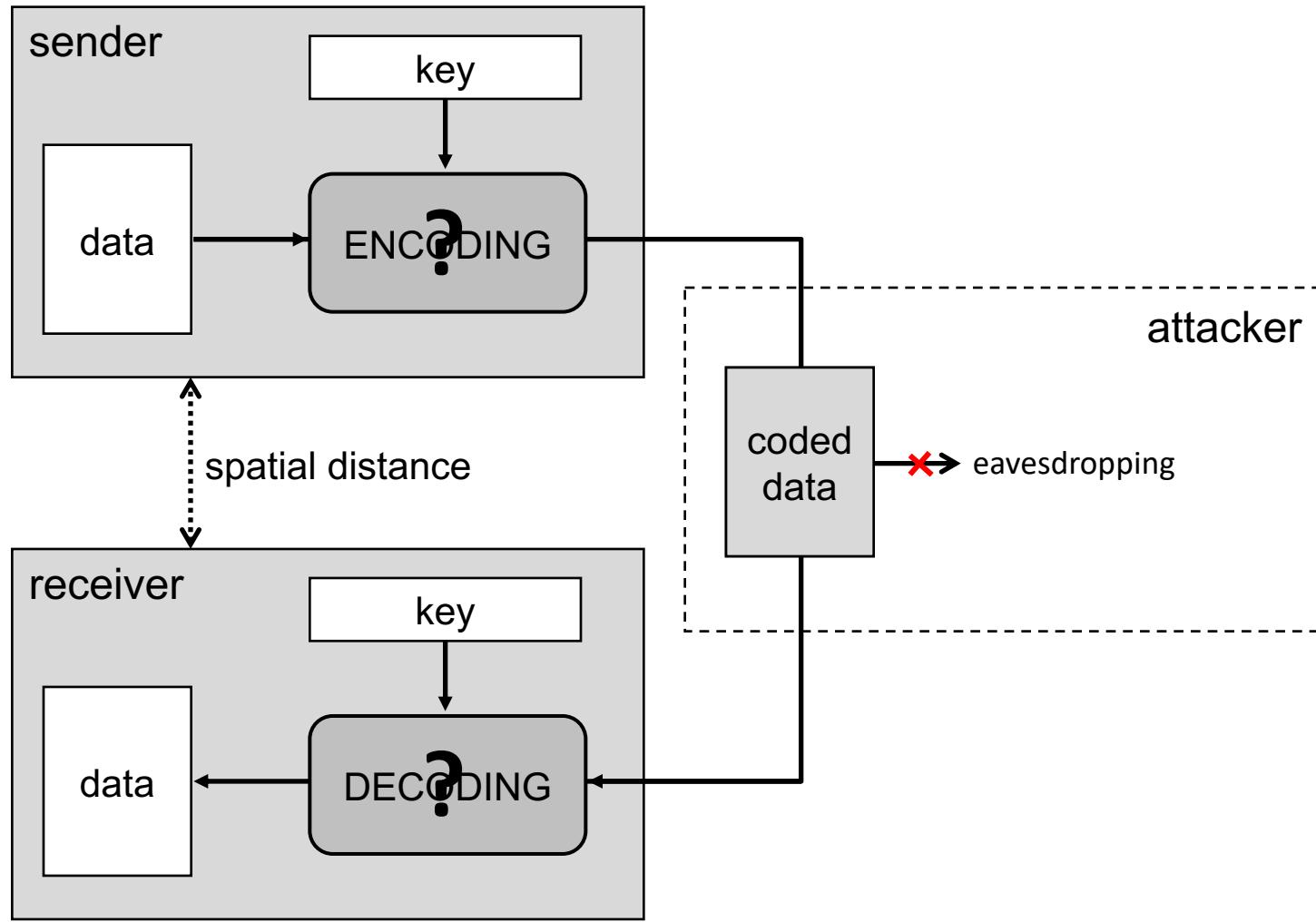
# History of crypto in a nutshell

---

- until the second half of the 20th century:
  - cryptography = encryption, ciphers
  - almost exclusively used in military and diplomacy
- from the second half of the 20th century:
  - cryptography is increasingly used in business applications (banking, electronic funds transfer)
  - besides confidentiality, integrity protection, authentication, and non-repudiation became important too
- from the end of the 20th century:
  - cryptography is used in everyday life of people (although they may be unaware of that)
    - » SSL/TLS – secure web transactions
    - » GSM/.../4G security – subscriber authentication, encryption on the air interface
    - » WiFi, Bluetooth, smart cards, disk encryption, ...

# Basic model

---

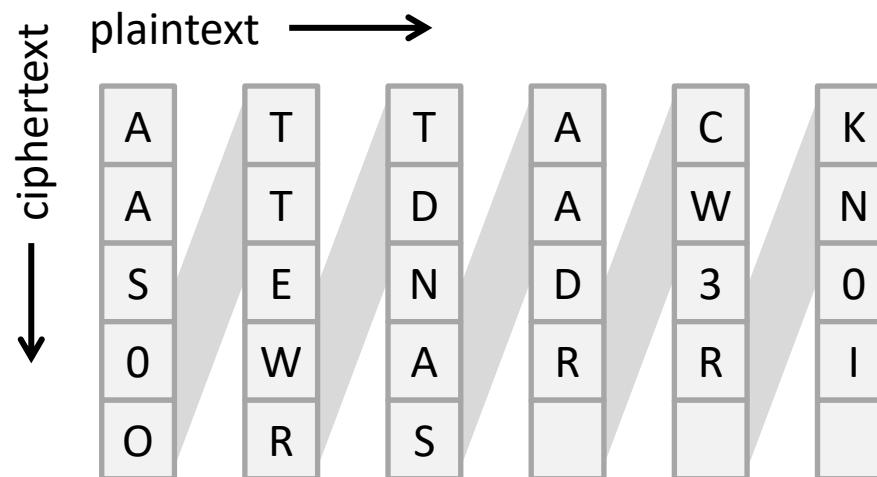


# Skytale

---



- used by the Spartans in the 3rd century BC
- *transposition cipher* (mixes letters of the plaintext)



# Skytale

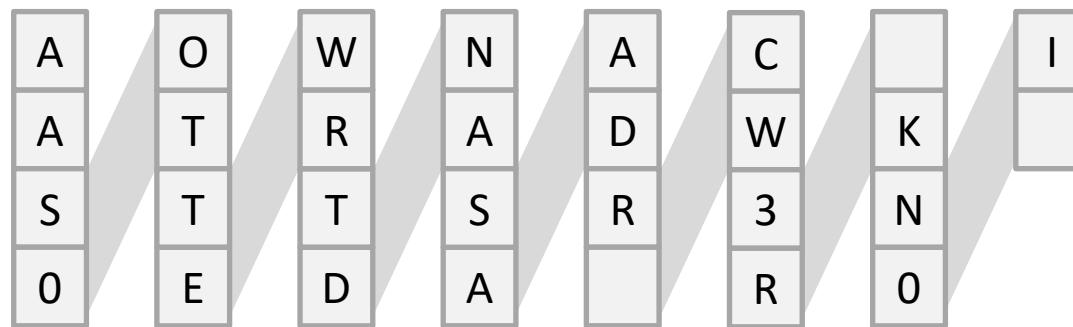
---



- what is the key?  
it is the (diameter of the) rod
- key space is small → easy to break

**using a rod with a wrong diameter:**

reading the plaintext →



# Caesar cipher

---



Crypto Museum  
[cryptomuseum.com](http://cryptomuseum.com)



# Caesar cipher

---

- used by Julius Caesar
- *substitution cipher* (replaces letters of the plaintext)
- each letter is replaced by the letter at some fixed number of positions (e.g., 3) down in the alphabet

plain:            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

cipher:          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

example:        BRUTUS → EUXWXV

- the key is the value of the shift (of the alphabet)
- what is the size of the key space?  
size of the key space is  $26 - 1 = 25 \rightarrow$  easy to break

# Monoalphabetic substitution

---

- generalization of the Caesar cipher
- replacement of letters is determined by a permutation

plain: 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

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

example: BRUTUS → TVXPXF

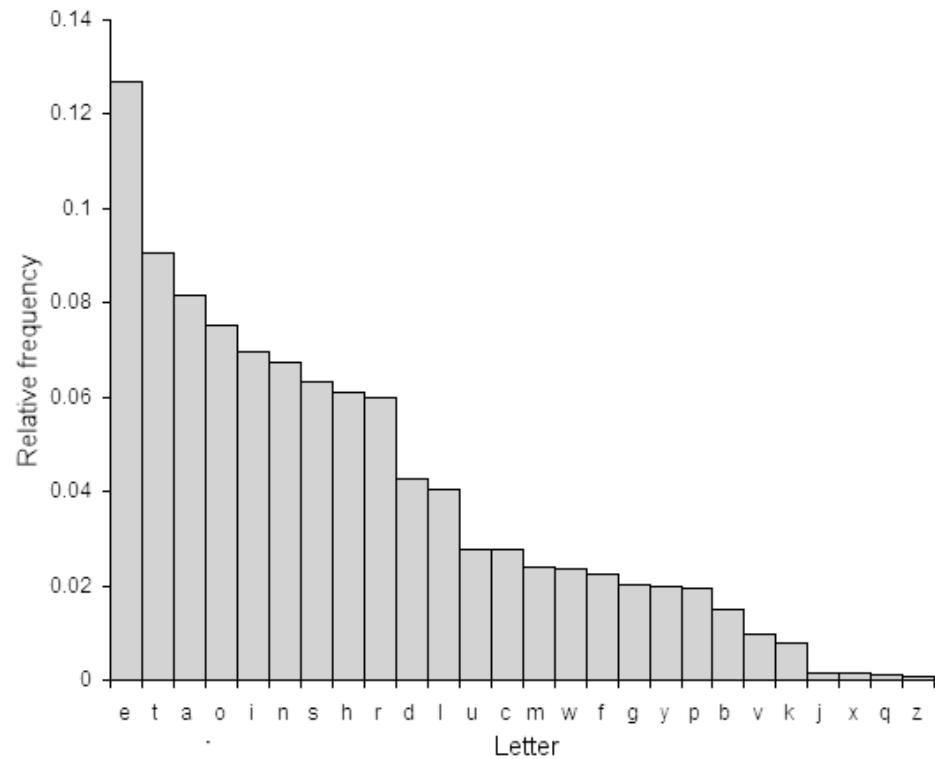
- the key is the permutation
- the key space is huge:  $26! \sim 1.56 \cdot 2^{88}$

- » time left until the next ice age .....  $2^{39}$  sec
- » time left until the Sun becomes a supernova .....  $2^{55}$  sec
- » age of the Earth .....  $2^{55}$  sec
- » age of the Universe .....  $2^{59}$  sec

# Breaking monoalphabetic substitutions

---

- every language has its own letter statistics
  - letter frequencies are independent of the actual text
  - there are letters that are more frequent than others
    - e.g., in English:  
 $e \rightarrow 12.7\%$ ,  $t \rightarrow 9.1\%$
  - and letters that are less frequent
    - e.g., in English:  
 $z \rightarrow 0.1\%$ ,  $j \rightarrow 0.2\%$
- in case of monoalphabetic substitution, the ciphertext preserves the letter statistics of the original plaintext!
  - after decoding the most frequent and least frequent letters, the rest of the text can be figured out much like solving a crossword puzzle



# If you wanna try...

---

RHINT TLHYF EIXQF AXTNK AFYFV QTRNF EHFEI XQFAD  
TXXTE ZAFBA IRRHI NTLQR IAIIMT EQNIA ITEFR HINTM  
IASGI NXSET EIACR HINTX EIARH FEXRT EOQNT RNFMH  
IAUKT ICWSI EXTET CQAKI BXKFE ATNIY QGFEA QISAQ  
XTCRX IXTRQ XBIRG FSACT CQADU XTRYI MFXFE RNTFI  
ACGFE MTEHI UHIYT AXETH ETATS ETYFA MSRZB QXKXK  
TVFIY FGNET IXQAV XKTXT NKAFY FVQTR XFETC SNTRH  
INTXE IARHF EXIXQ FANFR XRIAC TAIDY TXKTN FYFAQ  
JIXQF AFGMI ERQXK IRCTO TYFHT CXKTG IYNFA QIACG  
IYNFA QLYIS ANKOT KQNYT RDFXK CTRQV ATCXF DTETS  
RIDYT IACXK TCEIV FARHI NTNEI GBXKQ NKQRG YFBAQ  
AXFFE DQXDU XKTGI YNFAQ LYISA NKOTK QNYTX FRSHH  
YUXKT QAXTE AIXQF AIYRH INTRX IXQFA QRRBQ XKNIE  
VFIMI AATCO TERQF AFGCE IVFAQ RQACT OTYFH MTAXR  
HINTL QRGSA CTCDU VFOTE AMTAX RSDRQ CQTRI ACNFA  
XEINX RBQXK MSYXQ HYTTA XQXQT RRHIN TLRIN KQTOT  
MTAXR QANYS CTXKT GQERX HEQOI XTYUG SACTC YQWSQ  
CHEFH TYYIA XEFNZ TXGIY NFAXF ETINK FEDQX XKTGQ  
ERXHE QOIXT YUGSA CTCNF MHIAU XFRSN NTRRG SYYUY  
ISANK FEDQX IACET NFOTE IRHIN TNEIG XCEIV FAIAC  
XKTGQ ERXHE QOIXT NFMHI AUXFR TACIR HINTN EIGXC  
EIVFA XFXKT QRRXK TYISA NKFGR TROQQ QBIRX KTGQE  
RXRHI NTLCT YQOTE UQAXF VTFRU ANKEF AFSRF EDQXB

...

\* the plaintext is an English text

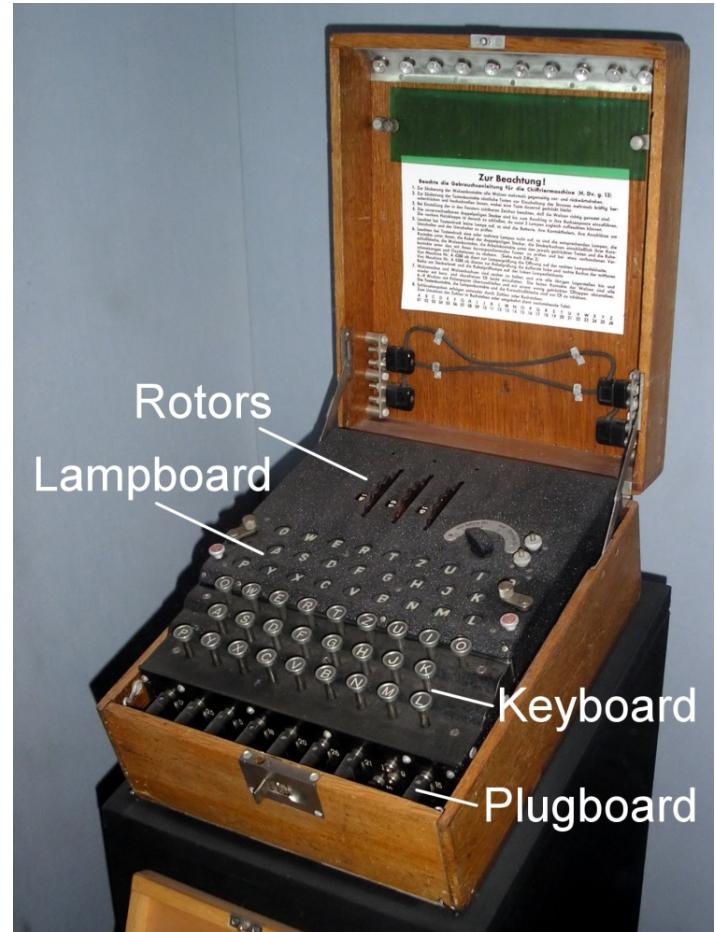
# The Enigma

- first electro-mechanical cipher
- adopted by the German Army in 1926
- used heavily by Germans in WWII

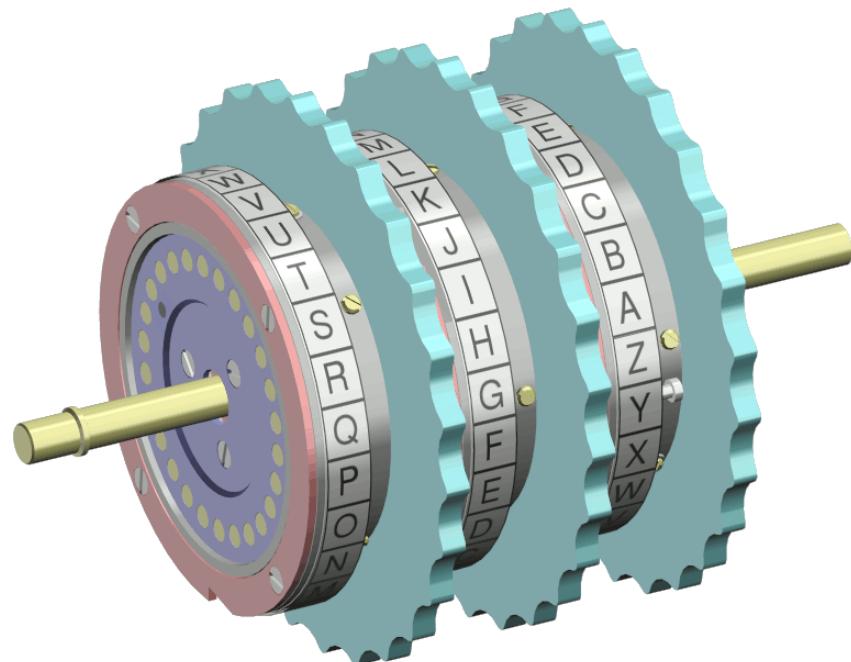
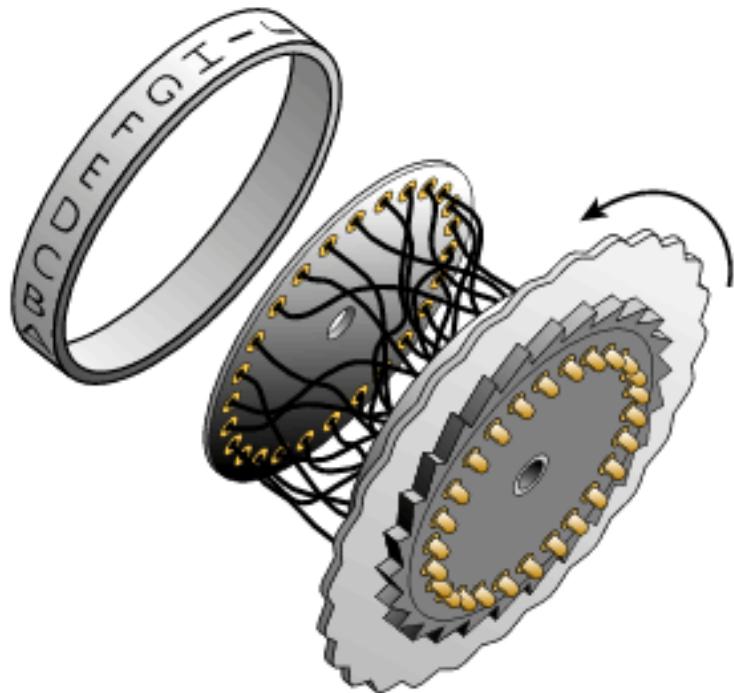


# Main components of the Enigma

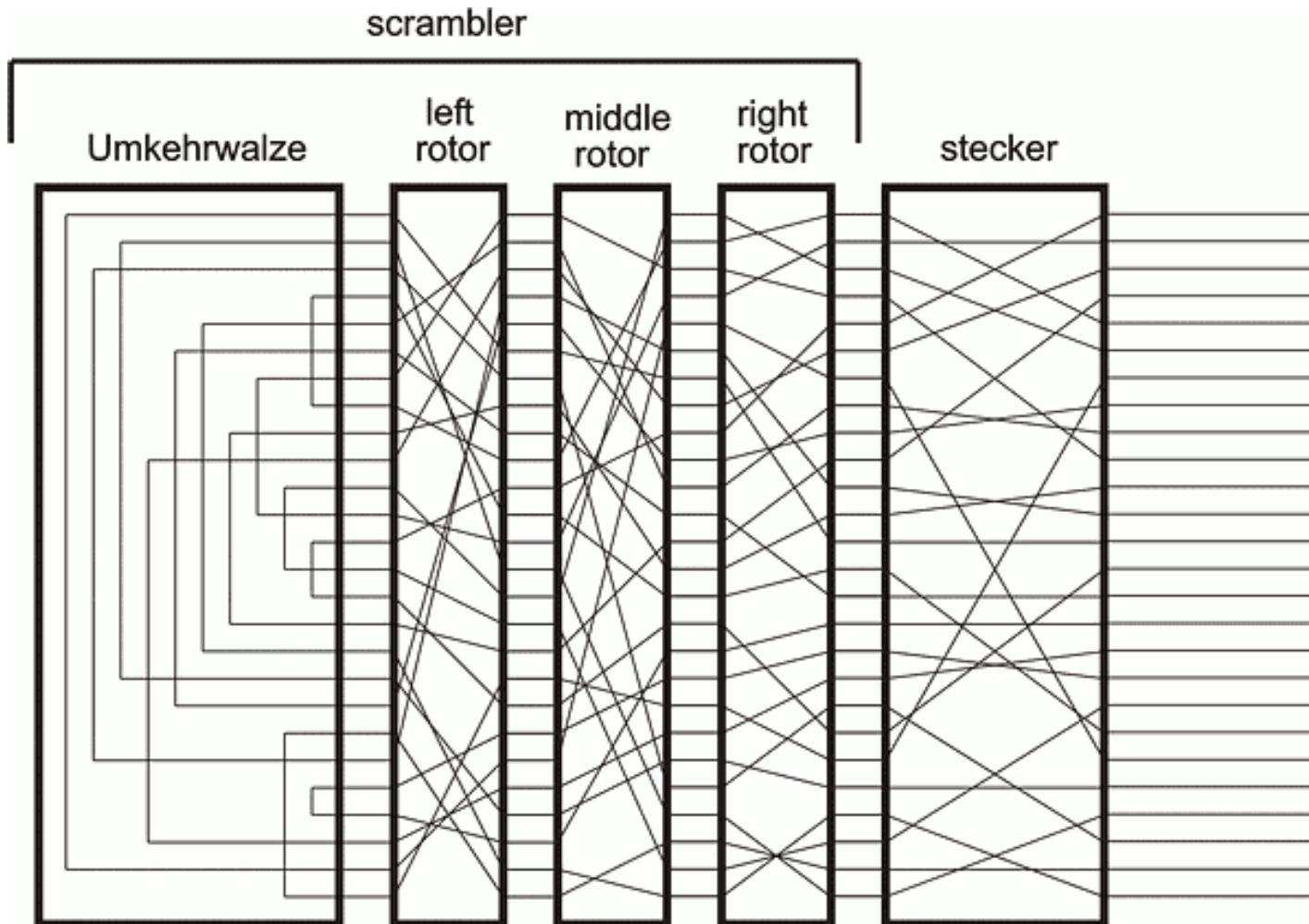
- keyboard  
for input of the plaintext / ciphertext
- lampboard  
for display of the ciphertext / plaintex
- plugboard  
for swapping some input letter pairs
- scrambler unit (including the rotors)  
producing the ciphertext from the plaintext (and vice versa)



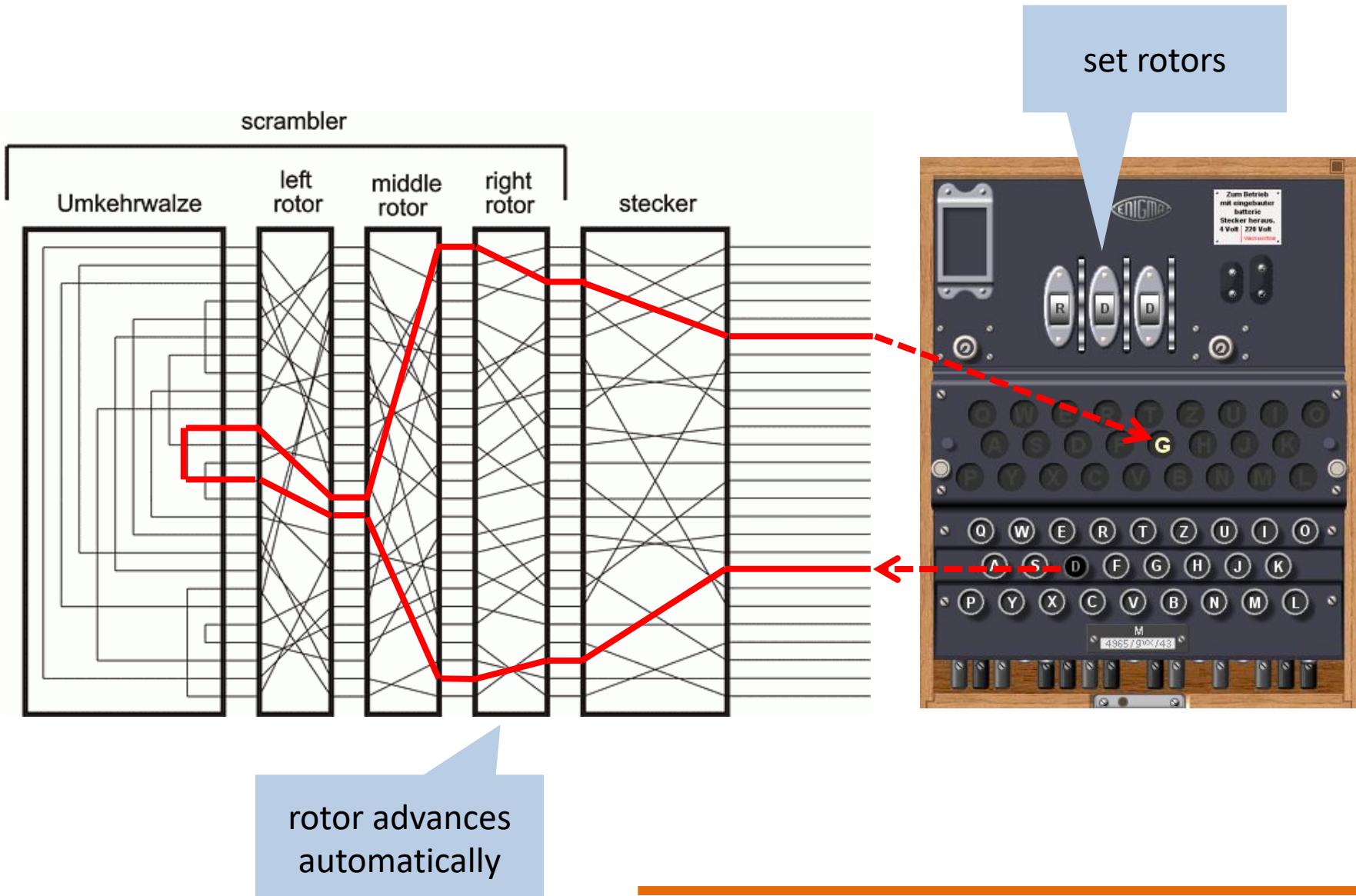
# The rotors



# The scrambler unit and the plugboard



# Enigma in action



# Let's try it out!



<http://enigma.louisedade.co.uk/enigma.html>

**ENIGMA**  
Navy M3/M4 Emulator

---

Introduction

---

**Enigma Emulator**

---

Daily Key Generator

---

Emulator Help

---

Enigma Mechanics

---

Further Reading

## Navy M3/M4 Enigma Machine Emulator

Enigma Type: M3 (Navy/Army/Airforce) ▾

Umkehrwalze: --- B --- ▾

Walzenlage: II ▾ III ▾ I ▾

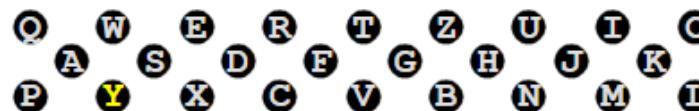
Ringstellung: S O S

Grundstellung: Q W Q

Functions: [Save Settings](#) | [Clear Settings](#) | [Help!](#)

Steckerbrett

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



Type Letter:

>> <<

Message/Cipher Text In:

ATTA CKAT DAWN

Cipher/Clear Text Out:

XJZI DEWO COKY

Group letters in blocks of: 4 ▾

Input Method:  Single letter,  Block of text

# Enigma key space

- What determines the Enigma key?
  - letter pairs swapped (e.g., A/L – P/R – T/D – B/W – K/F – O/Y)
  - order of rotors in the slots (e.g., II – III – I)
  - initial position of the rotors (e.g., R – D – D)

- What is the key space size?

$100391791500 \times 6 \times 26^3 \sim 2^{53}$

$$\frac{\binom{26}{2} \binom{24}{2} \binom{22}{2} \binom{20}{2} \binom{18}{2} \binom{16}{2}}{6!}$$

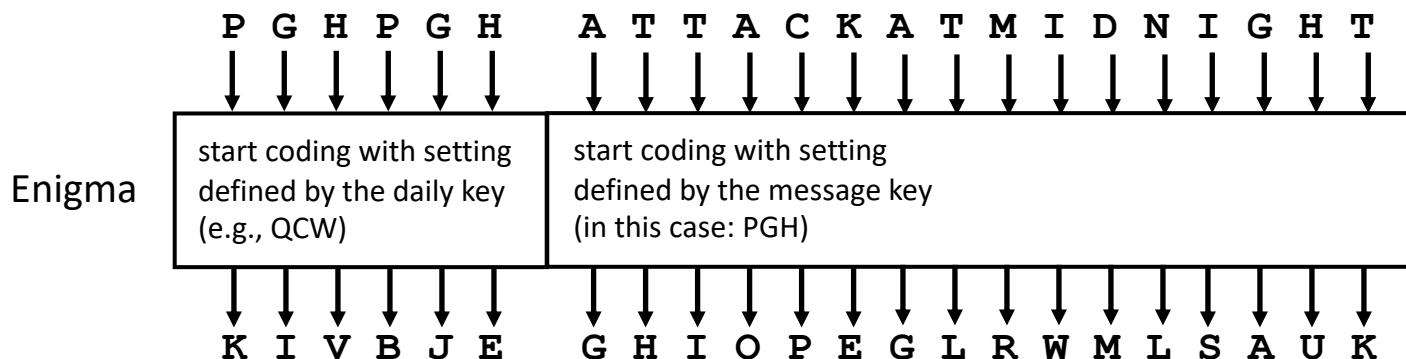
- yet, Enigma was broken by the Allies in WWII
  - exploiting protocol weaknesses and weak keys
  - code breaking was partly automated → birth of first computers
  - credit goes to Marian Rejewski and Alan Turing



# Breaking the Enigma

---

- every morning, the Germans distribute a daily key to their units
- however, they do not directly use the daily key to encrypt messages
- instead:
  - they generate a fresh message key for every message
  - they encrypt the message key with the daily key, and send this at the beginning of the communication
  - then they encrypt the message with the message key, and send it to the receiver
  - the receiver first decrypts the message key with the daily key and then decrypts the message with the message key
- in order to cope with errors during transmission, the message key is repeated twice at the beginning of the message!
- example:



# Breaking the Enigma

---

- Rejewski thought that the repetition of the message key at the beginning of the message is a weakness that may be exploited
  - a guess for the daily key can be confirmed by checking if decoding with the guessed key produces a repeating letter triplet at the beginning of the decoded message
- the Polish codebreakers built a machine that tried different guesses for the daily key in an automated way
  - the machine consisted of 6 Enigma copies (each corresponding to one of the 6 possible rotor orders)
  - the machine continuously modified the position setting of the rotors, and attempted decrypting some intercepted message, until it found the daily key
- from 1933, Poland was able to routinely break encrypted German communications

# Breaking the Enigma

---

- in December 1938, the Germans increase the security of the Enigma
  - 2 new rotors (operators have to choose 3 rotors out of 5, and the order in which they are put in the machine → this increases possible rotor placements from 6 to 60)
  - increase the number of letter pairs swapped on the plugboard from 6 to 10
  - key space grows to  $\sim 2^{66}$
- in April 1939, Hitler breaks the non-aggression treaty with Poland
- in July 1939, Poland reveals their Enigma breaking capability to England
- on August 16, 1939, the design documents of the Enigma breaking machine are transferred to London
- on September 1, 1939, Germany invades Poland

# Breaking the Enigma

---

- some weaknesses exploited by the British
  - cillies
    - » German Enigma operators sometimes used very weak (far from random) message keys (e.g., QWE, BNM)
    - » an operator always used the same message key (C.I.L.) – perhaps the initials of his wife or girl friend?
    - » these weak keys were called *cillies* (~silly)
  - Germans had usage constraints that actually weakened their system
    - » rotors had to be changed every day, and the same rotor must not be placed in the same slot on two consecutive days
    - » e.g., after I-II-V, they could not use III-II-IV
    - » this actually reduced the size of the key space that the British had to search over

# Breaking the Enigma

---

- in September 1939, Alan Turing joins the code breakers in Bletchley Park
- his task is to find a new method for breaking the cipher that does not rely on the repetition of the message key at the beginning of the coded message
- Turing invents a new method that is essentially an attack known today as the known-plaintext attack
  - German messages are well structured
  - some messages contain guessable words at guessable locations
  - e.g., every morning at 6am, they send a weather forecast, which includes the word "wetter" always at the same position within the message
- the British build new Enigma breaking machines (Victory, Agnus Dei) based on the plans of Turing in 1940
- indeed, Germans change their message key sending protocol in May 1940, but this does not affect the cryptanalytic capabilities of the British anymore

"THE BEST BRITISH FILM OF THE YEAR"



THE INDEPENDENT

"AN INSTANT CLASSIC"



EMPIRE

"A SUPERB THRILLER"



TIME OUT

THE TIMES

# THE IMITATION GAME

BENEDICT  
CUMBERBATCH

KEIRA  
KNIGHTLEY

12A  
RATED BY PEGI

BASED ON THE INCREDIBLE TRUE STORY



# Milestones of modern cryptography

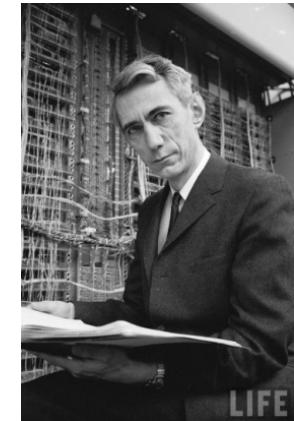
---

- Shannon's work on information theoretical characterization of encryption [1948]
- substitution-permutation ciphers and the Data Encryption Standard (DES) [1970's]
- the birth of public key cryptography [1976-78]
- quantum cryptography [1980's]
- cryptography becomes available to the general public [1990's]
- post-quantum cryptography [now!]

# The birth of modern cryptography

---

- first theoretically sound formulation of the notion of security of an encryption algorithm
  - information theory based definition of perfect secrecy
  - necessary conditions for a cipher to be perfectly secure
  - proved that the one-time pad provides perfect secrecy
  
- ideas to build strong block ciphers usable in practice
  - create a complex cipher by repeated use of otherwise simple transformations (aka. product ciphers)
    - » logical operations (e.g., XOR)
    - » small substitutions (non-linear lookup tables)
    - » bit transpositions (linear bit permutations)



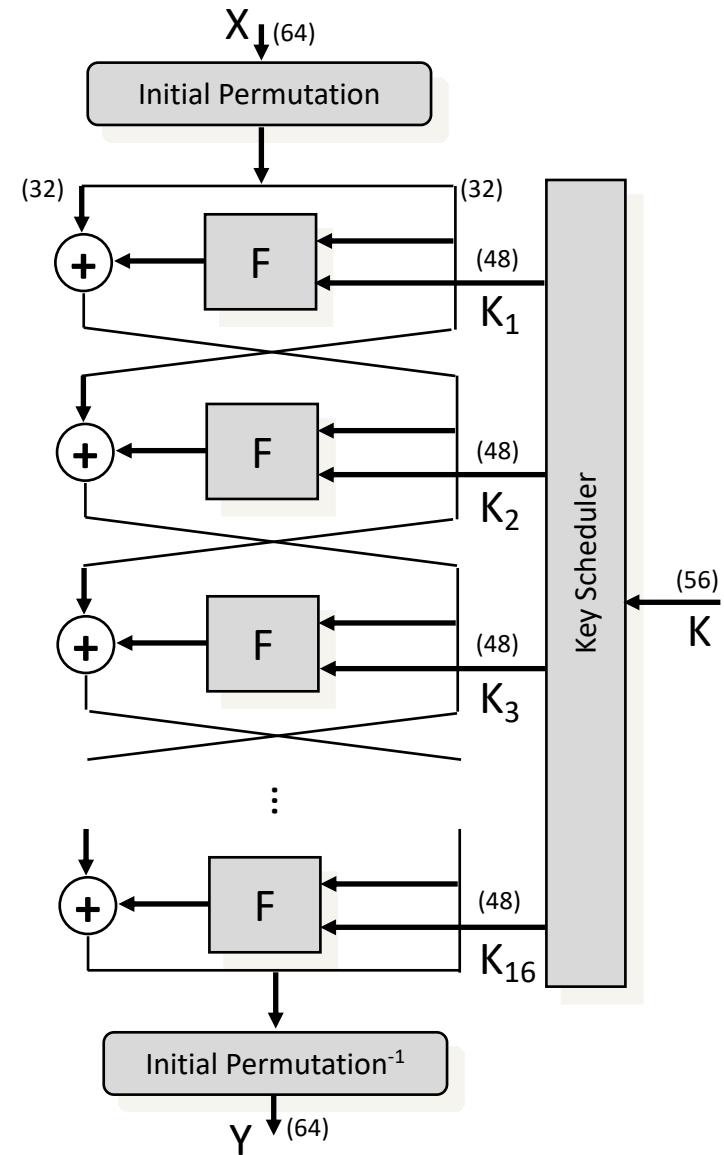
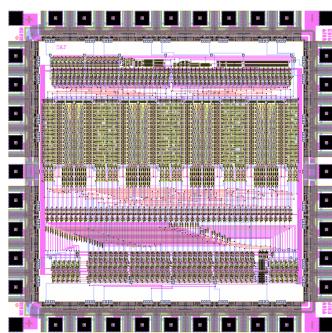
Claude E. Shannon

# Data Encryption Standard (DES)

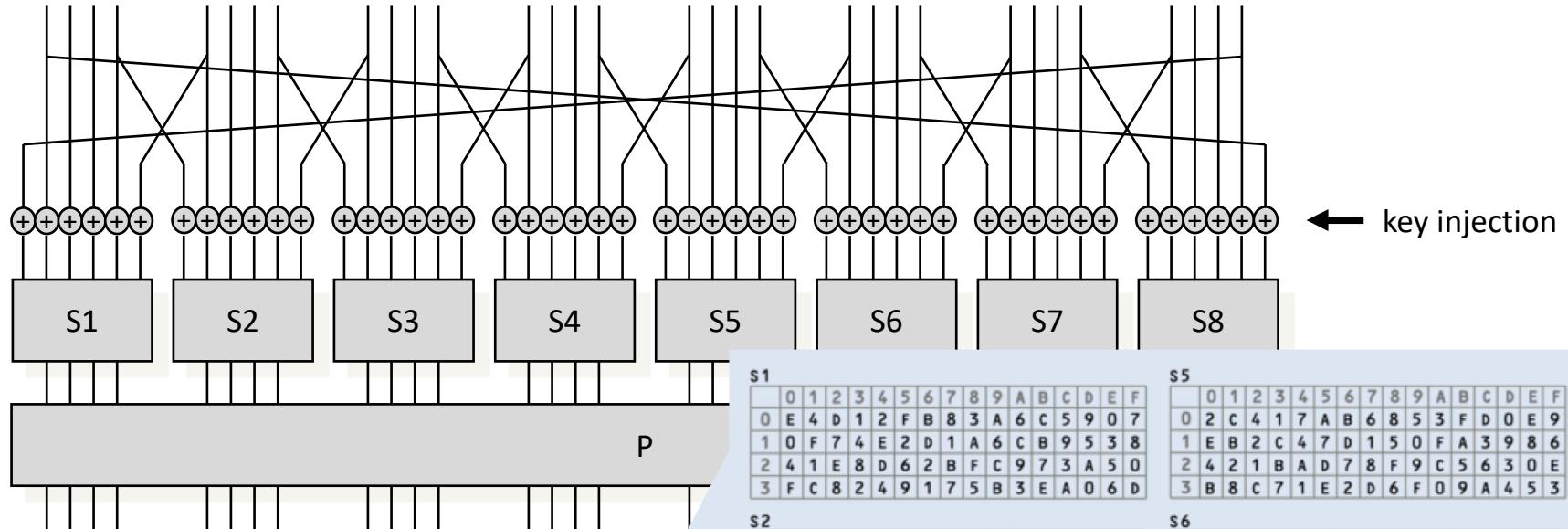
- based on Lucifer, a cipher developed by IBM in the 70's
- symmetric key **block cipher**
- features:
  - Feistel structure (same structure can be used for encoding and decoding)
  - number of rounds: 16
  - input block size: 64 bits
  - output block size: 64 bits
  - key size: 56 bits

HW implementation:

DES chip



# DES round function F



- Si – substitution box (S-box)
  - » non-linear look-up tables
- P – permutation box (P-box)
  - » linear bit permutation

S1	<table border="1"><tr><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>A</td><td>B</td><td>C</td><td>D</td><td>E</td><td>F</td></tr><tr><td>0</td><td>E</td><td>4</td><td>D</td><td>1</td><td>2</td><td>F</td><td>B</td><td>8</td><td>3</td><td>A</td><td>6</td><td>C</td><td>5</td><td>9</td><td>0</td><td>7</td></tr><tr><td>1</td><td>0</td><td>F</td><td>7</td><td>4</td><td>E</td><td>2</td><td>D</td><td>1</td><td>A</td><td>6</td><td>C</td><td>B</td><td>9</td><td>5</td><td>3</td><td>8</td></tr><tr><td>2</td><td>4</td><td>1</td><td>E</td><td>8</td><td>D</td><td>6</td><td>2</td><td>B</td><td>F</td><td>C</td><td>9</td><td>7</td><td>3</td><td>A</td><td>5</td><td>0</td></tr><tr><td>3</td><td>F</td><td>C</td><td>8</td><td>2</td><td>4</td><td>9</td><td>1</td><td>7</td><td>5</td><td>B</td><td>3</td><td>E</td><td>A</td><td>0</td><td>6</td><td>D</td></tr></table>	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	0	E	4	D	1	2	F	B	8	3	A	6	C	5	9	0	7	1	0	F	7	4	E	2	D	1	A	6	C	B	9	5	3	8	2	4	1	E	8	D	6	2	B	F	C	9	7	3	A	5	0	3	F	C	8	2	4	9	1	7	5	B	3	E	A	0	6	D	S5	<table border="1"><tr><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>A</td><td>B</td><td>C</td><td>D</td><td>E</td><td>F</td></tr><tr><td>0</td><td>2</td><td>C</td><td>4</td><td>1</td><td>7</td><td>A</td><td>B</td><td>6</td><td>8</td><td>5</td><td>3</td><td>F</td><td>D</td><td>0</td><td>E</td><td>9</td></tr><tr><td>1</td><td>E</td><td>B</td><td>2</td><td>C</td><td>4</td><td>7</td><td>D</td><td>1</td><td>5</td><td>0</td><td>F</td><td>A</td><td>3</td><td>9</td><td>8</td><td>6</td></tr><tr><td>2</td><td>4</td><td>2</td><td>1</td><td>B</td><td>A</td><td>D</td><td>7</td><td>8</td><td>F</td><td>9</td><td>C</td><td>5</td><td>6</td><td>3</td><td>0</td><td>E</td></tr><tr><td>3</td><td>B</td><td>8</td><td>C</td><td>7</td><td>1</td><td>E</td><td>2</td><td>D</td><td>6</td><td>F</td><td>0</td><td>9</td><td>A</td><td>4</td><td>5</td><td>3</td></tr></table>	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	0	2	C	4	1	7	A	B	6	8	5	3	F	D	0	E	9	1	E	B	2	C	4	7	D	1	5	0	F	A	3	9	8	6	2	4	2	1	B	A	D	7	8	F	9	C	5	6	3	0	E	3	B	8	C	7	1	E	2	D	6	F	0	9	A	4	5	3	S6	<table border="1"><tr><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>A</td><td>B</td><td>C</td><td>D</td><td>E</td><td>F</td></tr><tr><td>0</td><td>C</td><td>1</td><td>A</td><td>F</td><td>9</td><td>2</td><td>6</td><td>8</td><td>0</td><td>D</td><td>3</td><td>4</td><td>E</td><td>7</td><td>5</td><td>B</td></tr><tr><td>1</td><td>A</td><td>F</td><td>4</td><td>2</td><td>7</td><td>C</td><td>9</td><td>5</td><td>6</td><td>1</td><td>D</td><td>E</td><td>0</td><td>B</td><td>3</td><td>8</td></tr><tr><td>2</td><td>9</td><td>E</td><td>F</td><td>5</td><td>2</td><td>8</td><td>C</td><td>3</td><td>7</td><td>0</td><td>4</td><td>A</td><td>1</td><td>D</td><td>B</td><td>6</td></tr><tr><td>3</td><td>4</td><td>3</td><td>2</td><td>C</td><td>9</td><td>5</td><td>F</td><td>A</td><td>B</td><td>E</td><td>1</td><td>7</td><td>6</td><td>0</td><td>8</td><td>D</td></tr></table>	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	0	C	1	A	F	9	2	6	8	0	D	3	4	E	7	5	B	1	A	F	4	2	7	C	9	5	6	1	D	E	0	B	3	8	2	9	E	F	5	2	8	C	3	7	0	4	A	1	D	B	6	3	4	3	2	C	9	5	F	A	B	E	1	7	6	0	8	D	S7	<table border="1"><tr><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>A</td><td>B</td><td>C</td><td>D</td><td>E</td><td>F</td></tr><tr><td>0</td><td>4</td><td>B</td><td>2</td><td>E</td><td>F</td><td>0</td><td>8</td><td>D</td><td>3</td><td>C</td><td>9</td><td>7</td><td>5</td><td>A</td><td>6</td><td>1</td></tr><tr><td>1</td><td>D</td><td>0</td><td>B</td><td>7</td><td>4</td><td>9</td><td>1</td><td>A</td><td>E</td><td>3</td><td>5</td><td>C</td><td>2</td><td>F</td><td>8</td><td>6</td></tr><tr><td>2</td><td>1</td><td>4</td><td>B</td><td>D</td><td>C</td><td>3</td><td>7</td><td>E</td><td>A</td><td>F</td><td>6</td><td>8</td><td>0</td><td>5</td><td>9</td><td>2</td></tr><tr><td>3</td><td>6</td><td>B</td><td>D</td><td>8</td><td>1</td><td>4</td><td>A</td><td>7</td><td>9</td><td>5</td><td>0</td><td>F</td><td>E</td><td>2</td><td>3</td><td>C</td></tr></table>	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	0	4	B	2	E	F	0	8	D	3	C	9	7	5	A	6	1	1	D	0	B	7	4	9	1	A	E	3	5	C	2	F	8	6	2	1	4	B	D	C	3	7	E	A	F	6	8	0	5	9	2	3	6	B	D	8	1	4	A	7	9	5	0	F	E	2	3	C	S8	<table border="1"><tr><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>A</td><td>B</td><td>C</td><td>D</td><td>E</td><td>F</td></tr><tr><td>0</td><td>D</td><td>2</td><td>8</td><td>4</td><td>6</td><td>F</td><td>B</td><td>1</td><td>A</td><td>9</td><td>3</td><td>E</td><td>5</td><td>0</td><td>C</td><td>7</td></tr><tr><td>1</td><td>1</td><td>F</td><td>D</td><td>8</td><td>A</td><td>3</td><td>7</td><td>4</td><td>C</td><td>5</td><td>6</td><td>B</td><td>0</td><td>E</td><td>9</td><td>2</td></tr><tr><td>2</td><td>7</td><td>B</td><td>4</td><td>1</td><td>9</td><td>C</td><td>E</td><td>2</td><td>0</td><td>6</td><td>A</td><td>D</td><td>F</td><td>3</td><td>5</td><td>8</td></tr><tr><td>3</td><td>2</td><td>1</td><td>E</td><td>7</td><td>4</td><td>A</td><td>8</td><td>D</td><td>F</td><td>C</td><td>9</td><td>0</td><td>3</td><td>5</td><td>6</td><td>B</td></tr></table>	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	0	D	2	8	4	6	F	B	1	A	9	3	E	5	0	C	7	1	1	F	D	8	A	3	7	4	C	5	6	B	0	E	9	2	2	7	B	4	1	9	C	E	2	0	6	A	D	F	3	5	8	3	2	1	E	7	4	A	8	D	F	C	9	0	3	5	6	B
0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F																																																																																																																																																																																																																																																																																																																																																																																																																														
0	E	4	D	1	2	F	B	8	3	A	6	C	5	9	0	7																																																																																																																																																																																																																																																																																																																																																																																																																													
1	0	F	7	4	E	2	D	1	A	6	C	B	9	5	3	8																																																																																																																																																																																																																																																																																																																																																																																																																													
2	4	1	E	8	D	6	2	B	F	C	9	7	3	A	5	0																																																																																																																																																																																																																																																																																																																																																																																																																													
3	F	C	8	2	4	9	1	7	5	B	3	E	A	0	6	D																																																																																																																																																																																																																																																																																																																																																																																																																													
0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F																																																																																																																																																																																																																																																																																																																																																																																																																														
0	2	C	4	1	7	A	B	6	8	5	3	F	D	0	E	9																																																																																																																																																																																																																																																																																																																																																																																																																													
1	E	B	2	C	4	7	D	1	5	0	F	A	3	9	8	6																																																																																																																																																																																																																																																																																																																																																																																																																													
2	4	2	1	B	A	D	7	8	F	9	C	5	6	3	0	E																																																																																																																																																																																																																																																																																																																																																																																																																													
3	B	8	C	7	1	E	2	D	6	F	0	9	A	4	5	3																																																																																																																																																																																																																																																																																																																																																																																																																													
0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F																																																																																																																																																																																																																																																																																																																																																																																																																														
0	C	1	A	F	9	2	6	8	0	D	3	4	E	7	5	B																																																																																																																																																																																																																																																																																																																																																																																																																													
1	A	F	4	2	7	C	9	5	6	1	D	E	0	B	3	8																																																																																																																																																																																																																																																																																																																																																																																																																													
2	9	E	F	5	2	8	C	3	7	0	4	A	1	D	B	6																																																																																																																																																																																																																																																																																																																																																																																																																													
3	4	3	2	C	9	5	F	A	B	E	1	7	6	0	8	D																																																																																																																																																																																																																																																																																																																																																																																																																													
0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F																																																																																																																																																																																																																																																																																																																																																																																																																														
0	4	B	2	E	F	0	8	D	3	C	9	7	5	A	6	1																																																																																																																																																																																																																																																																																																																																																																																																																													
1	D	0	B	7	4	9	1	A	E	3	5	C	2	F	8	6																																																																																																																																																																																																																																																																																																																																																																																																																													
2	1	4	B	D	C	3	7	E	A	F	6	8	0	5	9	2																																																																																																																																																																																																																																																																																																																																																																																																																													
3	6	B	D	8	1	4	A	7	9	5	0	F	E	2	3	C																																																																																																																																																																																																																																																																																																																																																																																																																													
0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F																																																																																																																																																																																																																																																																																																																																																																																																																														
0	D	2	8	4	6	F	B	1	A	9	3	E	5	0	C	7																																																																																																																																																																																																																																																																																																																																																																																																																													
1	1	F	D	8	A	3	7	4	C	5	6	B	0	E	9	2																																																																																																																																																																																																																																																																																																																																																																																																																													
2	7	B	4	1	9	C	E	2	0	6	A	D	F	3	5	8																																																																																																																																																																																																																																																																																																																																																																																																																													
3	2	1	E	7	4	A	8	D	F	C	9	0	3	5	6	B																																																																																																																																																																																																																																																																																																																																																																																																																													

# Security of DES

---

- average complexity of a brute force attack is  $2^{55}$ 
  - was suspected breakable by NSA back in the 70's
  - definitely became breakable by the late 90's by distributed computing
  - new standard AES was accepted in 2001
- algebraic attacks were discovered in the late 80's and early 90's
  - best known attacks:
    - » linear cryptanalysis (LC)
      - requires  $\sim 2^{43}$  known plaintext – ciphertext pairs
    - » differential cryptanalysis (DC)
      - requires  $\sim 2^{47}$  chosen plaintexts (and corresponding ciphertexts)
  - it was revealed in the late 90's that the designers of DES had known about DC, and optimized the DES S-boxes such that DES provides maximum resistance against DC

# A breakthrough in modern cryptography

---

Whitfield Diffie and Martin Hellman:  
New Directions in Cryptography  
*IEEE Transactions on Information Theory, 1976*



Martin Hellman and Whitfield Diffie

# The key exchange problem

---

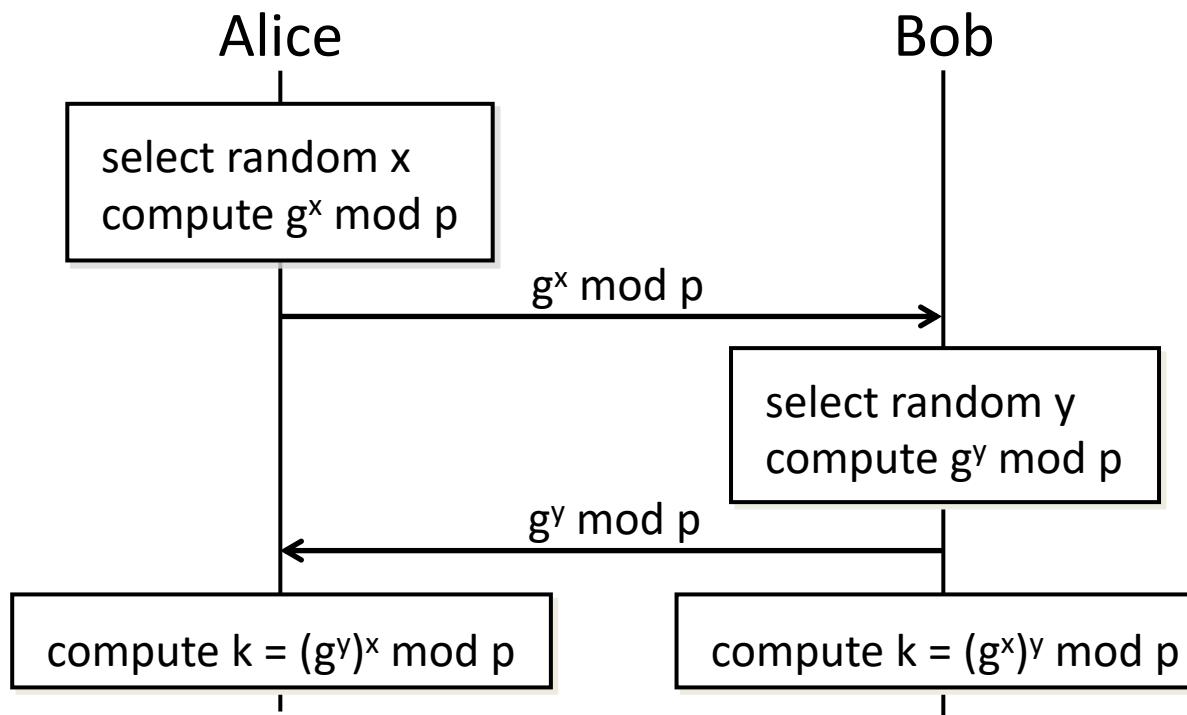
- by the 70's digital computers and telecommunication networks were increasingly used in the financial sector
- banks could use symmetric key ciphers, such as Lucifer and later DES, to encrypt sensitive data
- but they faced a practical question: how to setup a shared DES key between two end points (e.g., two remote branches of the same bank) ???
  - in case of earlier military and diplomatic applications, keys were transferred by agents in a physically secure way
  - this was expensive and inflexible for banks

# The Diffie-Hellman key exchange protocol

---

public parameters:

a large prime  $p$  and a generator element  $g$  of  $\mathbb{Z}_p^* = \{1, 2, \dots, p-1\}$



# The Diffie-Hellman key exchange protocol

---

- if an attacker can only eavesdrop the communications between Alice and Bob, then he has only  $g^x \bmod p$  and  $g^y \bmod p$
- to compute  $g^{xy} \bmod p$ , he would need  $x$  or  $y$
- it is hard to compute  $x$  from  $g^x \bmod p$ 
  - this is the so called "discrete logarithm" problem
  - no polynomial time algorithm is known to solve it
  - if  $p$  is large, then computing discrete logarithm ( $\bmod p$ ) is practically infeasible
- there seem to exist one way functions:
  - given  $x$ , it is easy to compute  $f(x)$
  - given  $y$ , it is hard to find an  $x$  for which  $y = f(x)$
- can we use such functions to realize a sort of asymmetric key cryptography ???

# The idea of asymmetric key cryptography

---

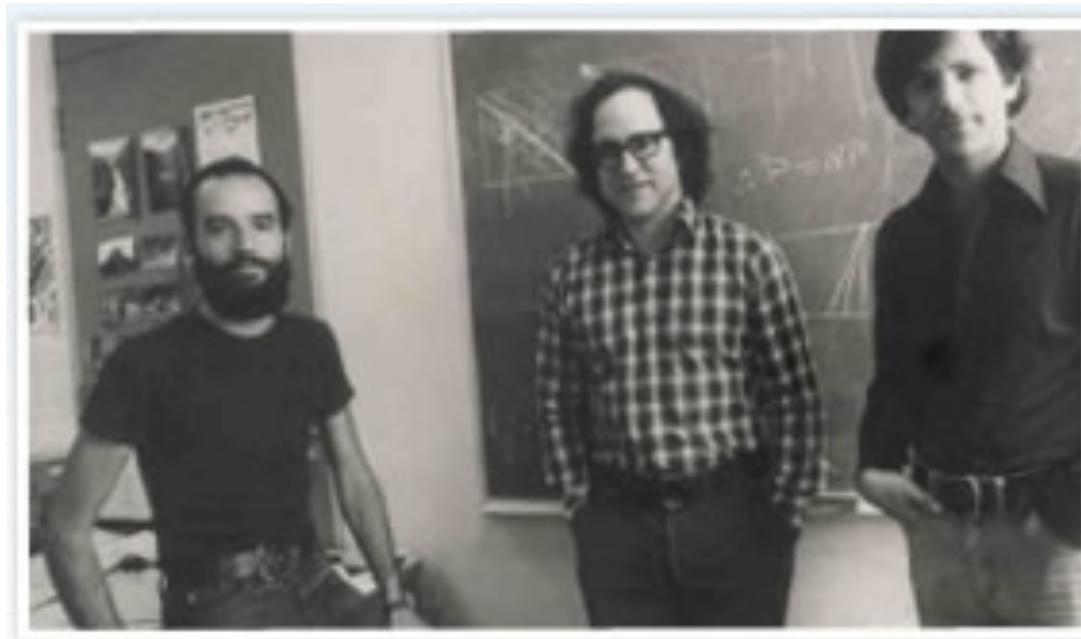
- encoding and decoding keys are not the same (unlike in symmetric key cryptography)
- computing the decoding key from the encoding key is hard (infeasible in practice)
- encoding key can be made public, decoding key should be kept secret
  - anybody can obtain the public encoding key of Alice, and send an encrypted message to her
  - only Alice can decrypt the message with the private decoding key
  - an attacker cannot compute the private key from the public key
  - aka. public key cryptography
  - solves the key exchange problem (but has other issues to solve)



# The RSA cryptosystem

---

Ronald Rivest, Adi Shamir, Leonard Adleman:  
A Method for Obtaining Digital Signatures and Public-Key Cryptosystems  
*Communications of the ACM*, 1978.



Adi Shamir, Ronald Rivest, and Leonard Adleman

# The RSA cryptosystem

---

- key-pair generation algorithm:
  - choose two large primes  $p$  and  $q$  (easy)
  - $n = pq$ ,  $\phi(n) = (p-1)(q-1)$  (easy)
  - choose  $e$ , such that  $1 < e < \phi(n)$  and  $\gcd(e, \phi(n)) = 1$  (easy)
  - compute the inverse  $d$  of  $e$  mod  $\phi(n)$ , i.e.,  $d$  such that  $ed \bmod \phi(n) = 1$  (easy if  $p$  and  $q$  are known)
  - output **public key:  $(e, n)$**  (public exponent and modulus)
  - output **private key:  $d$**  (private exponent)
- encryption algorithm:
  - represent the plaintext message as an integer  $m \in [0, n-1]$
  - compute the ciphertext  $c = m^e \bmod n$
- decryption algorithm:
  - compute the plaintext from the ciphertext  $c$  as  $m = c^d \bmod n$

# Security of asymmetric key algorithms

---

- security is typically related to the difficulty of solving some hard mathematical problem
  - e.g., factoring or discrete logarithm
- provable security by reduction proofs:
  - we show that any efficient algorithm that breaks our crypto scheme could be used to efficiently solve a believed to be hard mathematical problem
  - this means that breaking our crypto scheme is at least as hard as solving the hard mathematical problem
- there exist provably secure crypto systems, but most of them are not efficient (fast) enough for practical applications
- most of the public key crypto schemes that we use in practice are not provably secure (or only partial proofs exist)

# Example: Security of the RSA crypto system

---

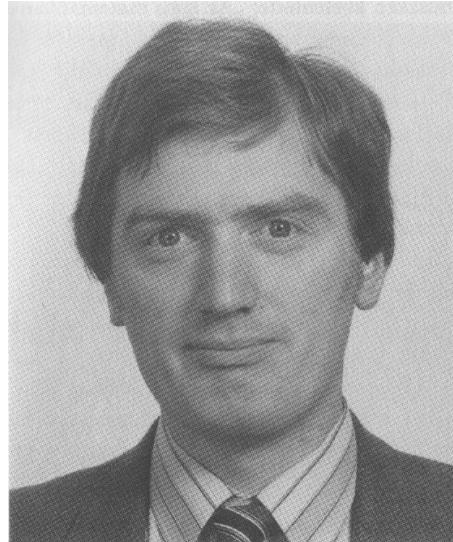
- factoring integers is believed to be a hard problem
  - given a composite integer  $n$ , find its prime factors
  - true complexity is unknown
  - it is believed that no polynomial time algorithm exists to solve it
- computing  $d$  from  $(e, n)$  is equivalent to factoring  $n$
- computing  $m$  from  $c$  and  $(e, n)$  may not be equivalent to factoring  $n$  (this is known as the RSA problem)
  - if the factors  $p$  and  $q$  of  $n$  are known, then one can easily compute  $d$ , and using  $d$ , one can also compute  $m$  from  $c$
  - we don't know if one could factor  $n$ , given that he can efficiently compute  $m$  from  $c$  and  $(e, n)$

# The secret story of public key cryptography

---



James Ellis



Clifford Cocks



Malcolm Williamson

# The secret story of public key cryptography

---

- Ellis, Cocks, and Williamson worked for GCHQ (British security agency)
- in 1969, Ellis defined the general model of asymmetric key cryptography (called it non-secret key coding)
- in 1973, Cocks invented a cryptosystem same as RSA
  - he was introduced to the idea of non-secret key crypto
  - he worked in the field of number theory, and immediately thought of using factoring as an underlying hard problem
- in 1974, Williamson (a friend of Cocks) invented a key exchange protocol same as the Diffie-Hellman protocol
- by 1975, Ellis, Cocks, and Williamson worked out all the major results of public key cryptography, which were (re)invented some years later
- the story was made pulic only in 1997

# Pretty Good Privacy (PGP)

---

- Phil Zimmermann
  - a peace activist in the 1980s during the Nuclear Weapons Freeze campaign
  - saw the need to develop what would later become PGP
    - » for protecting human rights overseas
    - » for protecting grassroots political organizations in the US
- US Senate Bill 266 of 1991
  - Congressional discussion on requiring that all communications equipment and services have a “backdoor” in them to permit government anti-criminal and counterterrorism activities
  - familiar?
- first working version of PGP arrived in 1991 (when it was still legal)
  - free software that used strong encryption (e.g., RSA)
  - strong crypto available to ordinary people for the first time in history
  - new opportunities for human rights organizations and other users concerned with privacy



# Patent and export problems with PGP

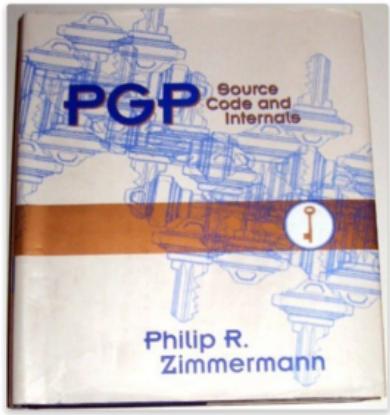
---

- the RSA algorithm was patented in the US by MIT, and was licensed to RSA Data Security Inc.
  - years of disagreement about the permission to use RSA in PGP
  - finally, RSADSI created the RSAREF library for use in freeware and shareware, and PGP 2.5 used RSAREF (in the US)
  - an “international” version of PGP, developed completely outside of the US, used the original implementation of the RSA algorithm
- Public Key Partners filed a complaint in 1992 with US Customs, complaining that Zimmermann was exporting cryptography without the appropriate licenses
  - until 1997, international regulation considered cryptography a weapon
  - free and open cryptosystems were regulated as munitions in the US
  - a criminal investigation of Zimmermann was started in 1992
  - printed books were and are exempt from the export controls
  - the investigation of Zimmermann was dropped in 1996
  - export controls on cryptography were radically liberalized in 2000

# PGP and the crypto war

---

- publication of *PGP Source Code and Internals* (MIT Press, 1995)



**PGP: Source Code and Internals** Hardcover – June 9, 1995

by Philip R. Zimmermann (Author)

★★★★★ 1 customer review

▶ See all formats and editions

Hardcover  
from \$285.00

9 Used from \$285.00

3 New from \$1,008.50

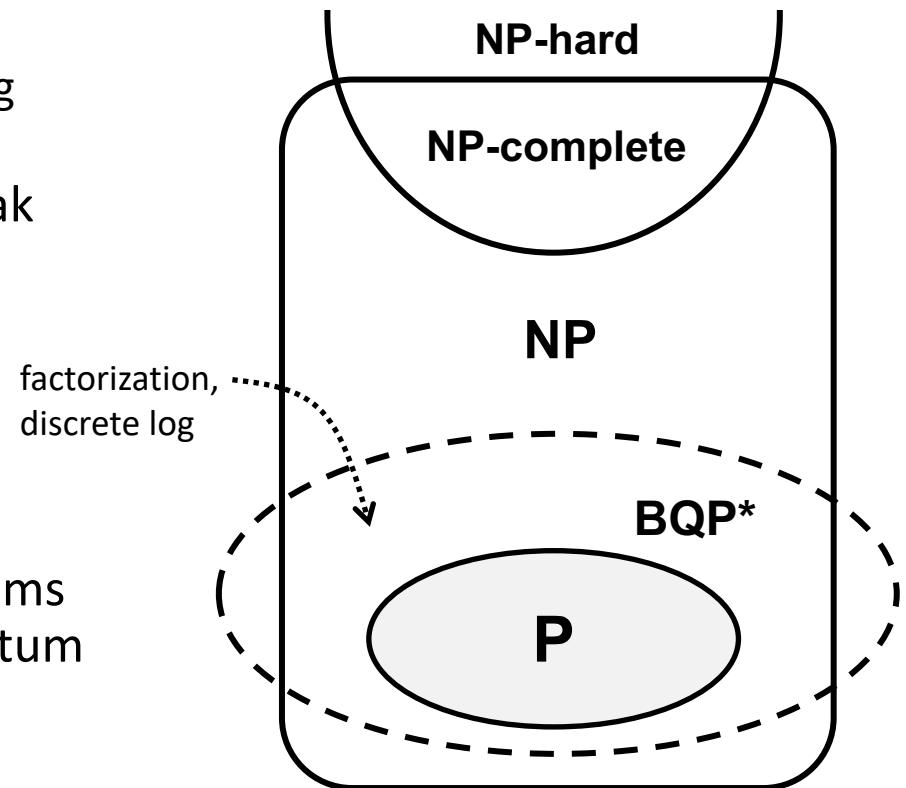
**amazonstudent** **FREE TWO-DAY SHIPPING**  
FOR COLLEGE STUDENTS [Learn more](#)

- later, Pretty Good Privacy Inc. published the source code of PGP in a more sophisticated set of books
  - also included specialized software tools optimized for easy optical character recognition (OCR) scanning of C source code
  - this made it easy to export unlimited quantities of cryptographic source code, rendering the export controls moot

# Quantum and post-quantum crypto

---

- quantum cryptography  
(started in the 1980's)
  - using quantum effects to solve traditional problems in new ways
    - » e.g., quantum key exchange using polarized photons
  - using quantum computers to break modern ciphers efficiently
    - » e.g., the Shor factorization algorithm to break RSA
- post-quantum cryptography
  - developing cryptographic algorithms that resist attacks even by a quantum computer
    - » see <http://pqcrypto.org/>



\* BQP – Bounded error Quantum Polynomial time

# Practical applications of cryptography

---

- today, national and international laws, regulations, and expectations about privacy, data governance, and corporate governance either imply or require the widespread use of strong cryptography
  - secure communication over public channels / networks
    - » WWW (https / TLS)
    - » GSM/.../4G
    - » WiFi (WPA, WPA2)
    - » Bluetooth
  - secure data storage
    - » disk encryption (TrueCrypt, BitLocker, ...)
    - » encrypted cloud storage (Tresorit, CipherCloud, ...)
  - authentication
    - » smart cards (e.g., bank cards)
    - » ignition keys of cars
    - » electronic tickets in public transport (automated fare collection systems)
  - software authentication and integrity protection
    - » digitally signed code (e.g., drivers, applets, Android packages)
  - ...

# Further readings

---

