18733: Applied Cryptography Anupam Datta (CMU)

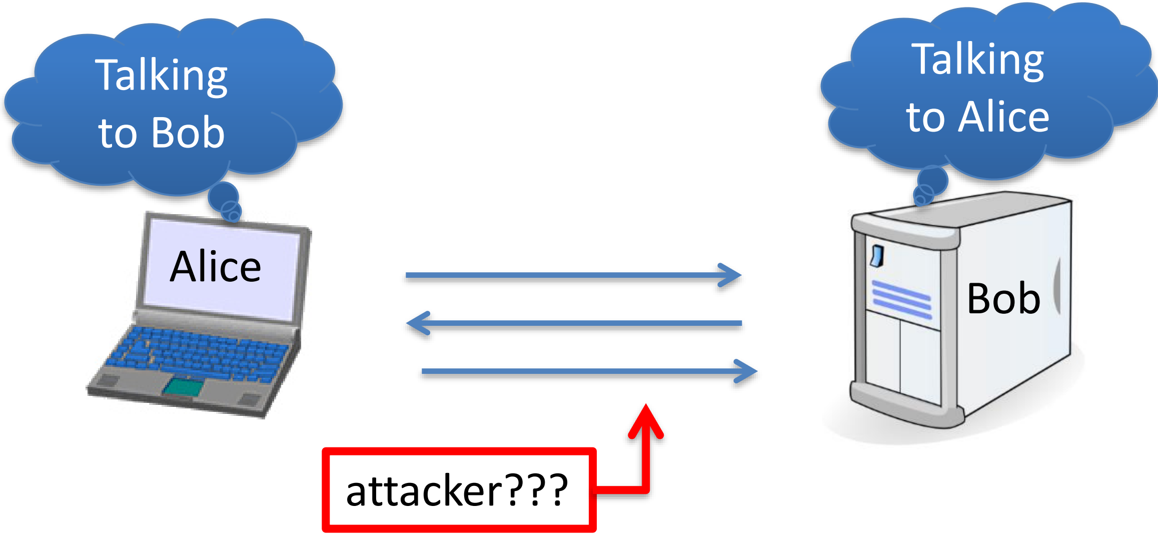
Course Overview

Introduction

What is cryptography?

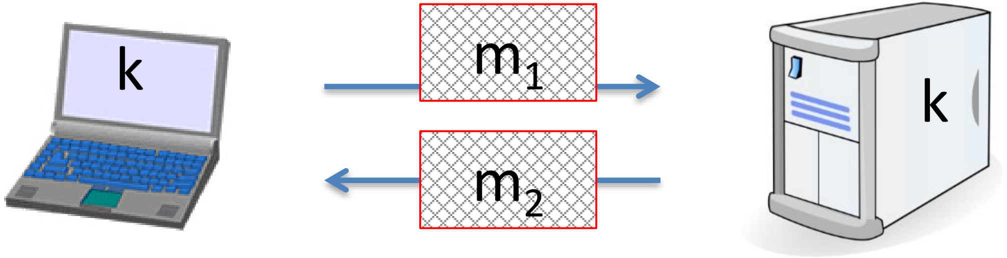
Crypto core

Secret key establishment:



attacker???

Secure communication:

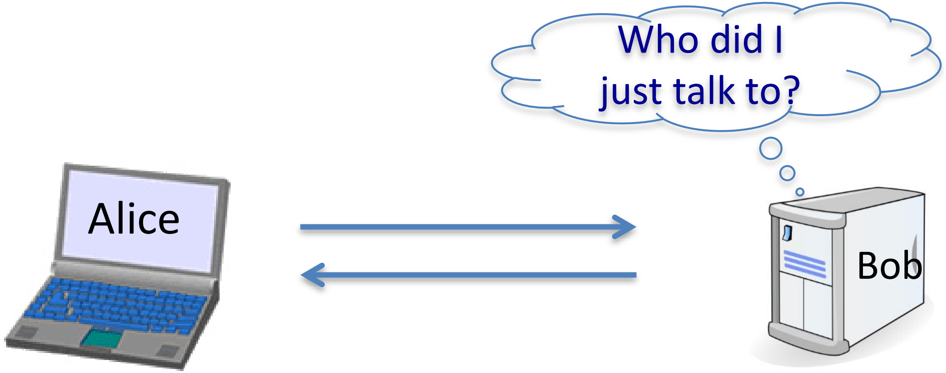


confidentiality and integrity

But crypto can do much more

- Digital signatures

- Anonymous communication

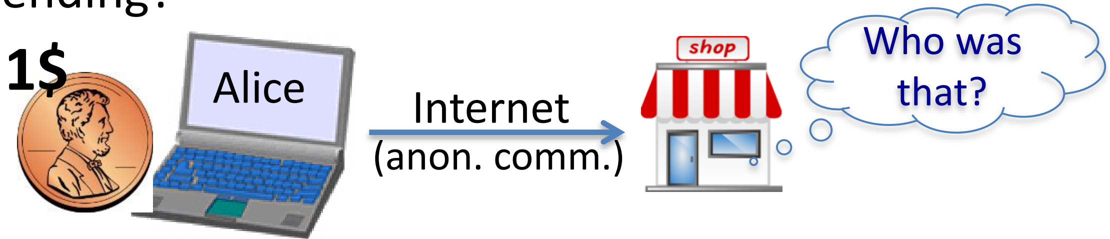


But crypto can do much more

- Digital signatures

- Anonymous communication

- Anonymous **digital** cash
  - Can I spend a “digital coin” without anyone knowing who I am?
  - How to prevent double spending?



Protocols

- Elections
- Private auctions

A rigorous science

The three steps in cryptography:

- Precisely specify threat model
- Propose a construction
- Prove that breaking construction under threat model will solve an underlying hard problem

Turing Award winning ideas

- Manuel Blum (1995)
- Andrew Yao (2000)
- Ron Rivest, Adi Shamir, Leo Adleman (2002)
- Silvio Micali, Shafi Goldwasser (2012)
- Martin Hellman, Whit Diffie (2015)

Impact on practice

- SSL/TLS for secure online communication
  - Employs practical encryption, signature schemes (e.g., based on RSA, Diffie-Hellman)

- Anonymous communication systems like Tor

- Electronic cash schemes like Bitcoin

End of Segment

Logistics

Introductions

- Instructor: Anupam Datta
  - Office hours: SV Bldg 23, #221 + Google Hangout (details on Piazza)

    Mon 1:30-2:30 PM PT

- TA: Gihyuk Ko
  - Office hours: Pitt CIC 2214 + Google Hangout (details on Piazza)

    Thursday 10am - 11am Pacific

Extra office hours on demand

Logistics

- Lectures: Monday & Wednesday, 11:30-1:20pm Pacific
- Recitation: Friday 8:30-9:20am Pacific (attend!)
- Web page: http://www.ece.cmu.edu/~ece733/
- Course blackboard (for grades)
- Piazza (for all other communication)
  - Please enroll; you should have received invitation
- Course work and grading:
  - Homework (80%) – 4 x 20% [written + 1 programming problem per hw]
    - Best 4 of 5 homeworks
  - Mini-poject (10%) [programming problem]
  - Class participation (10%)

Logistics (3)

Collaboration policy:

- You are allowed to discuss homework problems and approaches for their solution with other students in the class, but are required to figure out and write out detailed solutions independently and to acknowledge any collaboration or other source

CMU Computing Policy

CMU Academic Integrity Policy

Logistics (4)

Example Violations:

- Submission of work completed or edited in whole or in part by another person.
- Supplying or communicating unauthorized information or materials, including graded work and answer keys from previous course offerings, in any way to another student.
- Use of unauthorized information or materials, including graded work and answer keys from previous course offerings.
- …not exhaustive list

If in doubt, ask me!

Prerequisites

- An undergraduate course equivalent to 15-251 is recommended or permission of instructor
- An introductory course in computer security such as 18-730 is required or permission of instructor
- If in doubt, please talk to me after class
- Quick class poll

Course Objectives and Content

Course Structure

Three modules

- Symmetric Key Cryptography
  - Encryption, message integrity, hash functions
- Public Key Cryptography
  - Encryption, digital signatures
- Protocols
  - Authentication
  - Accountability
  - Anti-surveillance

Course Resources

- Lecture and recitation
- Textbook (recommended for first two modules)
  - Jonathan Katz, Yehuda Lindell, Introduction to Modern Cryptography
    - eBook available from CMU Library: http://cm.eblib.com/patron/FullRecord.aspx?p=1619504
    - Hardcopy on reserve at Sorrells Library

Learning Outcomes

- Understand theory and practice of cryptography

The 3A's:

- Algorithms: Understand constructions of cryptographic primitives and protocols
- Analysis: Understand security definitions and proofs of primitives and protocols
- Applications: Understand how to use cryptography correctly and attacks that exploit incorrect use

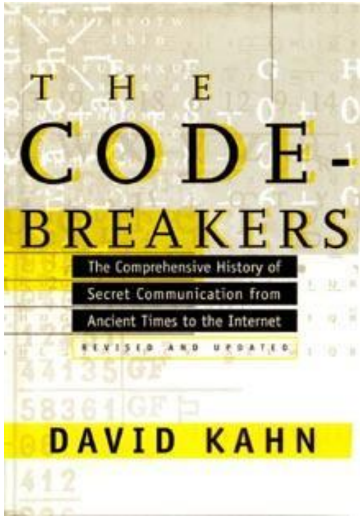
End of Segment

Introduction

History

History

David Kahn, “The code breakers” (1996)


THE CODE-BREAKERS
The Comprehensive History of Secret Communication from Ancient Times to the Internet
REVISED AND UPDATED
DAVID KAHN

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?

$|\mathcal{K}| = 26$

$|\mathcal{K}| = 26!$ (26 factorial)

$|\mathcal{K}| = 2^{26}$

$|\mathcal{K}| = 26^{2}$

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

UKBYBIPOUZBCUFEEBORUKBYBHOBBRFESPVKBWFOFERVNBCVBZPRUBOFERVNBCVBPCYYFVUFO
FEIKNWFRFIKJNUPWRFIPOUNVNIPUBRNCUKBEFWWFDNCHXCYBOHOPYXPUBNCUBOYNRVNIWN
CPOJIOFHOPZRVFZIXUBORJRUBZRBCHNCBBONCHRJZSFWNVRJRUBZRPCYZPUKBZPUNVPWPCYVF
ZIXUPUNFCPWRVNBCVBRPYYNUNFCPWWJUKBYBIPOUZBCUIPOUNVNIPUBRNCHOPYXPUBNCUB
OYNRVNIWNCPOJIOFHOPZRNCRVNBCUNENVVFZIXUNCHPCYVFZIXUPUNFCPWZPUKBZPUNVR

B	**36**	→ E
N	**34**	
U	**33**	→ T
P	**32**	→ A
C	**26**	

NC	**11**	→ IN
PU	**10**	→ AT
UB	**10**	
UN	**9**	

digrams

UKB	**6**	→ THE
RVN	**6**	
FZI	**4**	

trigrams

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

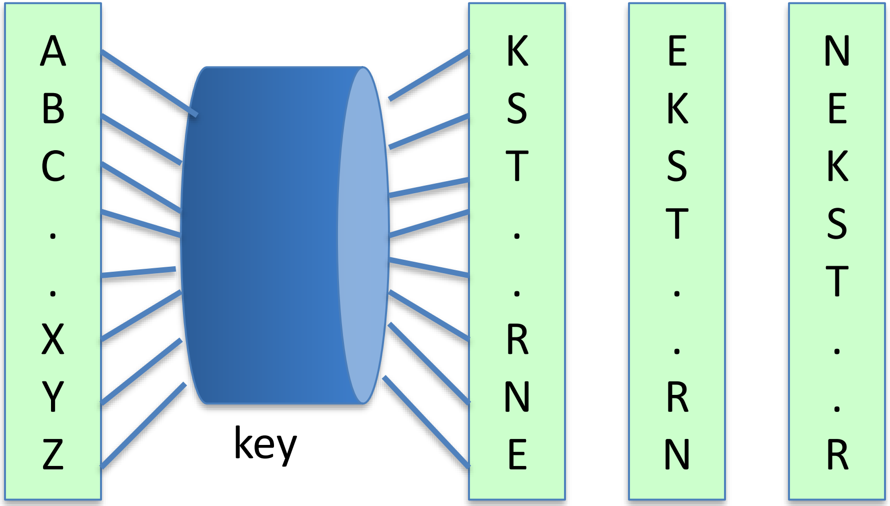
k =	C	R	Y	P	T	O	C	R	Y	P	T	O	C	R	Y	P	T
m =	W	H	A	T	A	N	I	C	E	D	A	Y	T	O	D	A	Y
c =	Z	Z	Z	J	U	C	L	U	D	T	U	N	W	G	C	Q	S

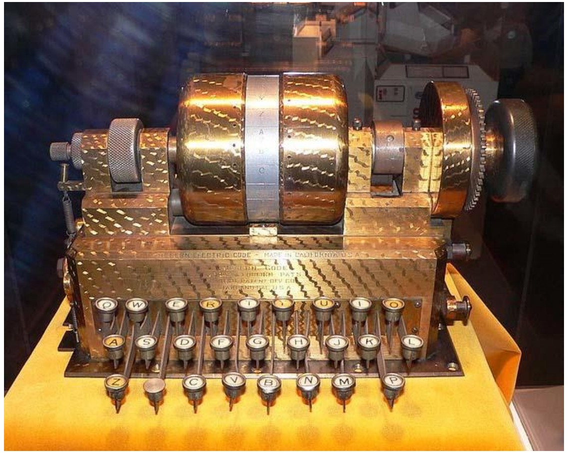
(+ mod 26)

suppose most common = "H" → first letter of key = "H" – "E" = "C"

3. Rotor Machines (1870-1943)

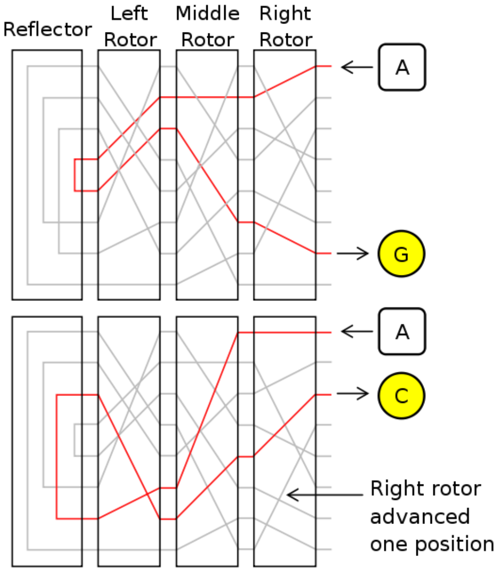
Early example: the Hebern machine (single rotor)



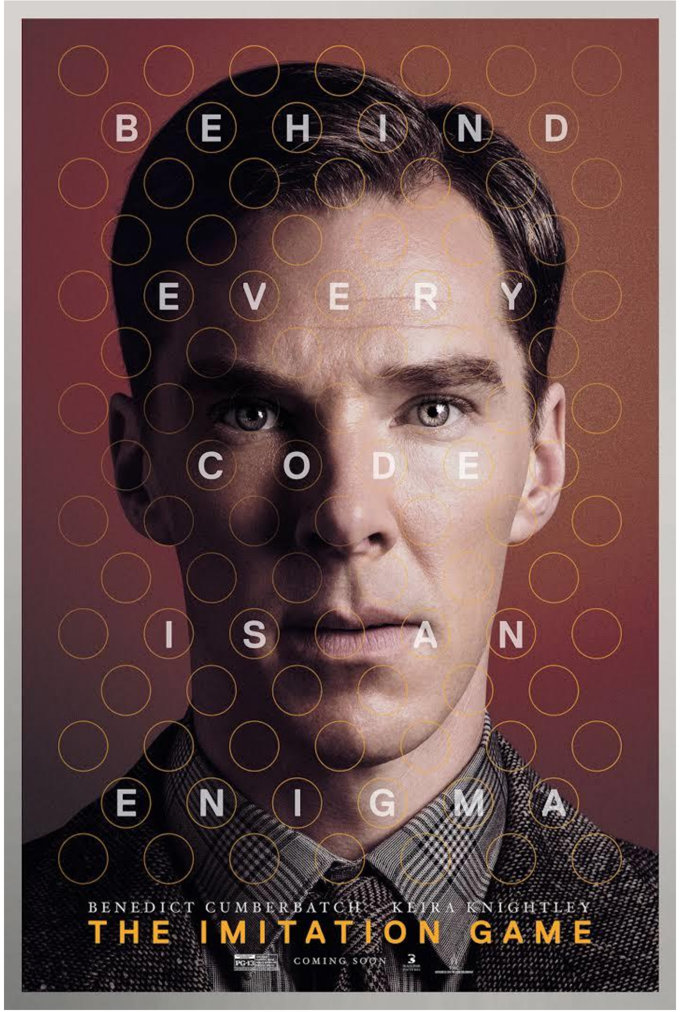


Rotor Machines (cont.)

Most famous: the Enigma (3-5 rotors)



https://en.wikipedia.org/wiki/Enigma_machine


BEHIND
EVERY
CODE
IS AN
ENIGMA
BENEDICT CUMBERBATCH KEIRA KNIGHTLEY
THE IMITATION GAME
PG-13
COMING SOON

4. Data Encryption Standard (1974)

DES: # keys = 2^{56} , block size = 64 bits

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

End of Segment

See also: http://en.wikibooks.org/High_School_Mathematics_Extensions/Discrete_Probability

Introduction

Discrete Probability (crash course)

U: finite set (e.g. $U = \{0,1\}^n$)

Def: **Probability distribution** P over U is a function $P: U \longrightarrow [0,1]$

such that $\sum_{x \in U} P(x) = 1$

Examples:

1. Uniform distribution: for all $x \in U$: $P(x) = 1/|U|$

2. Point distribution at x_0: $P(x_0) = 1$, $\forall x \neq x_0$: $P(x) = 0$

Distribution vector: $\big(P(000), P(001), P(010), \ldots , P(111) \big)$

Events

- For a set $A \subseteq U$: $\Pr[A] = \sum_{x \in A} P(x) \in [0,1]$
- The set A is called an **event**

note: $\Pr[U]=1$

Example: $U = \{0,1\}^8$

- $A = \{$ all x in U such that $\mathrm{lsb}_2(x)=11$ $\} \subseteq U$

for the uniform distribution on $\{0,1\}^8$:

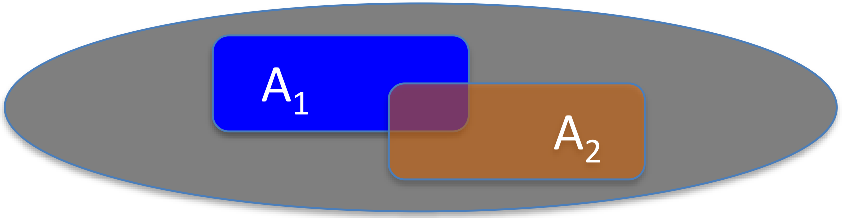
$\Pr[A] =$?

1/4

The union bound

- For events A_1 and A_2

$$\Pr\Big[\, A_1 \cup A_2 \,\Big] \le \Pr[A_1] + \Pr[A_2]$$



Example:

$A_1 = \Big\{$ all x in $\{0,1\}^n$ s.t $\mathrm{lsb}_2(x)=11$ $\Big\}$; $A_2 = \Big\{$ all x in $\{0,1\}^n$ s.t. $\mathrm{msb}_2(x)=11$ $\Big\}$

$$\Pr\Big[\, \mathrm{lsb}_2(x)=11 \text{ or } \mathrm{msb}_2(x)=11 \,\Big] = \Pr\Big[A_1 \cup A_2\Big] \le \tfrac{1}{4}+\tfrac{1}{4} = \tfrac{1}{2}$$

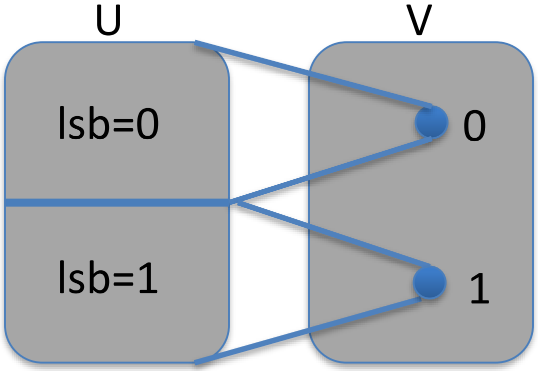
Random Variables

Def: a random variable X is a function $X: U \longrightarrow V$

Example: $X: \{0,1\}^n \longrightarrow \{0,1\}$; $X(y) = \text{lsb}(y) \in \{0,1\}$

For the uniform distribution on U:

$\Pr[\, X=0 \,] = 1/2$, $\Pr[\, X=1 \,] = 1/2$



More generally:

rand. var. X induces a distribution on V: $\Pr[\, X=v \,] := \Pr\big[\, X^{-1}(v) \,\big]$

The uniform random variable

Let U be some set, e.g. $U = \{0,1\}^n$

We write $r \xleftarrow{R} U$ to denote a **uniform random variable** over U

$$\text{for all } a \in U: \quad \Pr\big[\, r = a \,\big] = 1/|U|$$

(formally, r is the identity function: $r(x)=x$ for all $x \in U$)

Let r be a uniform random variable on $\{0,1\}^2$

Define the random variable $X = r_1 + r_2$

Then Pr[X=2] =

¼

Hint: Pr[X=2] = Pr[r=11]

Randomized algorithms

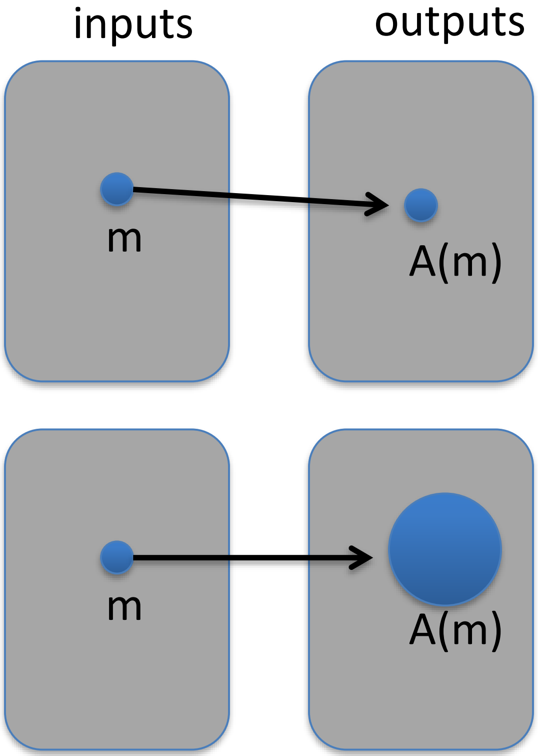
- Deterministic algorithm: $y \longleftarrow A(m)$

- Randomized algorithm

  $y \longleftarrow A(m\,;r)$ where $r \xleftarrow{R} \{0,1\}^n$

  output is a random variable

  $y \xleftarrow{R} A(m)$



Example: $A(m\,;k) = E(k, m)$, $y \xleftarrow{R} A(m)$

End of Segment

See also: http://en.wikibooks.org/High_School_Mathematics_Extensions/Discrete_Probability

Introduction

Discrete Probability (crash course, cont.)

Recap

U: finite set (e.g. $U = \{0,1\}^n$)

Prob. distr. P over U is a function $P: U \longrightarrow [0,1]$ s.t. $\sum_{x \in U} P(x) = 1$

$A \subseteq U$ is called an **event** and $\Pr[A] = \sum_{x \in A} P(x) \in [0,1]$

A **random variable** is a function $X: U \longrightarrow V$.

X takes values in V and defines a distribution on V

Independence

Def: events A and B are **independent** if Pr[A and B] = Pr[A] · Pr[B]

random variables X,Y taking values in V are **independent** if

∀a,b∈V: Pr[X=a and Y=b] = Pr[X=a] · Pr[Y=b]

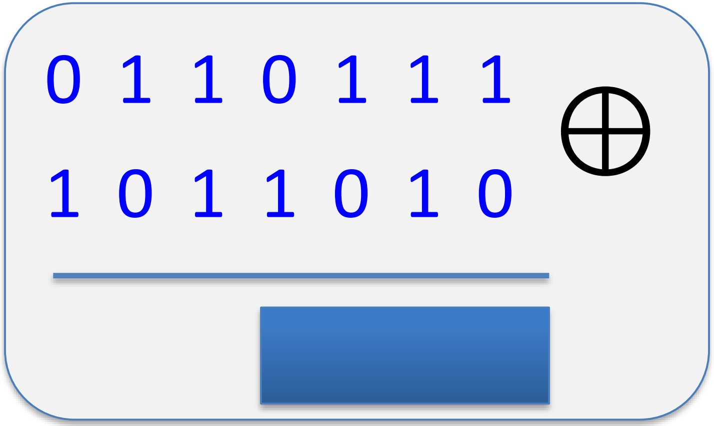
Example: $U = \{0,1\}^2 = \{00, 01, 10, 11\}$ and $r \xleftarrow{R} U$

Define r.v. X and Y as: X = lsb(r) , Y = msb(r)

Pr[X=0 and Y=0] = Pr[r=00] = ¼ = Pr[X=0] · Pr[Y=0]

Review: XOR

XOR of two strings in $\{0,1\}^n$ is their bit-wise addition mod 2



0 1 1 0 1 1 1

1 0 1 1 0 1 0

$\oplus$

An important property of XOR

Thm: Y a rand. var. over $\{0,1\}^n$, X an indep. uniform var. on $\{0,1\}^n$

Then $Z := Y \oplus X$ is uniform var. on $\{0,1\}^n$

Proof: (for $n=1$)

$\Pr[\ Z=0\ ] =$

The birthday paradox

Let $r_1, \ldots, r_n \in U$ be indep. identically distributed random vars.

Thm: when $n = 1.2 \times |U|^{1/2}$ then $\Pr[\ \exists i \neq j:\ r_i = r_j\ ] \geq ½$

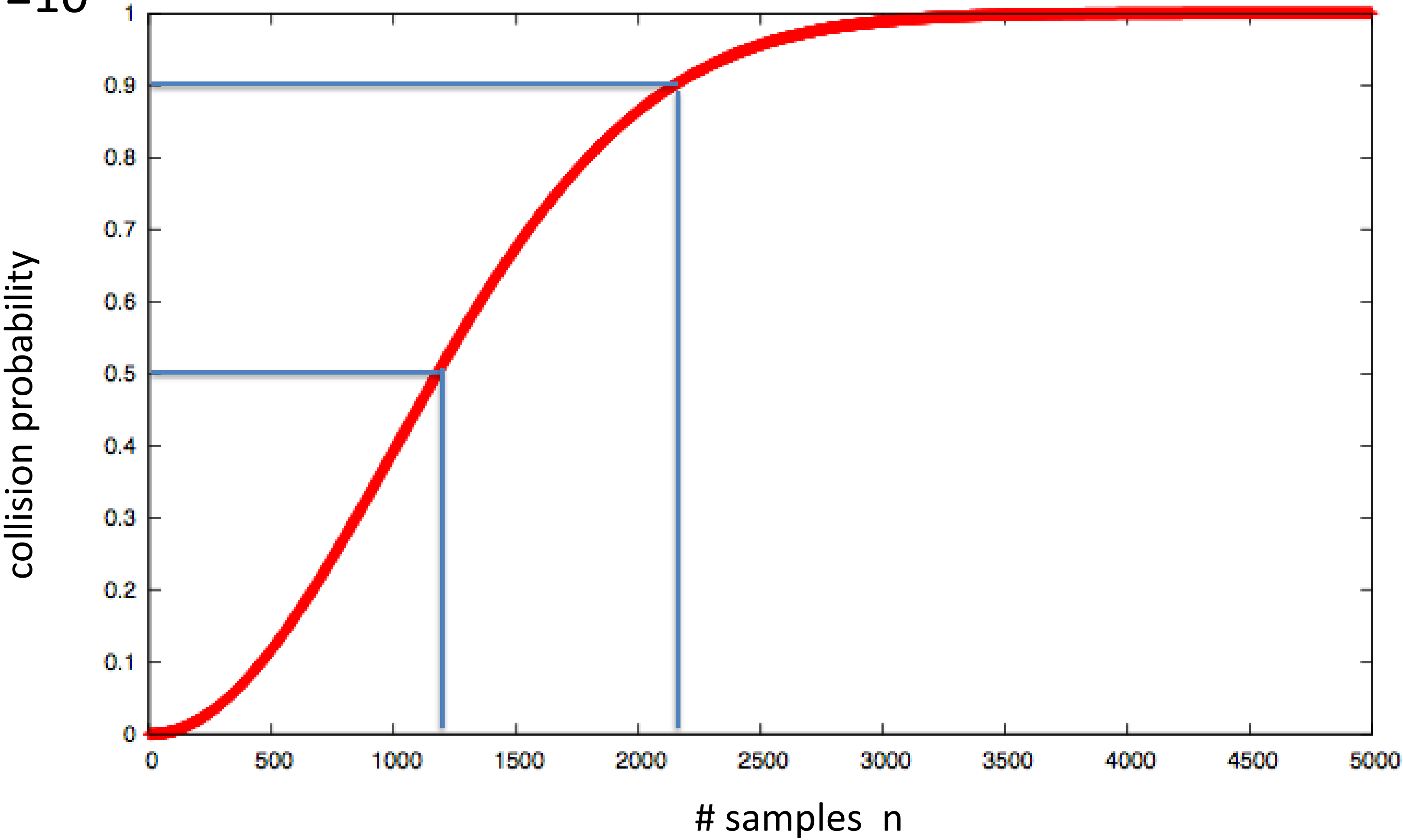
notation: $|U|$ is the size of U

Example: Let $U = \{0,1\}^{128}$

After sampling about 2^{64} random messages from U,

some two sampled messages will likely be the same

$|U|=10^6$


collision probability
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0
500
1000
1500
2000
2500
3000
3500
4000
4500
5000
# samples n

End of Segment