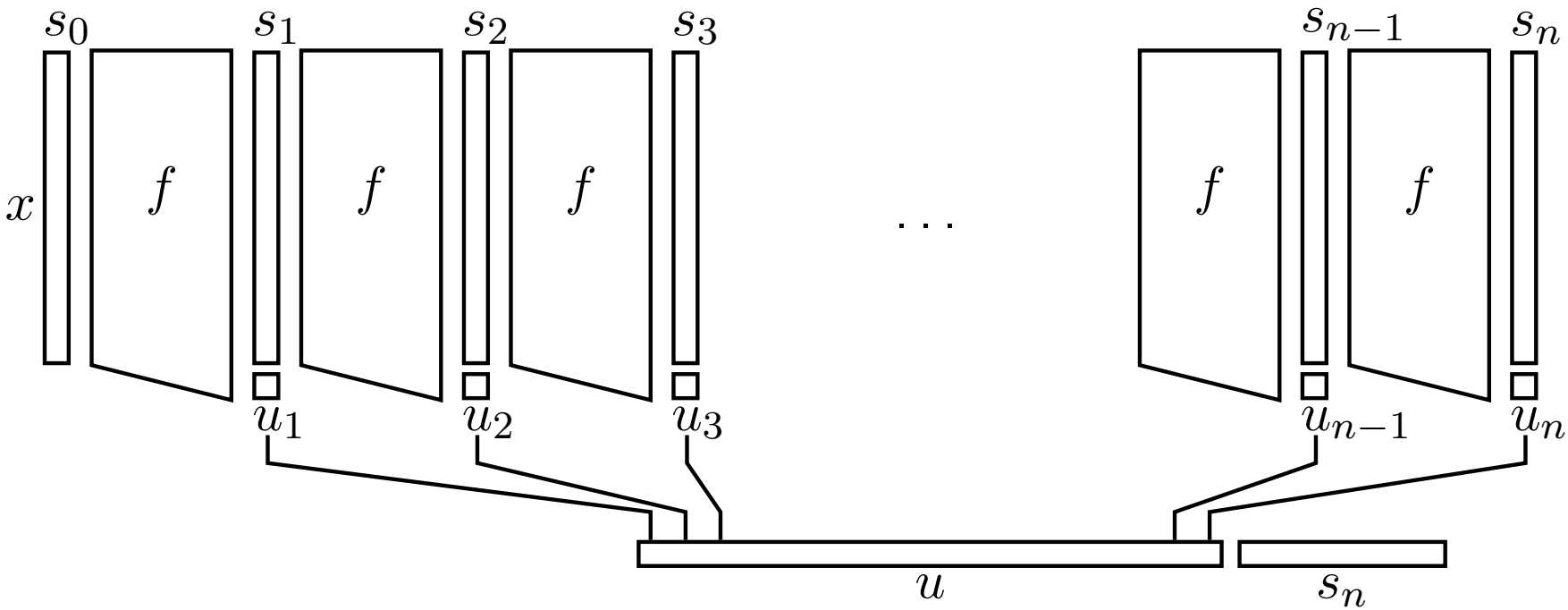
The art of cryptography: Heads and tails – Cryptographic random generation summer 2015

From short to long generators

Prof. Dr. Joachim von zur Gathen


s_0
s_1
s_2
s_3
s_{n-1}
s_n
x
f
f
f
$\ldots$
f
f
u_1
u_2
u_3
u_{n-1}
u_n
u
s_n

ALGORITHM. Long generator g from short generator f.

Input: Positive integers k and n, $f : \mathbb{B}^k \to \mathbb{B}^{k+1}$, and a seed $x \in \mathbb{B}^k$.

Output: A string $g(x) \in \mathbb{B}^{k+n}$.

1. $s_0 \leftarrow x$.
2. For i from 1 to n do step 3.
3. $\quad u_i s_i \leftarrow f(s_{i-1})$, where $u_i \in \mathbb{B}$ and $s_i \in \mathbb{B}^k$.
4. Return $g(x) = u_1 \cdots u_n s_n$.

We denote by $\mathsf{id}_j : \mathbb{B}^j \to \mathbb{B}^j$ the identity function and define for $i \geq 1$ functions f_i that leave the first $i-1$ bits unchanged and apply f to the last k bits, and their compositions:

$$f_i = \mathsf{id}_{\mathbb{B}_{i-1}} \times f : \begin{array}{ccc} \mathbb{B}^{k+i-1} & \longrightarrow & \mathbb{B}^{k+i}, \\ (x_1, \ldots, x_{k+i-1}) & \longmapsto & (x_1, \ldots, x_{i-1}, f(x_i, \ldots, x_{k+i-1})), \end{array}$$

$$g_i = f_i \circ \cdots \circ f_2 \circ f_1 : \quad \mathbb{B}^k \longrightarrow \mathbb{B}^{k+i}.$$

Thus $g_1 = f_1 = f$. We also set $g_0 = \mathsf{id}_k$ and $g = g_n$.

ALGORITHM. From long to short distinguishers.

Input: $x \in \mathbb{B}^{k+1}$.
Output: 0 or 1.

1. Choose $i \xleftarrow{\text{🎲}} \{1, \ldots, n\}$.
2. Choose $y \xleftarrow{\text{🎲}} U_{n-i}$.
3. Execute $\mathcal{A}$ on input $(y, h_i(x)) \in \mathbb{B}^{k+n}$ and return its output.

LEMMA. *Suppose that $\mathcal{A}$ is a (ϵ, s')-distinguisher between U_{k+n} and $g(U_k)$. Then $\mathcal{B}$ in Algorithm is a $(\epsilon/n, s' + nt)$-distinguisher between U_{k+1} and $f(U_k)$.*

We consider for $0 \leq i \leq n$ the hybrid random variable

$$Y_i = U_{n-i} \times g_i(U_k)$$

with values in $\mathbb{B}^{k+n}$. Thus $Y_n = g_n(U_k)$ and $Y_0 = U_{k+n}$ are the two random variables between which $\mathcal{A}$ distinguishes. For any $i \leq n$ we have

$$\begin{aligned} Y_i &= U_{n-i} \times g_i(U_k) = U_{n-i} \times h_i(f(U_k)) \text{ if } i \geq 0, \\ Y_{i-1} &= U_{n-i+1} \times g_{i-1}(U_k) \\ &= U_{n-i} \times U_1 \times g_{i-1}(U_k) = U_{n-i} \times h_i(U_{k+1}) \text{ if } i \geq 1. \end{aligned}$$

We have

$$\begin{aligned}
\text{prob}\{1 \xleftarrow{\text{🎲}} \mathcal{B}(f(U_k))\} =& \frac{1}{n} \sum_{1\le i\le n} \mathcal{E}_{\mathcal{A}}(U_{n-i} \times h_i(f(U_k))\} \\
=& \frac{1}{n} \sum_{1\le i\le n} \mathcal{E}_{\mathcal{A}}(Y_i) = \frac{1}{n} \sum_{1\le i\le n} e_i, \\
\text{prob}\{1 \xleftarrow{\text{🎲}} \mathcal{B}(U_{k+1})\} =& \frac{1}{n} \sum_{1\le i\le n} \mathcal{E}_{\mathcal{A}}(U_{n-i} \times h_i(U_{k+1})) \\
=& \frac{1}{n} \sum_{1\le i\le n} \mathcal{E}_{\mathcal{A}}(Y_{i-1})\} = \frac{1}{n} \sum_{1\le i\le n} e_{i-1}, \\
\mathcal{E}(\mathcal{B}(f(U_k))) - \mathcal{E}(\mathcal{B}(U_{k+1})) =& \frac{1}{n}\Big(\sum_{1\le i\le n} e_i - \sum_{1\le i\le n} e_{i-1}\Big) \\
=& \frac{1}{n}(e_n - e_0) \ge \frac{\epsilon}{n}.
\end{aligned}$$

THEOREM. *Let $f : \mathbb{B}^k \to \mathbb{B}^{k+1}$ be an (δ, s)-resilient generator that can be computed by a circuit of size t, let $n \geq 1$, and $g = g_n : \mathbb{B}^k \to \mathbb{B}^{k+n}$ as in Algorithm. Then g is an $(n\delta, s - nt)$-resilient generator, and can be computed with n applications of f.*

COROLLARY. *Let $f = (f_k)_{k\in\mathbb{N}}$ be a pseudorandom generator with $f_k \colon \mathbb{B}^k \longrightarrow \mathbb{B}^{k+1}$, and $p \in \mathbb{Z}[t]$ a positive polynomial. Then the above construction yields a pseudorandom generator $g = (g_k)_{k\in\mathbb{N}}$ with $g_k \colon \mathbb{B}^k \longrightarrow \mathbb{B}^{k+p(k)}$.*