Bonn-Aachen International Center for Information Technology
Cryptography and Game Theory — Raekow, Ziegler
IPEC Winter 2012

Scribe(s): Emil Atanasov — Lecture 9, Date: 19.03.2012

1 Circuit Protocol (CEPS)

We are discussing Circuit Protocol (CEPS). If $t < \frac{n}{2}$ parties are semi-honest then CEPS guaranties perfect correctness and privacy.

We will summarize this in the following table.

	semi-honest	active
perfect secure	$t < \frac{n}{2}$	$t < \frac{n}{3}$
computationaly bounded (PPT)	$t < n$	$t < \frac{n}{2}$

1.1 Protocols

Now let see "Why we design a protocols?".

From cryptographic point of view a protocol is designed to achieve a common goal. (i.e. Sharing a secret data, exchange some information, keeping the privacy). On the other hand the Game Theory subject tries to predict the outcome of given situation that best achieves given goals.

Let's look at the protocol from cryptographic point of view. What can be useful to play "matching pennies"? Matching pennies is a simple game which is played by two players. In all cases either player 1 is a winner or player 2. Each player throw a coin simultaneously with the other one. And then they decide who wins on the following rules. Player 1 wins if the both coins are with the same side on the ground. Otherwise wins the player 2.

Here is the payoff matrix of the game:

P1 \ P2	Head	Tail
Head	$1, -1$	$-1, 1$
Tail	$-1, 1$	$1, -1$

1.2 Commitment Protocols

Here is a problem from the reality. Alice wants to prove to Bob that she can predict a result of an event. How she can do that, in such way, thus Bob can be sure that she has not changed her mind after the actually event ? This example sound too abstract, lets pick a simple representation of this general problem. Alice wants to prove to Bob that she can predict the result from a coin flip. How she can do that, such that Bob is sure that she doesn't cheat?

Commitment Schemas give the answer of this problem. They have tow phases.

1. Commit phase - Alice takes a prediction, puts it in a box, locks it and gives the box to Bob.

2. Opening phase - Alice gives Bob the key for the box.

Note: The box is locked in commit phase, because Bob can change the prediction if the box is not locked.

Such schemas should suffice the following 3 properties:

1. Completeness - Bob can open the commitment and finds the value, put by Alice.
2. Secrecy - As long as Alice doesn't give the key to Bob, he cannot now Alice's prediction.
3. Binding - Bob knows that as soon as he has the box, Alice's prediction is fixed.

In short. The firsts property guaranties that the result of the experiment can be checked. The second one guaranties that Bob doesn't know the prediction. And the last one means that the predicted value is fixed and doesn't depend on the key which Bob will get from Alice.

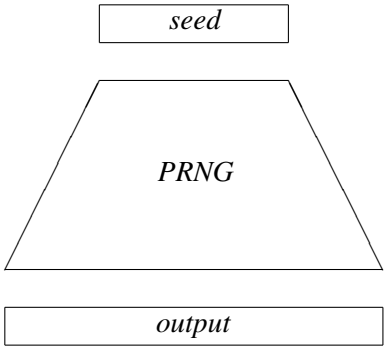
The last point is very important, otherwise Alice can try to hack the system and cheat Bob.

1.3 Pseudo Random Number Generator (PRNG)

We will try to answer the following questions in this section. What is PRNG? Where it is applied?

Definition 1 *(PRNG) A function* $G : \{0,1\}^* -> \{0,1\}^*$ *is PRNG if the followings properties are sufficed:*

1. *Efficiency - G can be deterministically computed in polynomial time (PPT)*
2. *Expansion -* $|G(x)| = l(|x|)$ *for some expansion function L(x) ¿¿ n*
3. *Pseudo randomness - For any PPT algorithm D (distinguisher) there is a negligible function, s.t.* $|\Pr[D(G(x)) = 1]_{x<-Un} - \Pr[D(z) = 1]_{z<-U_{l(n)}}| < \epsilon(n)$
   Un *- is uniform distribution*



Where the seed is n*-bit long and* $output$ *is* $l(n)$ *bit long.*

Remark:

1. Can construct a One Time Pad secure against PPT adversaries.
2. PRNGs exists. provided that one-way function exists.

Here we will discus a protocol for bit commitment (Naor '91). Let $G : \{0,1\}^n -> \{0,1\}^3 n$ be a PRNG.

[Model goes here]

Let $G : \{0,1\}^n -> \{0,1\}^{3n}$ be PRNG. Let denote Alice's prediction with $b \in \{0,1\}$. Here is simple formalization of the protocol in few steps:

1. Bob picks $n \in_R \{0,1\}^{3n}$ (@ random). Then he sends n to Alice.
2. Alice picks $r \in_R \{0,1\}^n$ (@ random). Now follows the commitment phase.
3. Alice computes c as follows: if $b = 0$ then $c = G(r)$ else $c = G(r) \oplus n$.
4. Then c which was computed in previous step is sent back to Bob.
5. Bob stores n, c.
6. Bob executes the experiment. To open the Alice's box he need the key.
7. Alice sends the key r to Bob.
8. Bob computes $G(r)$. If $c = G(r)$ then $b = 0$. If $c = G(r) \oplus n$ then $b = 1$.

Now we will prove some good characteristics of this model.

- Correctness - this is proved by the way we have constructed the protocol.
- Security - this follows from G which is PRNG, $\forall n \in \{0,1\}^3 n, U_{3n}$- uniform distribution. and $U_{3n} \oplus n$ which are identically distributed. Proof: Thus if Bob finds a n, s.t. $G(r)$ and $G(r) \oplus n$ are distinguishable then either $G(r)$ and U_{3n} are distinguishable or $G(r) \oplus n$ and U_{3n} are distinguishable (or both). This is contradiction with that G is PRNG.
- Binding - If Alice is able to convince Bob with $\Pr > \frac{1}{2} + \epsilon(n)$, for $\epsilon(n)$ non-negligable probability that she picked $1 - b$ instead of b, she must be able to choose r_1, r_2, s.t. $G(r_1) = G(r_2) \oplus n$ with $\Pr \geq \epsilon(n)$ for any chosen n by Bob.

  For any $n = G(s_1) \oplus + G(s_2)$ there are unique s_1, s_2. Actually there are 2^n possible values for s_1 and 2^n possible values for s_2. Therefore at most 2^{2n} "bad" n values. Since the space of all n is 2^{3n}, Bob has a chance of selecting a "bad" value with probability $\frac{1}{2^n}$. So even if Alice can compute such pair s_1, s_2, she can do this with probability at most $\frac{1}{2^n}$.

Now we will discuss Blum's protocol for matching pennies. Recall the game and it's goal from the beginning of this lecture.

[Model goes here]

Here is simple formalization of the protocol in steps:

1. Alice picks $r_0 \in_R \{0,1\}$ (@ random). Now follows the commitment phase.
2. Alice computes c as follows: $c = com(r_0)$.
3. Then c which was computed in previous step is sent back to Bob.
4. Bob picks $r_1 \in_R \{0,1\}$ (@ random). Then he sends r_1 to Alice.
5. Bob compares if $r_1 = r_0$ then Alice wins, otherwise Bob wins.
6. Alice opens c. She computes the same as Bob if $r_1 = r_0$ then Alice wins, otherwise Bob wins.

The protocol from above guarantees that all parties will receive their output from the protocol simultaneously.

Theorem 2 (Cleve '86) *For any protocol where [arties agree on one bit, there exist a strategy for one of the parties to bias the output by at least $\frac{\epsilon}{O(l)}$ where l is the number of rounds in the protocol.*

The theorem is out of the scope of this course, but proof could be found in [1].