

## 1 Last Lecture

Last lecture, we introduced the hybrid technique, and used it to prove security for the double encryption method. We also discussed how to use a PRG  $G$  to construct an encryption scheme  $\mathcal{E}_G$ . In this lecture, we will show that the encryption scheme  $\mathcal{E}_G$  satisfies No-Query-Semantic-Security.

## 2 Security of the encryption scheme $\mathcal{E}_G$

Recall the encryption scheme  $\mathcal{E}_G$  with key space  $\mathcal{K} = \{0, 1\}^n$ , message and ciphertext space  $\{0, 1\}^\ell$ . The construction uses a secure pseudorandom generator  $G : \{0, 1\}^n \rightarrow \{0, 1\}^\ell$ .

- $\text{KeyGen}(1^n)$ : chooses a uniformly random key  $k \in \{0, 1\}^n$
- $\text{Enc}(m \in \mathcal{M}, k \in \mathcal{K}) = m \oplus G(k)$
- $\text{Dec}(\text{ct}, k) = \text{ct} \oplus G(k)$

**Theorem 07.01.** Suppose there exists a prob. poly. time adversary  $\mathcal{A}$  and a non-negligible function  $\epsilon$  such that  $\mathcal{A}$  wins the No-Query-Semantic-Security game w.r.t  $\mathcal{E}_G$  with probability  $1/2 + \epsilon$ . Then there exists a prob. poly. time adversary  $\mathcal{B}$  that wins the PRG security game w.r.t.  $G$  with probability  $1/2 + \epsilon_{\mathcal{B}}$ , for some non-negligible function  $\epsilon_{\mathcal{B}}$ .

We had outlined a sketch of the proof in the previous lecture. Therefore, we will provide the full proof here. Recall the hybrid worlds structure from previous lecture. Let  $p_0$  (resp.  $p_1$ ) denote the probability of  $\mathcal{A}$  outputting 0 in world-0 (resp. world-1). Since  $\mathcal{A}$  wins the (bit guessing version of) No-Query-Semantic-Security with probability  $1/2 + \epsilon$ , it follows that  $p_0 - p_1 = \epsilon$ . Finally, let  $p_{\text{hyb},0}$  (resp.  $p_{\text{hyb},1}$ ) denote the probability of  $\mathcal{A}$  outputting 0 in hybrid-world-0 (resp. hybrid-world-1). We argued that using the perfect security of Shannon's OTP,  $p_{\text{hyb},0} = p_{\text{hyb},1} = p_{\text{hyb}}$ .

*Proof.* The reduction algorithm  $\mathcal{B}$  interacts with the PRG challenger and the adversary  $\mathcal{A}$ . It chooses a uniformly random bit  $\beta$  (this bit denotes whether it guesses  $p_0 - p_{\text{hyb}}$  is non-negligible, or  $p_{\text{hyb}} - p_1$ ) and does the following:

- If  $\beta = 0$ , the reduction algorithm guesses that  $p_0 - p_{\text{hyb}}$  is non-negligible. Therefore, it receives two messages  $m_0, m_1$  from  $\mathcal{A}$ . It receives a challenge  $u \in \{0, 1\}^\ell$  from the PRG challenger, and sends  $u \oplus m_0$  to  $\mathcal{A}$ . The adversary sends a bit  $b'$ , which the reduction forwards to the challenger.
- If  $\beta = 1$ , the reduction algorithm guesses that  $p_{\text{hyb}} - p_1$  is non-negligible. Therefore, it receives two messages  $m_0, m_1$  from  $\mathcal{A}$ . It receives a challenge  $u \in \{0, 1\}^\ell$  from the PRG challenger, and sends  $u \oplus m_1$  to  $\mathcal{A}$ . The adversary sends a bit  $b'$ , the reduction sends  $1 - b'$  to the challenger.

Note that in one case, the reduction simply forwards the adversary's guess, while in the other case, it sends  $1 - b'$ . This will be useful for the analysis below.

Let us now analyse the reduction algorithm  $\mathcal{B}$ 's probability of winning the PRG security game.

$$\begin{aligned}
& \Pr [\mathcal{B} \text{ wins the PRG security game}] \\
&= \Pr [\mathcal{B} \text{ outputs } 0 \mid \text{Challenger sends pseudorandom } u] \cdot \frac{1}{2} \\
&\quad + \Pr [\mathcal{B} \text{ outputs } 1 \mid \text{Challenger sends random } u] \cdot \frac{1}{2} \\
&= \Pr [\mathcal{A} \text{ outputs } 0 \mid \beta = 0, \text{Challenger sends pseudorandom } u] \cdot \frac{1}{4} \\
&\quad + \Pr [\mathcal{A} \text{ outputs } 1 \mid \beta = 1, \text{Challenger sends pseudorandom } u] \cdot \frac{1}{4} \\
&\quad + \Pr [\mathcal{A} \text{ outputs } 1 \mid \beta = 0, \text{Challenger sends random } u] \cdot \frac{1}{4} \\
&\quad + \Pr [\mathcal{A} \text{ outputs } 0 \mid \beta = 1, \text{Challenger sends random } u] \cdot \frac{1}{4}
\end{aligned}$$

The first equality follows from the definition of  $\mathcal{B}$  winning the security game. In the second equality, we divide into four cases, depending on whether the reduction algorithm's guess  $\beta = 0$  or  $1$ , and whether the challenger chose  $b = 0$  or  $1$ . Here, note that we use the fact that  $\mathcal{B}$  flips the adversary's guess in the case where  $\beta = 1$ .<sup>1</sup>

Next, observe that

$$\begin{aligned}
\Pr [\mathcal{A} \text{ outputs } 1 \mid \beta = 1, \text{Chall. sends pseudorandom } u] &= 1 - \Pr [\mathcal{A} \text{ outputs } 0 \mid \beta = 1, \text{Chall. sends pseudorandom } u] \\
\Pr [\mathcal{A} \text{ outputs } 1 \mid \beta = 0, \text{Chall. sends random } u] &= 1 - \Pr [\mathcal{A} \text{ outputs } 0 \mid \beta = 0, \text{Chall. sends random } u]
\end{aligned}$$

Finally, note that these probability terms are exactly the probabilities of  $\mathcal{A}$  outputting  $0$  in the four worlds.

$$\begin{aligned}
\Pr [\mathcal{A} \text{ outputs } 0 \mid \beta = 0, \text{Challenger sends pseudorandom } u] &= p_0 \\
\Pr [\mathcal{A} \text{ outputs } 0 \mid \beta = 1, \text{Challenger sends pseudorandom } u] &= p_1 \\
\Pr [\mathcal{A} \text{ outputs } 0 \mid \beta = 0, \text{Challenger sends random } u] &= p_{\text{hyb},0} \\
\Pr [\mathcal{A} \text{ outputs } 0 \mid \beta = 1, \text{Challenger sends random } u] &= p_{\text{hyb},1}
\end{aligned}$$

Putting these together, we get that the probability that  $\mathcal{B}$  wins the security game against the PRG challenger is:

$$\frac{1}{4} (p_0 + (1 - p_1) + (1 - p_{\text{hyb},0}) + p_{\text{hyb},1}) = \frac{1}{2} + \frac{1}{4} (p_0 - p_1)$$

□

### 3 How can we be sure that a function $G$ is a secure PRG?

It would be nice if we can give a formal proof that a function  $G$  is a secure PRG. Unfortunately, we don't know how to prove such statements. In particular, we don't have any unconditional proofs for statements of the form:

*XYZ problem cannot be solved by any polynomial time algorithm*

If we have an unconditional proofs for such statements, then that would resolve the P vs NP question, one of the biggest open questions in computer science.

From a practical viewpoint, we have a few candidate PRGs which have remained secure so far, and therefore we believe they are good candidates for PRGs.

---

<sup>1</sup>Thanks to Himanshu Singh for pointing this out during the lecture.

## 4 Candidate PRG

The PRGs used in practice are quite complicated to describe in a lecture (and also, not the focus of this course). We will see in a few lectures that the AES is often used to implement a PRG. Below, we outline a simple PRG based on the subset-sum problem — a well studied problem in computer science.

Let  $a_1, \dots, a_n$  be numbers in the range  $[0, 2^{n+1}-1]$ . Consider the function  $G_{a_1, \dots, a_n}$  which is parameterised by  $a_1, \dots, a_n$ . The function takes  $n$  bit input  $x_1 x_2 \dots x_n$ , computes  $\sum_i a_i \cdot x_i \bmod 2^{n+1}$  and outputs the result in binary.

Note that the function  $G_{a_1, \dots, a_n} : \{0, 1\}^n \rightarrow \{0, 1\}^{n+1}$ . Of course, there are bad choices of  $a_1, \dots, a_n$  for which  $G_{a_1, \dots, a_n}$  is not a secure PRG. However, as a heuristic, if  $a_1, \dots, a_n$  are chosen uniformly at random, then the function  $G_{a_1, \dots, a_n}$  is believed to be a secure PRG. Again, there is no formal proof for this. This function is very closely related to the **subset-sum** problem, which is **NP-complete**. However, the **NP-completeness** of **subset-sum** says that there exists some instance of the problem which is hard, it says nothing about the hardness of the average-case version.

## 5 Expanding output space of PRG

Let  $G : \{0, 1\}^n \rightarrow \{0, 1\}^{n+1}$  be a secure PRG. We want to build a PRG with greater ‘stretch’ — that is, a function that maps  $n$  bits to  $\ell$  bits, for some  $\ell > n + 1$ . For simplicity, let us start with  $\ell = n + 2$ .

Of course, there are a number of ways of achieving this, but how to ensure that the resulting function is also a secure PRG (assuming  $G$  is a secure PRG)? A very natural construction is the following chaining-based construction:

$$\begin{aligned} G' : \{0, 1\}^n &\rightarrow \{0, 1\}^{n+2} \\ G'(x) &= z_1 z_2 \dots z_{n+1} \parallel y_{n+1} \text{ where} \\ G(x) &= y_1 y_2 \dots y_{n+1}, G(y_1 y_2 \dots y_n) = z_1 z_2 \dots z_{n+1} \end{aligned}$$

To prove that  $G'$  is a secure PRG (assuming  $G$  is a secure PRG), we will use the hybrid technique. Suppose there exists a p.p.t. adversary  $\mathcal{A}$  that breaks the PRG security of  $G'$ . Then we will show a p.p.t. adversary  $\mathcal{B}$  that breaks the PRG security of  $G$ . As in the previous approaches, we will use the two-worlds formulation for  $\mathcal{A}$ , and the bit-guessing formulation for  $\mathcal{B}$ . The formal claim is as follows.

**Theorem 07.02.** Suppose there exists a p.p.t. adversary  $\mathcal{A}$  such that  $p_0 = \Pr \left[ \mathcal{A} \text{ outputs } 0 \text{ in world-0 (w.r.t. } G') \right]$ ,  $p_1 = \Pr \left[ \mathcal{A} \text{ outputs } 0 \text{ in world-1 (w.r.t. } G') \right]$  and  $p_0 - p_1 = \epsilon$ , a non-negligible function. Then there exists a p.p.t. adversary  $\mathcal{B}$  s.t.  $\Pr \left[ \mathcal{B} \text{ wins the bit-guessing PRG security game (w.r.t. } G) \right] \geq 1/2 + \Omega(\epsilon)$ .

We will see the proof in the next lecture.

## 6 Lecture summary, plan for next lecture, additional resources

**Summary** The main takeaway from this lecture are the hybrids required to prove security of  $\mathcal{E}_G$ , and the reductions for proving that consecutive hybrid worlds are indistinguishable.

- world 0 : adversary receives  $m_0 \oplus G(k)$  as the ciphertext.
- hybrid-world-0 : adversary receives  $m_0 \oplus \text{random } r$  as the ciphertext. This is indistinguishable from world 0, since  $G(k)$  and a random string are indistinguishable (via PRG security)
- hybrid-world-1: adversary receives  $m_1 \oplus \text{random } r$  as the ciphertext. Using the Shannon OTP’s security, hybrid-world-0 and hybrid-world-1 are identical.
- world 1 : adversary receives  $m_1 \oplus G(k)$  as the ciphertext. This is indistinguishable from hybrid-world-1, since  $G(k)$  and a random string are indistinguishable (via PRG security).

**Next Lecture:** We started with the construction for expanding output space of a PRG. In the next lecture, we will see a formal proof of security for the chaining argument. Our goal will be to prove the following: assuming  $G : \{0, 1\}^n \rightarrow \{0, 1\}^{n+1}$  is a secure PRG,  $G' : \{0, 1\}^n \rightarrow \{0, 1\}^{n+2}$  is also a secure PRG. In other words, if there exists an adversary that breaks the PRG security of  $G'$  (via the two worlds formulation), then there exists an adversary that breaks the PRG security of  $G$  (the bit guessing version).

**Relevant sections from textbook [Boneh-Shoup]:** Sections 3.2 (the proof description is a bit different from what we saw in class).