Introduction

Debdeep Mukhopadhyay

Assistant Professor
Department of Computer Science and Engineering
Indian Institute of Technology Kharagpur
INDIA -721302

Objectives

- **A Communication Game**
- **Concept of Protocols**
- **Magic Function**
- **Cryptographic Functions**

A Communication Game

- **Alice and Bob are the two most famous persons in cryptography.**
- **They are used every where…**
- **Consider a scenario, where Alice and Bob wishes to go for dinner together.**
- **Alice decides to go for Chinese, whereas Bob wants to go for Indian Food.**
- **Now how do they resolve?**

Let us use an "unbiased" coin

- **Alice tosses a coin (with his hands covering the coin) and asks Bob of his choice: <u>HEADS or TAILS</u>**
- **If Bob's choice matches with the outcome of the toss, then they go for Indian food. Else Alice has in her way.**
- **Consider the situation when both of them are far apart and communicate through a telephone. What is the problem?**

The problem is now of **"Trust"**

- **Bob cannot trust Alice, as Alice can tell a lie.**
  - **How do we solve this problem?**
- **Solutions to these kind of multi-party (plural number of players) are called technically "protocols"**
- **In order to resolve the problem, both Alice and Bob engage in a "protocol".**
  - **They use a magic function, f(x)**

Properties of f(x)

Assume, Domain and Range of f(x) are the set of integers

1. **For every integer x, it is easy to compute f(x) from x. But given f(x) it is hard to compute x, or find any information about x, like whether x is even or odd (one-wayness)**
2. **It is impossible to find a pair of distinct integers x and y, st. f(x)=f(y)**

The Protocol

- Both of them agree on the function f(x)
- an even number x represents HEAD
- an odd number x represents TAIL

Coin Flipping Over Telephone

- Alice picks up randomly a large integer, x and computes f(x)
- Bob tells Alice his guess of whether x is odd or even
- Alice then sends x to Bob
- Bob verifies by computing f(x)

Security Analysis

- **Can Alice cheat ?**
  - **For that Alice need to create a y≠x, st f(x)=f(y). *Hard to do.***
- **Can Bob guess better than a random guess?**
  - **Bob listens to f(x) which speaks nothing of x. So his probability of guess is ½ (random guess).**

A more concrete example

Alice and Bob wish to resolve a dispute over telephone. We can encode the possibilities of the dispute by a binary value. For this they engage a protocol:

Alice → Bob: Alice picks up randomly an x, which is a 200 bit number and computes the function f(x). Alice sends f(x) to Bob.

Bob → Alice: Bob tells Alice whether x was even or odd.

Alice → Bob: Alice then sends x to Bob, so that Bob can verify whether his guess was correct.

A more concrete example

- **If Bob's guess was right, Bob wins. Otherwise Alice has the dispute solved in her own way.**
- **They decide upon the following function, f: X → Y,**
  - **X is a 200 bit random variable**
  - **Y is a 100 bit random variable**

A Real Instance of f

- **The function f is defined as follows:**

  f(x) = (the most significant 100 bits of x) V (the least significant 100 bits of x), x ε X

  - **Here V denotes bitwise OR.**

Bob's Strategy

- **Bob's Experiment:**
  - **Input f(x)**
  - **Output Parity of x**
- **<u>Algorithm:</u>**

  If $[f(x)]_0$=0, then x is even

  else x is odd

Bob's Probability of Success

- **If X is chosen at random,**

  Pr[X is even]=Pr[X is odd]=1/2

Pr[Bob succeeds]=Pr[X is even]Pr[Bob Succeeds|X is even]+Pr[X is odd]Pr[Bob Succeeds|X is odd]

= ½ ½ + ½ 1 = ¾

Alice's Cheating Probability

- **Remember we compute Alice's cheating probability irrespective of Bob's strategy.**
- **Alice can cheat by changing the parity of x**
- **Case 1: X is even.**
  - **$f(x)]_0$=0, with prob.= ½ . In this case Alice cannot cheat.**
  - **$f(x)]_1$=1, with prob.= ½. In this case Alice can cheat.**
- **So in this case, prob. of success for Alice = ¼ .**

Alice's Cheating Probability

- **Case 2: X is odd.**
  - **$f(x)]_0$=0, this is not possible from the definition of f.**
  - **$f(x)]_0$=1. In this case Alice can cheat.**
- **So in this case, prob. of success for Alice = ½ .**
- **So, Alice can cheat with a prob. of ¼ + ½ = ¾**

How to build the magic function f(.) ?

- Throughout the course we shall see various techniques, methods etc all aimed at discovering these kind of functions.
- They shall be referred to with various terms, like:
  - one-way functions
  - pseudo-random generators
  - hash functions
  - symmetric and a-symmetric ciphers

Practical efficiency

- A mathematical problem is efficient or efficiently solvable when the problem is solved in time and space which can be measured by a small degree polynomial in the size of the problem.
  - The polynomial that describes the resource cost for the user should be small.

Practical efficiency

- **Eg, a protocol with the number of rounds between the users increasing quadratically with the number of users, is not "efficient"**
- **So, we "wish" protocols/algorithms which are not only secure but also efficient.**

References

- **Wenbo Mao, "Modern Cryptography, Theory and Practice", Pearson Education (Low Priced Edition)**

Next Days Topic

- **Overview on Modern Cryptography**