Build a provably secure PRF from PRG

The basic idea of pseudorandom functions F_k is c = (r, F_k(r) + m). These are easy to compute and computationally indistinguishable from a random function and there are $2^{n(2^n)}$ possible functions.

Definition of PRF :

Let f : {0,1}* x {0,1}* -> {0,1}* be an efficient, length-preserving, keyed function. We say that F is a pseudorandom function if for all probabilistic polynomial-time distinguishers D, there exists a negligible function negl such that:

$$|\Pr[D^{F_k(\cdot)}(1^n) = 1] - \Pr[D^{f(\cdot)}(1^n) = 1]| <= \mathrm{negl}(n),$$

Where k <- $\{0,1\}^n$ is chosen uniformly at random and f is chosen uniformly at random from the set of functions mapping n-bit strings to n-bit strings.

To construct PRF from PRG :

Let G be a pseudorandom generator with expansion factor l(n) = 2n. Denote by $G_0(k)$ the first half of G's output, and by $G_1(k)$ the second half of G's output. For every k belonging to $\{0,1)^n$, define the function $F_k : \{0,1\}^n \rightarrow \{0,1\}^n$ as :

$$F_k(x_1,x_2,...x_n) = G_{x\_n}(...(G_{x\_2}(G_{x\_1}(k)))...)$$

If G is a pseudorandom generator with expansion factor l(n) = 2n, then the above method gives a pseudorandom function.