1 Summary

In this lecture, we present two variants of the Even-Goldreich-Lempel Oblivious Transfer (OT) protocol we saw in the previous lecture. We show how to do domain extension for OT (i.e.) go from $\ell$-bit string OT to $n$-bit string OT. We also show how to construct 1-out-of-$N$ OT from 1-out-of-2 OT.

2 Variants of Even-Goldreich-Lempel OT protocol

We present the following variant of the OT protocol seen in the last lecture. Let $\mathcal{E} = (\text{Gen, Enc, Dec})$ be a CPA secure encryption scheme that has an algorithm $\text{Samp}$ that can sample valid looking ciphertexts.

\begin{align*}
\text{Sender}(x_0, x_1) & \quad \text{Receiver}(b) \\
(pk, sk) & \leftarrow \text{Gen}(1^n) \\
&(pk, s_0, s_1) \leftarrow \{0, 1\} \\
c_b & = \text{Enc}_{pk}(s_b) \\
c_{1-b} & \leftarrow \text{Samp} \\
s_0, c_0 = \text{Dec}_{sk}(c_0) \\
s_1, c_1 = \text{Dec}_{sk}(c_1) \\
s_0 \oplus x_0, s_1 \oplus x_1 & = x_b = s_b \oplus (s_b \oplus x_b)
\end{align*}

Figure 1: Variant of Even-Goldreich-Lempel OT

The protocol in Figure 1 guarantees information theoretic security against the receiver and computational security against the sender. This protocol can be instantiated with the ElGamal encryption scheme.

The next variant is based on the Decisional Diffie-Hellman (DDH) assumption. Consider the group $G$ of order $q$ with generators $g_0$ and $g_1$. The DDH assumption states that, for $r, r' \in \mathbb{Z}_q$, the tuples $(g_0, g_1, g_0^r, g_1^{r'})$ and $(g_0, g_1, g_0^{r'}, g_1^{r'})$ are computationally indistinguishable.

Consider the following protocol with group $G$ of order $q$ and generators $g_0$ and $g_1$. 

4-1
Sender \((x_0, x_1)\)

Receiver \(b\)

- Choose \(r \in \mathbb{Z}_q\)
- If \(b = 0\), set \(h_0 = g_0^r, h_1 = g_1^r\)
- If \(b = 1\), set \(h_0 = g_0^r, h_1 = g_1^{r+1}\)

\[ a_0, b_0, a_1, b_1 \leftarrow \mathbb{Z}_q \]

\[ c_0 = (g_0^{a_0} g_1^{b_0}, h_0^{a_0} h_1^{b_0} x_0) \]

\[ c_1 = (g_0^{a_1} g_1^{b_1}, h_0^{a_1} h_1^{b_1} / g_1^{b_1} x_1) \]

Parse \(c_b\) as \((c_1, c_2)\)

Compute \(x_b = c_2 / (c_1)^r\)

Figure 2: OT protocol based on DDH

3 More constructions using OT hybrid

3.1 Domain Extension

We show a construction of OT for \(\ell\)-bit strings from OT for \(n\)-bit strings.

The sender and receiver use the OT protocol for \(n\)-bits to choose key \(k_b\) corresponding to the receiver’s choice bit from keys \(k_0, k_1\) held by the sender. The sender sends both \(\ell\)-bit strings \(m_0, m_1\) encrypted using the keys \(k_0, k_1\) respectively. The receiver then recovers the correct message using his key \(k_b\).

The protocol is as described in Figure 3.

Note that \(n\) is the security parameter. This protocol cannot be used for small values of \(n\). For example, when \(n = 1\), the key would be of length 1 and this protocol is no longer secure.

3.2 Constructing 1-out-of-N OT from 1-out-of-2 OT

Let us show how to construct 1-out-of-N OT (i.e.) the receiver choosing to receive 1 string among \(N\) held by the sender from a 1-out-of-2 OT construction. To illustrate the general idea, we show a construction of 1-out-of-4 OT from 1-out-of-2 OT.
Sender\((m_0, m_1 \in \{0, 1\}^\ell)\) 

Sender \((m_0, m_1, m_{10}, m_{11})\) 

The sender holds two pairs of keys and the receiver chooses one key from each pair corresponding to the message he wants to receive. Note that the receiver has to hold two bits here. Let \(F\) be a pseudorandom function such that \(F(k, \cdot) : \{0, 1\}^n \rightarrow \{0, 1\}^n\).

Note that using the corresponding keys as pads for each message (i.e.) setting \(c_{ii'} = m_{ii'} \oplus k_i \oplus k_{i'}\) reveals more than just the message chosen by the receiver. This is because the receiver can xor all the ciphertexts to learn the xor of all the messages.

Another secure construction would be to consider the keys as encryption keys and encrypt the messages as \(c_{ii'} = \text{Enc}_{k_i}(\text{Enc}_{k_{i'}}(m_{ii'}))\) and let the receiver obtain the correct message by decrypting the corresponding ciphertext with the keys he received from the OT.