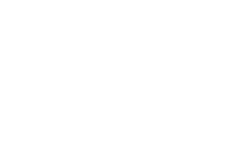
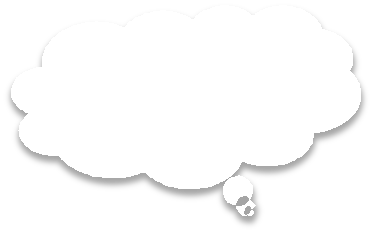
Introduction



What is cryptography?

Crypto core



Talking

to Bob

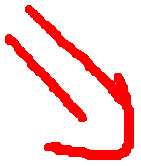
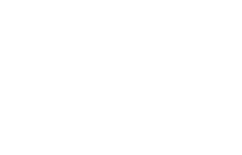
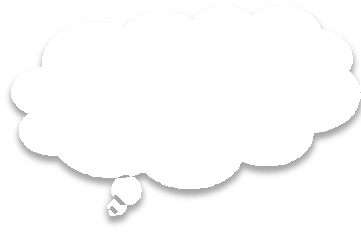
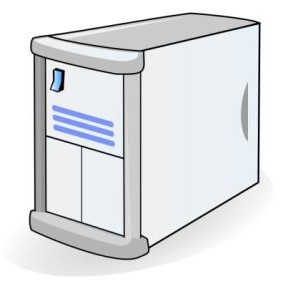
Talking

to Alice

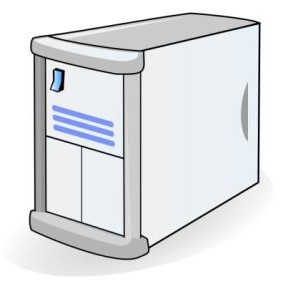
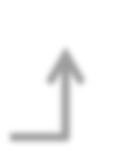
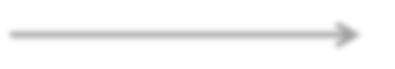
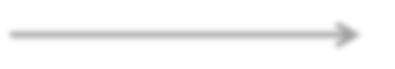
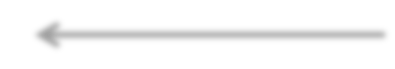
Secret key establishment:

Alice

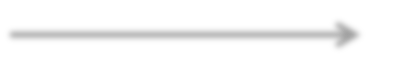
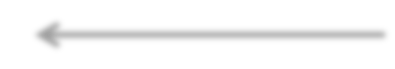
Bob



attacker???



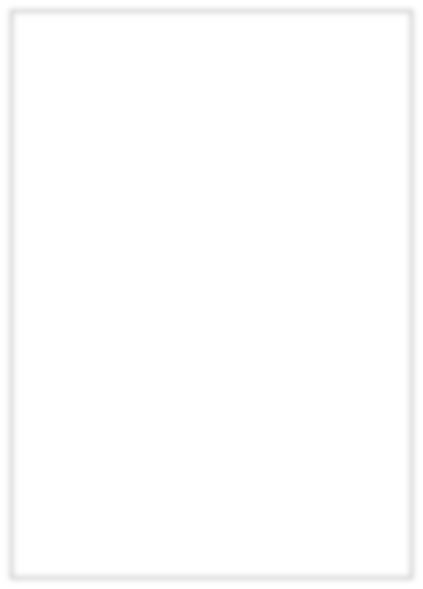
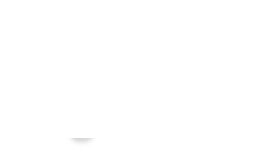
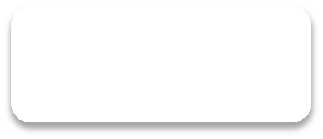
Secure communication: k m1 k



m2

confidentiality and integrity

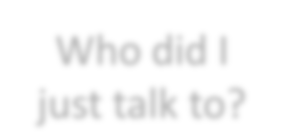
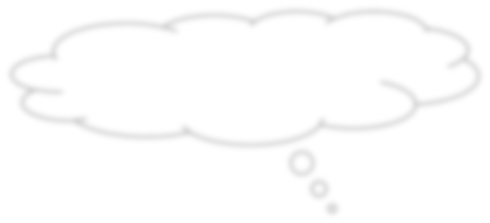
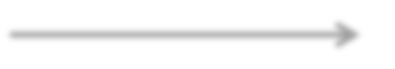
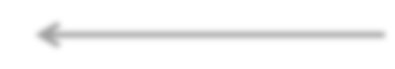
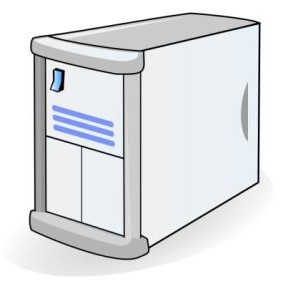
But crypto can do much more



• Digital signatures

• Anonymous communication

Who did I



just talk to?

Alice



Bob

Alice

signature

But crypto can do much more

• Digital signatures

• Anonymous communication

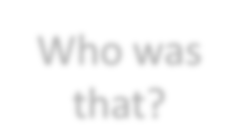
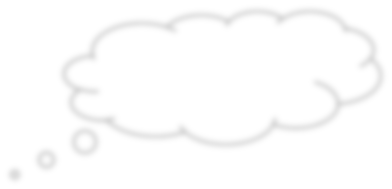
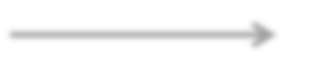
• Anonymous **digital** cash

– Can I spend a “digital coin” without anyone knowing who I am?

– How to prevent double spending?

**1$** Alice

Internet

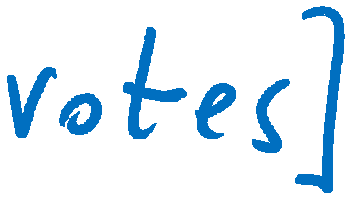
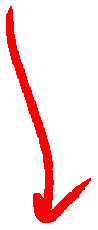
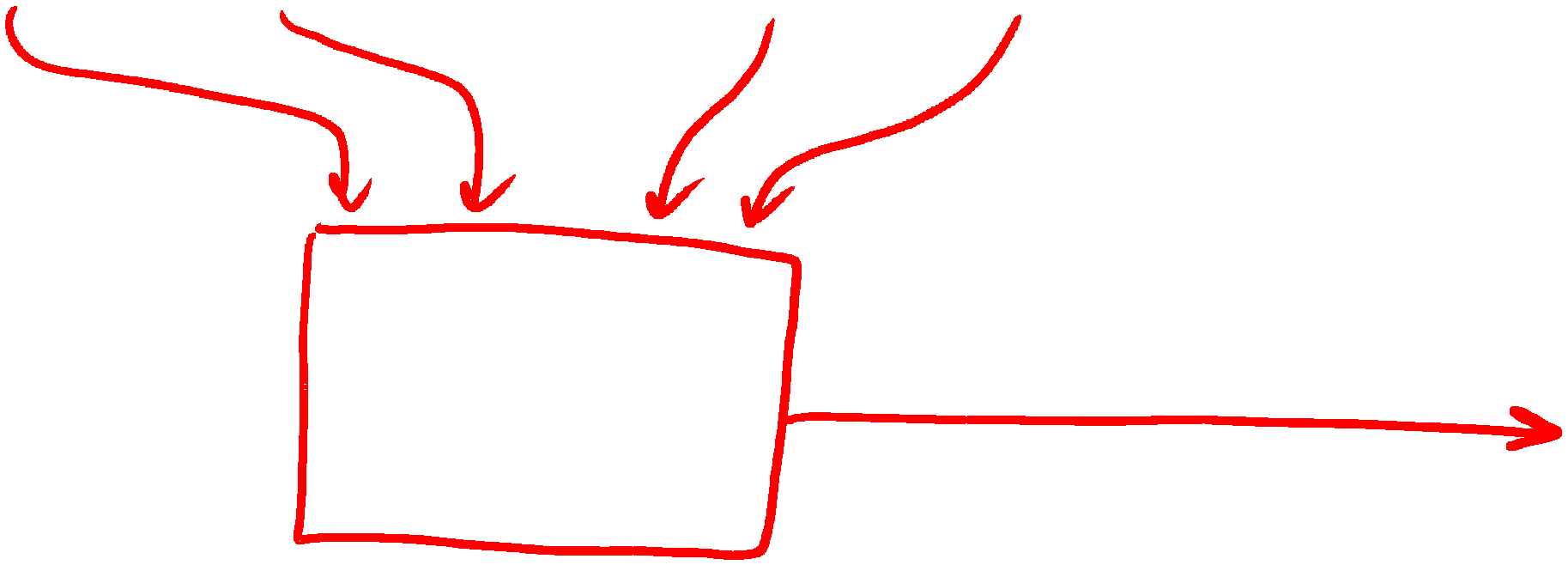
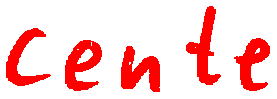


(anon. comm.)

Who was

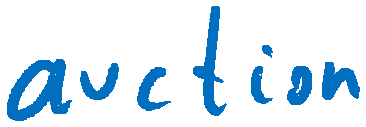
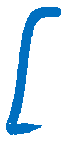
that?

Protocols



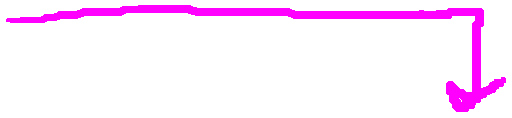
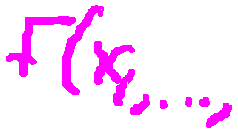
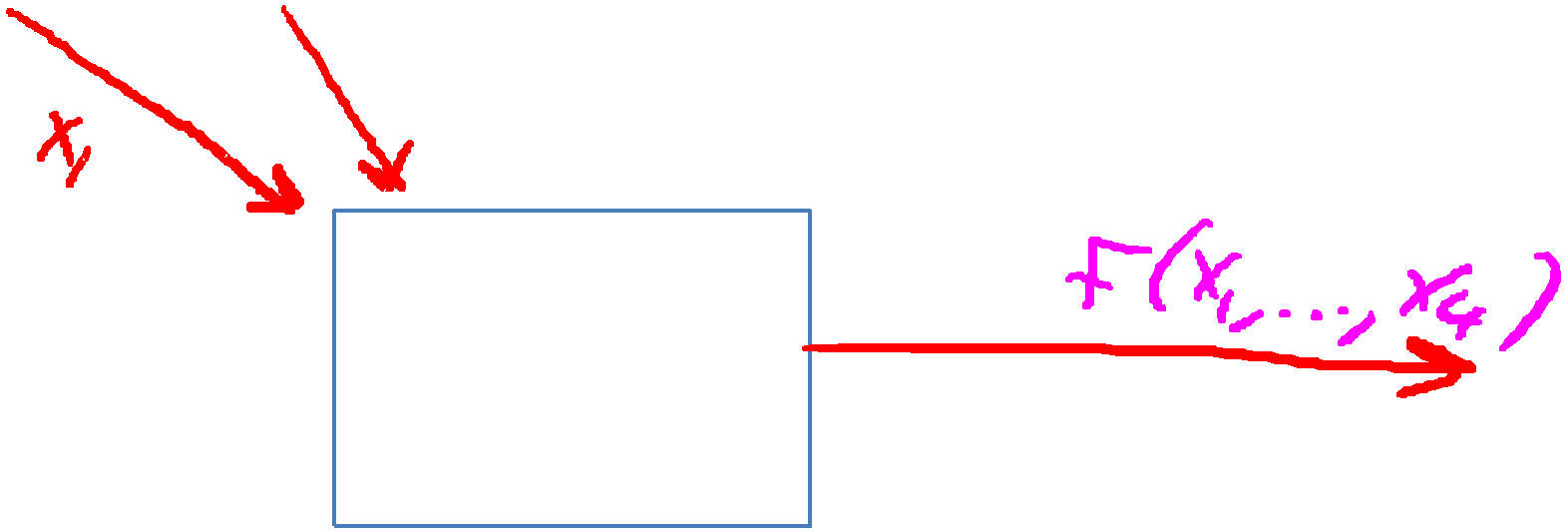
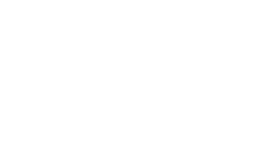
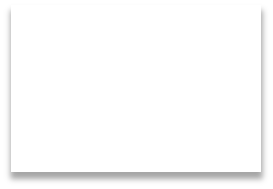
• Elections

• Private auctions



Protocols

• Elections



• Private auctions

Goal: compute f(x1, x2, x3, x4)



trusted

authority

“Thm:” anything that can done with trusted auth. can also

be done without

• Secure multi-party computation

Crypto magic

• Privately outsourcing computation

search

What did she

search for?

query

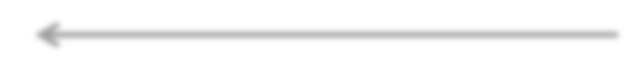
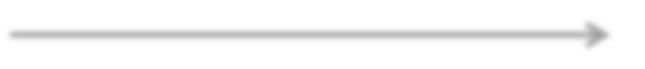
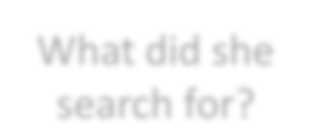
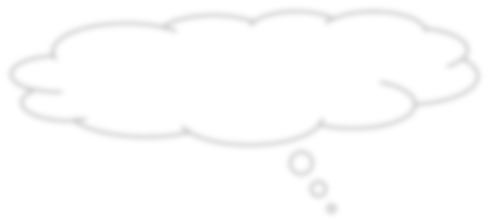
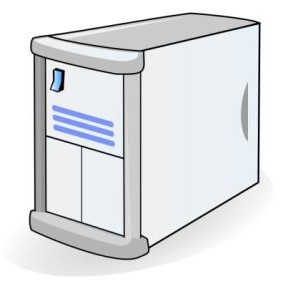
results

Alice



E[ query ]

E[ results ]



• Zero knowledge (proof of knowledge)

???

!

N=p∙q

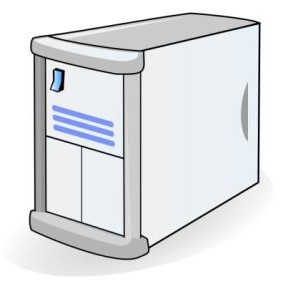
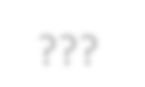
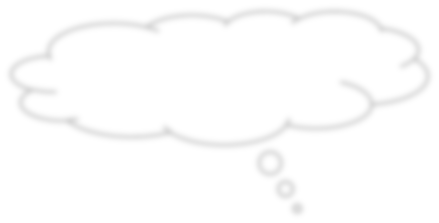
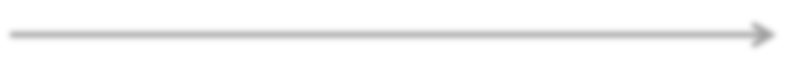
Alice

I know the factors of N !

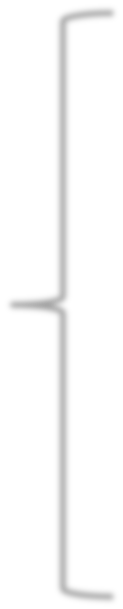
proof π

N

Bob



A rigorous science



The three steps in cryptography:

• Precisely specify threat model

• Propose a construction

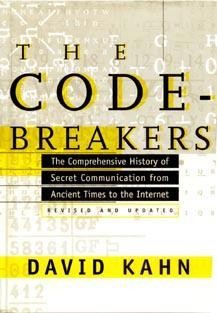
• Prove that breaking construction under

threat mode will solve an underlying hard problem

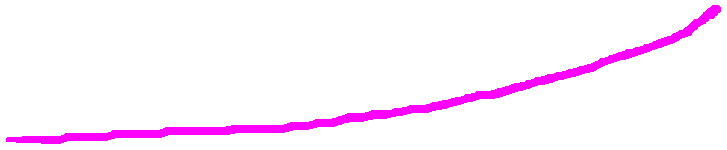
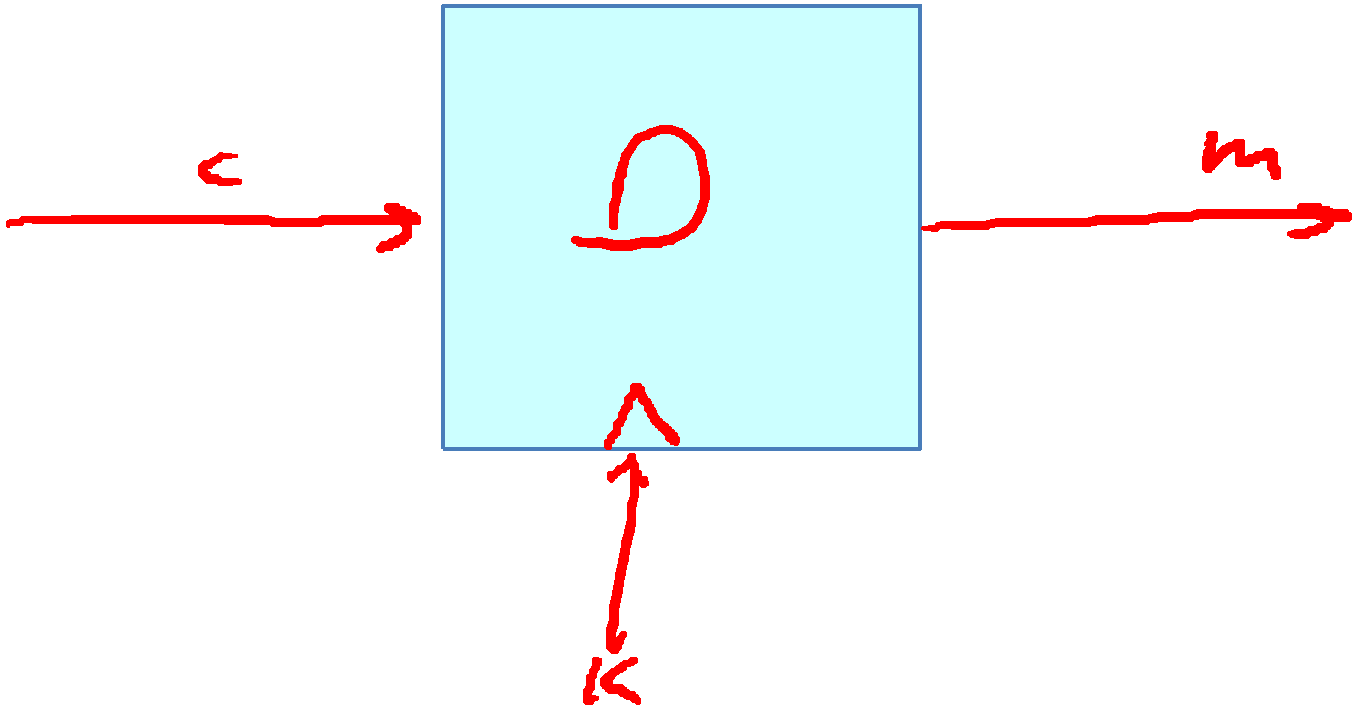
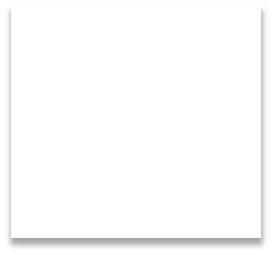
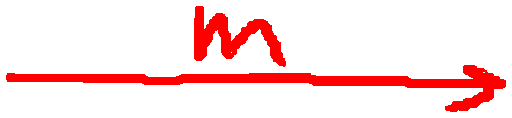
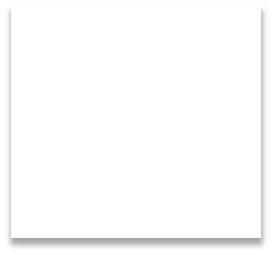
Dan Boneh

History

David Kahn, “The code breakers” (1996)

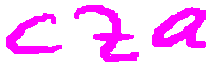
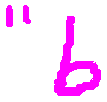
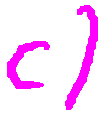
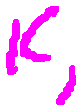
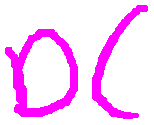
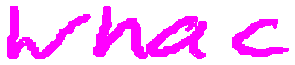
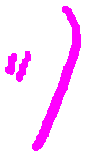
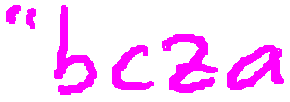


Symmetric Ciphers

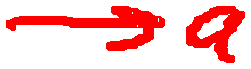
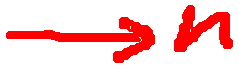
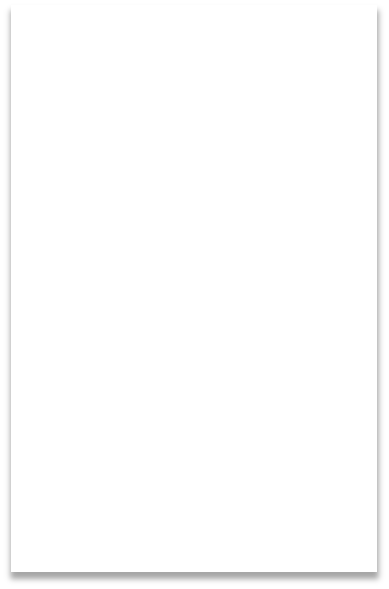


Few Historic Examples (all badly broken)

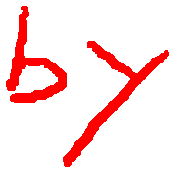
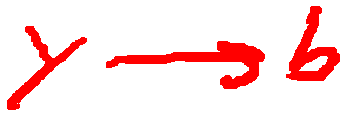
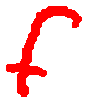
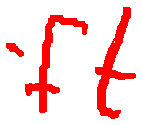
1. Substitution cipher



k :=



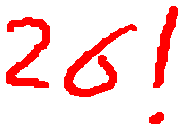
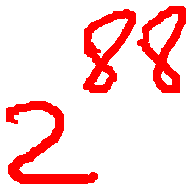
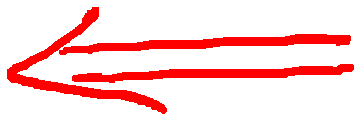
Caesar Cipher (no key)



What is the size of key space in the substitution cipher



assuming 26 letters?



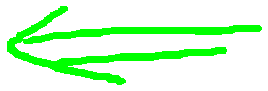
How to break a substitution cipher?

What is the most common letter in English text?

“X”

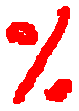
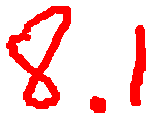
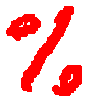
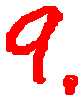
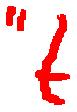
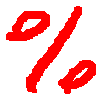
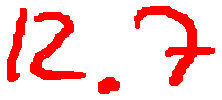
“L”

“E” “H”

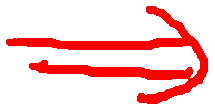
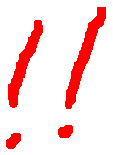
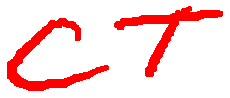
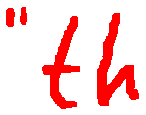


How to break a substitution cipher?

(1) Use frequency of English letters



(2) Use frequency of pairs of letters (digrams)



An Example

|  |  |
| --- | --- |
| **NC** | **11** |
| **PU** | **10** |
| **UB** | **10** |
| **UN** | **9** |

UKBYBIPOUZBCUFEEBORUKBYBHOBBRFESPVKBWFOFERVNBCVBZPRUBOFERVNBCVBPCYYFVUFO FEIKNWFRFIKJNUPWRFIPOUNVNIPUBRNCUKBEFWWFDNCHXCYBOHOPYXPUBNCUBOYNRVNIWN CPOJIOFHOPZRVFZIXUBORJRUBZRBCHNCBBONCHRJZSFWNVRJRUBZRPCYZPUKBZPUNVPWPCYVF ZIXUPUNFCPWRVNBCVBRPYYNUNFCPWWJUKBYBIPOUZBCUIPOUNVNIPUBRNCHOPYXPUBNCUB OYNRVNIWNCPOJIOFHOPZRNCRVNBCUNENVVFZIXUNCHPCYVFZIXUPUNFCPWZPUKBZPUNVR

 E

|  |  |
| --- | --- |
| **B** | **36** |
| **N** | **34** |
| **U** | **33** |
| **P** | **32** |
| **C** | **26** |

 T

 A

**digrams**

 IN

 AT

**trigrams**

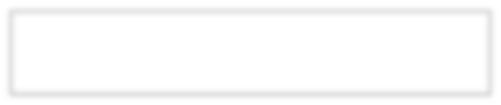
|  |  |
| --- | --- |
| **UKB** | **6** |
| **RVN** | **6** |
| **FZI** | **4** |

 THE

2. Vigener cipher (16’th century, Rome)

k = **C R Y P T O**

**C R Y P T O**



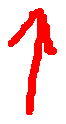
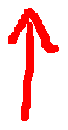
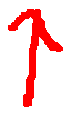
**C R Y P T**

(+ mod 26)

m = **W H A T A N I C E D A Y T O D A Y**



c = **Z Z Z J U C L U D T U N W G C Q S**



suppose most common = “H” first letter of key = “H” – “E” = “C”

3. Rotor Machines (1870-1943)

Early example: the Hebern machine (single rotor)

A

B

C

.

.

X

Y

Z key

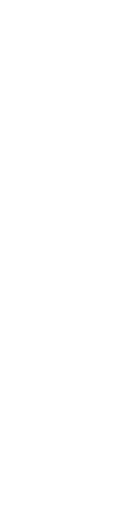
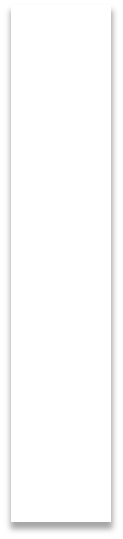
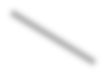
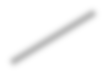
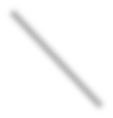
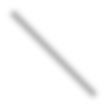
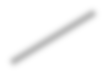
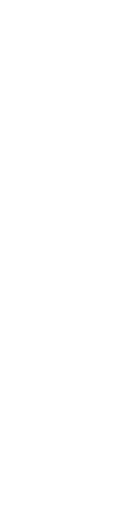
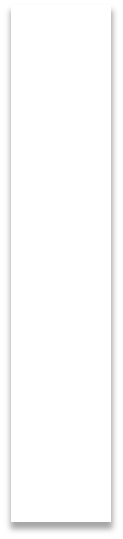
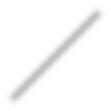
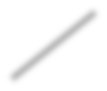
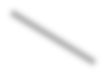
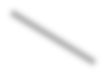
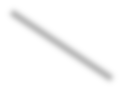
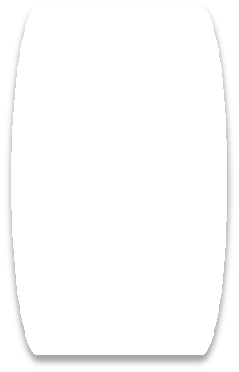
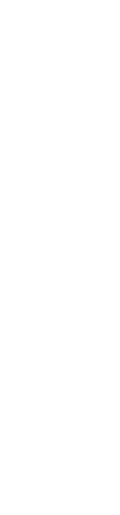
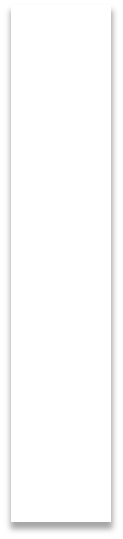
K E N

S K E

T S K

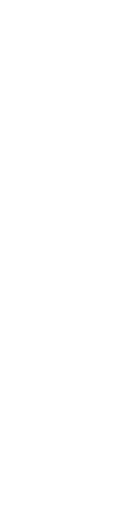
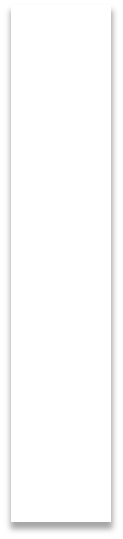
. T S

. . T



R . .

N R .

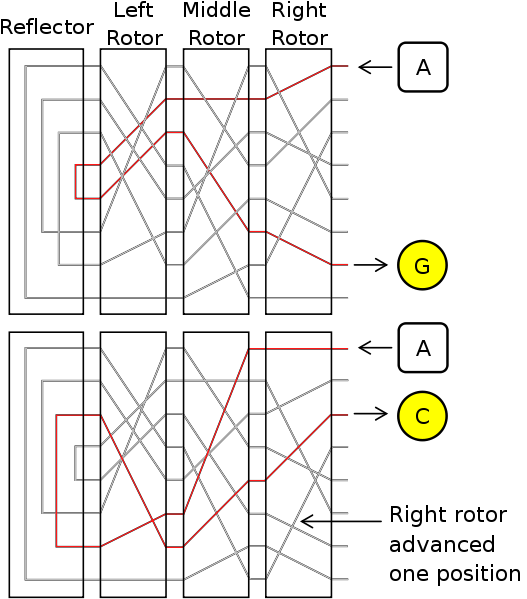


E N R



Rotor Machines (cont.)

Most famous: the Enigma (3-5 rotors)



# keys = 264 = 218 (actually 236 due to plugboard)

4. Data Encryption Standard (1974)

DES: # keys = 256 , block size = 64 bits

Today: AES (2001), Salsa20 (2008) (and many others)