1 Topic Covered

- Pseudorandom Functions.
- From OWFs to PRFs via PRGs: the GGM Construction.
- From PRGs to One-Time Symmetric Encryption.

2 Pseudorandom Functions (PRF)

High-level idea:

- PRG: short random seed \( s \mapsto G(s) \) long “random looking” output.
- PRF: short random key \( K \mapsto F_K(\cdot) \) “random looking” function.

**Definition 1** [Pseudorandom Function] A pseudorandom function (PRF) is a family of functions \( \{ F_K : \{0,1\}^{m(n)} \to \{0,1\}^{\ell(n)} \}_{n \in \mathbb{N}, K \in \{0,1\}^n} \) such that:

- **Efficiency**: one can compute \( F_K(x) \) in \( \text{poly}(n) \)-time (given \( K \) and \( x \));
- **Security**: for any poly-time adversary \( A \):

\[
\left| \Pr \left[ A^{F_K(\cdot)}(1^n) = 1 \right] - \Pr \left[ A^{R(\cdot)}(1^n) = 1 \right] \right| \leq \text{negl}(n).
\]

where \( K \leftarrow \{0,1\}^n \) and \( R \leftarrow F(\{0,1\}^{m(n)} \to \{0,1\}^{\ell(n)}) \) with \( F(\{0,1\}^{m(n)} \to \{0,1\}^{\ell(n)}) \) denoting the set of all functions mapping \( m(n) \) bits to \( \ell(n) \) bits.

**Remark 1** Note that describing a truly random function \( R : \{0,1\}^{m(n)} \to \{0,1\}^{\ell(n)} \) would require \( 2^{m(n)} \cdot \ell(n) \) bits. So we cannot even efficiently describe such functions, let alone evaluate them. However, the point of building pseudorandom functions, is to have a function \( F_K \) that looks like a random function to the outside world, but can be described with only an \( n \) bit key \( K \) and can be evaluated efficiently.

You might want to think of \( A^{R(\cdot)} \) as the experiment where the outputs of \( R(\cdot) \) are chosen “on the fly” uniformly at random, since \( R(\cdot) \) doesn’t have a short description as a truly random function. In other words, each time \( A \) calls the function on a fresh input \( x \), we choose a fresh output \( y \) and remember the pair \( (x,y) \) in case \( x \) gets queried again.
Another way to define the security of a pseudorandom function is to use a definition based on the indistinguishability of two experiments.

**Definition 2 [Indistinguishability of Experiments]** Let $\exp_1(n)$ and $\exp_2(n)$ denote two experiments with an adversary $A$. We say that the experiments $\exp_1(n)$ and $\exp_2(n)$ are (computationally) indistinguishable if for all PPT adversary $A$,

$$|\Pr[\exp_1_A(n) = 1] - \Pr[\exp_2_A(n) = 1]| \leq \text{negl}(n).$$

where the notation $\exp_A(n)$ denotes the adversary $A$ participating in the experiment. $\Diamond$

Defining a security notion via indistinguishability of experiments is very common and convenient. For instance, we can define the security of a pseudorandom function $\{F_K : \{0,1\}^{m(n)} \to \{0,1\}^{\ell(n)}\}_{n \in \mathbb{N}, K \in \{0,1\}^n}$ as the indistinguishability of the experiments $\exp_1(n)$ and $\exp_2(n)$, defined as follows: in $\exp_1(n)$, one picks $K \leftarrow \{0,1\}^n$ uniformly at random and the adversary is given black-box access to the function $F_K(\cdot)$, while in $\exp_2(n)$, one picks $R \leftarrow F(\{0,1\}^{m(n)} \to \{0,1\}^{\ell(n)})$ uniformly at random and the adversary is given black-box access to the function $F(\cdot)$.

### 3 From PRGs to PRFs: the GGM construction

**Theorem 1 (Golreich, Goldwasser, Micali [1])** Given any pseudorandom generator (PRG) we can construct a pseudorandom function (PRF) for any polynomials $m(\cdot), \ell(\cdot)$ defining the lengths of the input and output. (Since we also know that PRGs can be constructed from OWFs, this says that PRFs can be constructed from OWFs).

Let us first show how to construct a PRF with output length $n$.

**Construction 1** Let $G : \{0,1\}^n \to \{0,1\}^{2n}$ denote a pseudorandom generator, and let us denote by $(G_0(K), G_1(K)) = G(K)$ the first and second (n-bit) halves of $G(K)$. Let $\{F_K : \{0,1\}^{m(n)} \to \{0,1\}^{\ell(n)}\}_{n \in \mathbb{N}, K \in \{0,1\}^n}$ denote the family of functions defined by:

$$F_K(x) = G_{x_{m(n)}}(G_{x_{m(n)-1}}(\ldots(G_{x_1}(K))\ldots)),$$

where $x = x_1 \ldots x_{m(n)}$ is the input and $K \in \{0,1\}^n$ is the key.

A more intuitive view of this construction is depicted in Figure 1 for $m(n) = 3$ and using the following notation: $K_{s\|b} = G_b(K_s)$, for any $K_s \in \{0,1\}^n$, any $s \in \{0,1\}^s$ and any $b \in \{0,1\}$. Therefore, the evaluation of $F_K$ on input $x = x_1x_2x_3$ is the value $K_{x_1x_2x_3} = G_{x_3}(G_{x_2}(G_{x_1}(K)))$.

**Theorem 2** Assuming $G$ is a pseudorandom generator, the family of functions defined in Construction 1 is a pseudorandom function.

**Efficiency.** Efficiency is straightforward as evaluating $F_K$ on a fresh input corresponds simply to evaluating $m(n)$ times the pseudorandom generator $G$ and as $m(n)$ is polynomial.

**Security (sketch of a proof).** The idea for the proof is to show that, under the security of the underlying pseudorandom generator, one can change values in each node in
a fixed level of the tree, defined in Figure 1, to uniformly random values. Then, starting from the root and changing one level at a time, one can change all the values in the nodes of the tree to uniformly random values. There is a small problem with this intuition: there is an exponential number of nodes ($2^i$ nodes at level $i$), thus this proof is not polynomial time. One can easily circumvent this problem by simulating values on the fly: one changes only the values that are useful to respond to the adversary queries. As the adversary is polynomial-time, one needs to change only a polynomial number of values. A formal proof is given below.

**Proof:** Let us first prove the following intermediate result.

**Claim 1** Let $G : \{0,1\}^n \rightarrow \{0,1\}^{2n}$ be a pseudorandom generator. Let $t(n)$ be a polynomial number (in $n$). Then, we have:

$$\{(G(K_1), \ldots, G(K_{t(n)}))\}_{K_1, \ldots, K_{t(n)} \in \{0,1\}^n} \approx_c \{\mathcal{U}_{2n}, \ldots, \mathcal{U}_{2n}\},$$

where $\mathcal{U}_{2n}$ denotes the uniform distribution over $\{0,1\}^{2n}$.

**Proof:** We show the following claim by a simple hybrid argument. Let $\mathcal{D}_i = \{(G(K_1), \ldots, G(K_{t(n)-i})), \mathcal{U}_{2n}, \ldots, \mathcal{U}_{2n}\}_{K_1, \ldots, K_{t(n)-i} \in \{0,1\}^n}$, for $i = 0, \ldots, t(n)$. First, it is clear that $\mathcal{D}_0 = \{(G(K_1), \ldots, G(K_{t(n)}))\}_{K_1, \ldots, K_{t(n)} \in \{0,1\}^n}$ and that $\mathcal{D}_{t(n)} = \{\mathcal{U}_{2n}, \ldots, \mathcal{U}_{2n}\}$. Then, we just need to show that $\mathcal{D}_i \approx_c \mathcal{D}_{i+1}$ for all $i = 0, \ldots, t(n) - 1$ to prove the above claim.

The only difference between $\mathcal{D}_i$ and $\mathcal{D}_{i+1}$ lies in the $(i+1)$-th component of the vector, which is on the one hand computed as the evaluation of the PRG $G$ on a uniformly random input $K_i$, and on the other hand set to a uniformly random value. By definition of the security of a pseudorandom generator, these two distributions are computationally indistinguishable.

The claim follows.

**Remark 2** The above claim is no longer true if $t(n)$ is superpolynomial.
We can now proof Theorem 2 using this statement.

Let us define the experiment $\text{Exp}^i(n)$, for $i = 0, \ldots, m(n)$ as follows: it starts by initializing two empty arrays $T_1, T_2$. When adversary $A$ makes a query $x \in \{0,1\}^{m(n)}$, one checks if $s = x_1 \ldots x_i$ is in $T_1$. If it does not, one picks $K_s \sim \{0,1\}^n$ at random and adds $s$ to $T_1$ and $K_s$ to $T_2$ in the last position. If it does, let $K_s = T_2[i_1(s)]$, where $i_1(s)$ denotes the index of $s$ in $T_1$. Finally, output $y = G_{x_{m(n)}}(G_{x_{m(n)-1}}(\ldots(G_{x_{i+1}}(K_s))\ldots))$

First, it is clear that $\text{Exp}^0(n)$ and $\text{Exp}^{m(n)}(n)$ are exactly the same as the ones defining the PRF security of $F$, as in $\text{Exp}^0(n)$, one just checks if $\varepsilon \in T_1$ at every query, so a key $K_\varepsilon$ is picked at random at the first query and is used to respond to all following queries by outputting $y = G_{x_{m(n)}}(G_{x_{m(n)-1}}(\ldots(G_{x_1}(K_\varepsilon))\ldots)) = F_{K_\varepsilon}(x)$, while in $\text{Exp}^{m(n)}(n)$, for every query $x$, one checks if $x \in T_1$ (which is not the case if $x$ is a new input) and outputs a random value $K_x \sim \{0,1\}^n$ for this input. Then, all values output are truly random values. Then, we just need to argue that $\text{Exp}^i(n)$ and $\text{Exp}^{i+1}(n)$ are indistinguishable for any $i = 0, \ldots, m(n) - 1$, and by a standard hybrid argument, the security of the PRF will follow.

The only difference between experiments $\text{Exp}^i(n)$ and $\text{Exp}^{i+1}(n)$ is the following: For any input $x \in \{0,1\}^{m(n)}$, in $\text{Exp}^i(n)$, one evaluates $G$ on input $G_{x_{i+1}}(K_{x_1 \ldots x_i})$ where $K_{x_1 \ldots x_i}$ is a random $n$-bit string and use the part $G_{x_{i+2}}$ as input for the outer PRG call, while in $\text{Exp}^{i+1}(n)$, one evaluates $G$ directly on a uniformly random $n + 1$-bit string $K_{x_1 \ldots x_{i+1}}$ associated to $x_1 \ldots x_{i+1}$ and use to part $G_{x_{i+2}}(K_{x_1 \ldots x_{i+1}})$ as input for the outer PRG call.

Hence, to argue the indistinguishability of experiments $\text{Exp}^i(n)$ and $\text{Exp}^{i+1}(n)$, it is sufficient to prove that the two distributions $\{G(K_1), \ldots, G(K_{t(n)})\}_{K_1, \ldots, K_{t(n)} \in \{0,1\}^n}$ and $\{(\mathcal{U}_{1n}, \ldots, \mathcal{U}_{2n})\}$ are indistinguishable, where $K_1, \ldots, K_{t(n)}$ denote all the random $n$-bit strings associated to all strings $x_1 \ldots x_i \in \{0,1\}^i$ that are prefix of queries of $A$. As $t(n)$ is at most the number of queries made by $A$, which is polynomial in $n$, this follows directly from Claim 1.

Theorem 2 follows.

We can now use the above construction to build pseudorandom function for any polynomial output length.

Claim 2 Let $F = \{F_K : \{0,1\}^{m(n)} \rightarrow \{0,1\}^n\}_{n \in \mathbb{N}, K \in \{0,1\}^n}$ be a pseudorandom function and $G : \{0,1\}^n \rightarrow \{0,1\}^{\ell(n)}$ be a pseudorandom generator, then $F' = \{G \circ F_K : \{0,1\}^{m(n)} \rightarrow \{0,1\}^{\ell(n)}\}_{n \in \mathbb{N}, K \in \{0,1\}^n}$ is a pseudorandom function.

Proof: [Claim 2] The proof follows an hybrid argument. Let $A$ be an adversary against the PRF security of $F'$. We can assume without loss of generality that $A$ never repeats a query. Let $x_1, \ldots, x_{t(n)}$ denote its queries. Then, assuming $F$ is a pseudorandom function, the distributions $\{G \circ F_K(x_1), \ldots, G \circ F_K(x_{t(n)})\}_{K \in \{0,1\}^n}$ and $\{G(s_1), \ldots, G(s_n)\}_{s_1, \ldots, s_{t(n)} \in \{0,1\}^n}$ are computationally indistinguishable. Now, assuming $G$ is a pseudorandom generator, the distributions $\{G(s_1), \ldots, G(s_n)\}_{s_1, \ldots, s_{t(n)} \in \{0,1\}^n}$ and $\{y_1, \ldots, y_{t(n)}\}_{y_1, \ldots, y_{t(n)} \in \{0,1\}^{\ell(n)}}$ are computationally indistinguishable.

The claim follows.

Proof: [Theorem 1] Theorem 1 now easily follows from Theorem 2, Claim 2 and from the fact that one-way functions imply pseudorandom generators (Goldreich-Levin).
4 One-Time Symmetric Encryption

Definition 3 [One-Time Symmetric Encryption] $\Pi = (\text{Enc}, \text{Dec})$ is a one-time symmetric encryption scheme with message length $\ell(n)$ if $\text{Enc} : \{0,1\}^* \times \{0,1\}^* \rightarrow \{0,1\}^*$ and $\text{Dec} : \{0,1\}^* \times \{0,1\}^* \rightarrow \{0,1\}^*$ are two PPT algorithms such that:

- **Correctness:** for all $n \in \mathbb{N}, K \in \{0,1\}^n, m \in \{0,1\}^{\ell(n)}$,
  \[ \Pr[\text{Dec}(K, \text{Enc}(K, m)) = m] = 1. \]

- **Security:** for any PPT adversary $\mathcal{A}$, experiments $\text{Exp}_0^0(n)$ and $\text{Exp}_1^1(n)$ are indistinguishable, where $\text{Exp}_b^b(n)$ is defined as follows: $\mathcal{A}(1^n)$ outputs $m_0, m_1 \in \{0,1\}^{\ell(n)}$, one picks $K \overset{\$}{\leftarrow} \{0,1\}^n$ at random and sends $c = \text{Enc}(K, m_b)$ to $\mathcal{A}$. $\mathcal{A}$ outputs $b'$ (output of the experiment).

Proposition 1 Assuming $G : \{0,1\}^n \rightarrow \{0,1\}^{\ell(n)}$ is a pseudorandom generator, $\Pi = (\text{Enc}, \text{Dec})$ with $\text{Enc} : (K, m) \in \{0,1\}^n \times \{0,1\}^{\ell(n)} \rightarrow G(K) \oplus m \in \{0,1\}^{\ell(n)}$ and $\text{Dec} : (K, c) \in \{0,1\}^n \times \{0,1\}^{\ell(n)} \rightarrow G(K) \oplus c \in \{0,1\}^{\ell(n)}$ is a one-time symmetric encryption scheme.

Proof: [Proposition 1] The proof follows an hybrid argument. Let $\mathcal{A}$ be an adversary against the one-time security of $\Pi$. Let $m_0, m_1$ denote the messages chosen by $\mathcal{A}$. Assuming $G$ is a pseudorandom generator, the distributions $\{G(K) \oplus m_0\}_{K \in \{0,1\}^n}$ and $\{R \oplus m_0\}_{R \in \{0,1\}^{\ell(n)}}$ are computationally indistinguishable. As $R$ is uniformly random, the distributions $\{R \oplus m_0\}_{R \in \{0,1\}^{\ell(n)}}$ and $\{R \oplus m_1\}_{R \in \{0,1\}^{\ell(n)}}$ are statistically indistinguishable (one-time pad). Finally, assuming $G$ is a pseudorandom generator, the distributions $\{R \oplus m_1\}_{R \in \{0,1\}^{\ell(n)}}$ and $\{G(K) \oplus m_1\}_{K \in \{0,1\}^n}$ are computationally indistinguishable.

The claim follows.

References