CONSTRUCTING PRGs.

Last lecture: PRGs $\Rightarrow$ SKE with $|\mathcal{K}| < |\mathcal{M}|$.

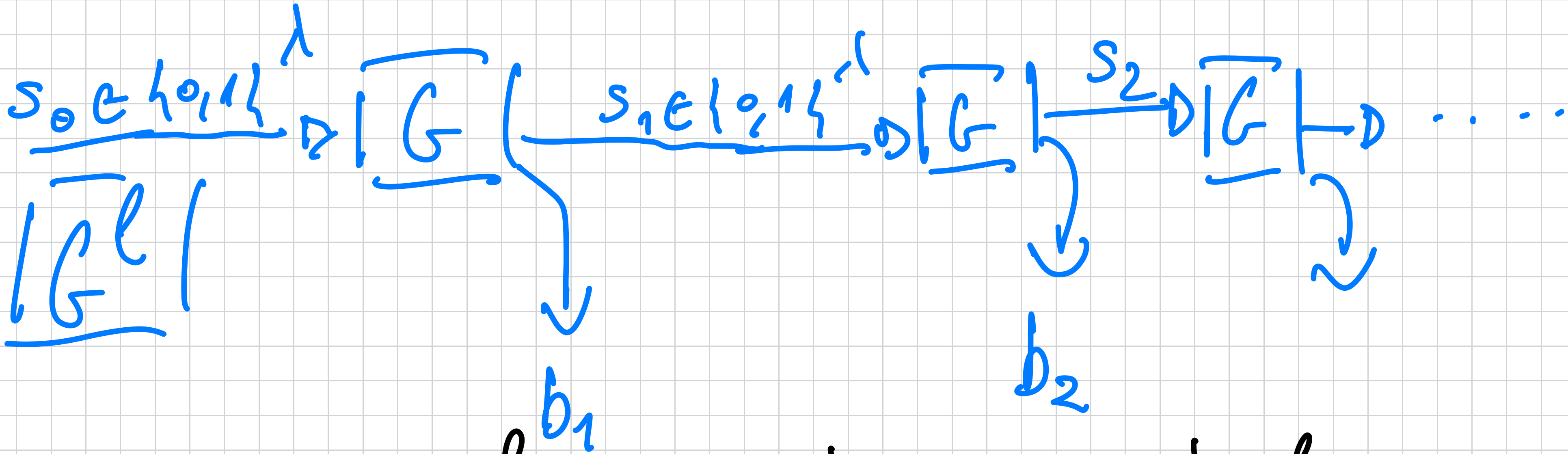
Today: How to construct PRGs?

We'll do it in two steps:

1) Assume we have secure $G : \{0,1\}^\lambda \to \{0,1\}^{\lambda+1}$. (i.e. $\ell(\lambda) = 1$ bit). Then amplify the stretch to $\ell(\lambda) = poly(\lambda)$.

2) Construct G with $\ell(\lambda) = 1$ or so.

Start with 1).



Formally; $G^\ell : \{0,1\}^\lambda \to \{0,1\}^{\lambda+\ell}$

- $s_0 \leftarrow U_\lambda$
- $\forall i \in [\ell]$, let $(s_i, b_i) = G(s_{i-1})$

  s.t. $s_i \in \{0,1\}^\lambda$ and $b_i \in \{0,1\}$.
- Output: $(b_1, b_2, \ldots, b_\ell, s_\ell)$

THM. The above G^{ℓ} is a PRG for any $\ell(\lambda) = poly(\lambda)$, assuming G is a PRG.

Proof. We use a technique called the HYBRID ARGUMENT.

We need to show $G^{\ell}(U_\lambda) \approx_c U_{\ell+\lambda}$. We can do this by defining hybrid distributions $H_0(\lambda), H_1(\lambda), \dots, H_\ell(\lambda)$ s.t.

(i) $H_0(\lambda) \equiv G^{\ell}(U_\lambda)$; $H_\ell(\lambda) \equiv U_{\lambda+\ell}$

(ii) $H_0(\lambda) \approx_c H_1(\lambda) \approx_c H_2(\lambda) \cdots \approx_c H_\ell(\lambda)$

Remark: Property (ii) implies $H_0(\lambda) \approx_c H_\ell(\lambda)$

as long as $\ell(\lambda) = \mathrm{poly}(\lambda)$ (follows by the triangle inequality).

The hybrids:

$$H_0(\lambda) \equiv G^{\ell}(\lambda) = (b_1, \dots, b_\ell, s_\ell)$$

$$b_1, \dots, b_i \leftarrow \{0,1\}$$

$$s_i \leftarrow U_\lambda$$

$$H_i(\lambda) \equiv$$

$$(b_{i+1}, \dots, b_\ell, s_\ell) = G^{\ell - i}(s_i)$$

$$H_\ell(\lambda) \equiv (b_1, \dots, b_\ell, s_\ell) \leftarrow U_{\lambda+\ell}$$

LEMMA. $\forall i \in [0, \ell - 1]: H_{i+1}(\lambda) \approx_c H_i(\lambda)$.

Proof. Fix i. Assume not: $\exists$ PPT A s.t.

$$\left| \Pr[A(z) = 1 : z \leftarrow H_{i+1}(\lambda)] - \Pr[A(z) = 1 : z \leftarrow H_i(\lambda)] \right| \geq \frac{1}{poly(\lambda)}$$

We construct PPT B attacking G:

$$A \xleftarrow{\text{START}} B \xleftarrow{z^* \in \{0,1\}^{\lambda+1}} C_{PRG}$$

$$b_1, \ldots, b_i \leftarrow U_1$$

$$\xleftarrow{z = (b_1, \ldots, b_\ell, s_\ell)} \quad z^* = (s_{i+1}, b_{i+1})$$

$$(b_{i+2}, \ldots, b_\ell, s_\ell) = G^{\ell - i - 1}(s_{i+1})$$

$$z^* \begin{cases} G(U_\lambda) \\ U_{\lambda+1} \end{cases}$$

$$\xrightarrow{b' \in \{0,1\}}\ ) \qquad \xrightarrow{b' \in \{0,1\}}\ )$$

I claim that the distribution of z is s.t.:

- If $z^* \equiv G(U_\lambda)$, $z \leftarrow H_i'(\lambda)$
- If $z^* \equiv U_{\ell+1}$, $z \leftarrow H_{i+1}(\lambda)$

Now:

$$\Pr[\mathcal{B}(z^*)=1 : z^* \leftarrow G(U_\lambda)]$$

$$= \Pr[\mathcal{A}(z)=1 : z \leftarrow H_i'(\lambda)]$$

$$\Pr[\mathcal{B}(z^*)=1 : z^* \leftarrow U_{\lambda+1}]$$

$$= \Pr[\mathcal{A}(z)=1 : z \leftarrow H_{i+1}(\lambda)]$$

$$\Rightarrow \Big| \Pr[\mathcal{B}(z^*)=1 : z^* \leftarrow G(U_\lambda)]$$

$$- \Pr[\mathcal{B}(z^*)=1 : z^* \leftarrow U_{\lambda+1}] \Big| \geq \frac{1}{poly(\lambda)}$$

$$\blacksquare$$

EXERCISE If $X \approx_c Y$, $Y \approx_c Z$ then $X \approx_c Z$.

For every PPT A:

$$\Big| \Pr[A(u)=1 : u \leftarrow X] - \Pr[A(u)=1 : u \leftarrow Z] \Big|$$

$$\Big| \Pr[A(u)=1 : u \leftarrow X] - \Pr[A(u)=1 : u \leftarrow Y]$$

$$+ \Pr[A(m)=1 : m \leftarrow Y] - \Pr[A(m)=1 : m \leftarrow Z] \Big|$$

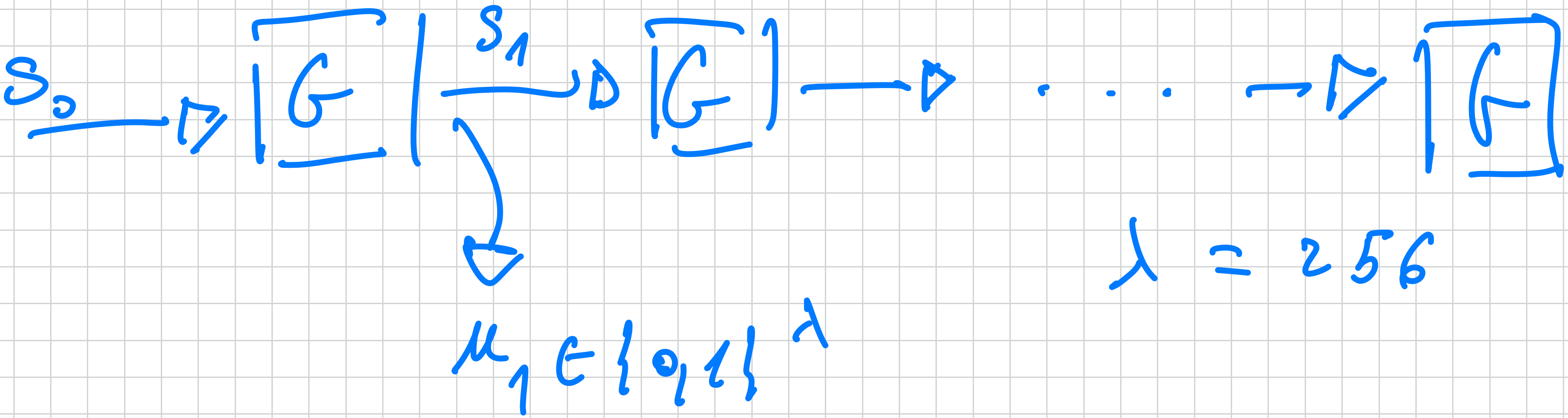
$$\leq \Big| \Pr[A(m)=1 : m \leftarrow X] - \Pr[A(m)=1 : m \leftarrow Y] \Big|$$

$$+ \Big| \Pr[A(m)=1 : m \leftarrow Y] - \Pr[A(m)=1 : m \leftarrow Z] \Big|$$

$$\leq \varepsilon_1(\lambda) + \varepsilon_2(\lambda) \leq negl(\lambda)$$

$$\varepsilon_1(\lambda), \varepsilon_2(\lambda) = negl(\lambda)$$

Real-world PRGs (e.g. /dev/rand
/dev/urand)



More on details:

- How to generate s_0? Randomness extractors. Theory: leftover hash lemma. Practice: AES.
- Which G? Theory: we can get one

from ANY OWF f or assuming hardness of FACTORING, DISCRETE LOG, LWE, ...

PRACTICE : AES.

- Not yet the final design. Because of the internal state is compromised all the future outputs are predictable.

The real-world construction keep refreshing the state : If state is s_i', $Ext(X)$

$$s_N' = s_N \oplus Ext(X)$$

Made with Goodnotes

How to construct G in theory:

THM If OWFs exist, then so do PRGs with $\ell(\lambda) = 1$.

The proof has to do with following question: What info about x is hidden given $f(x)$?

$$\begin{array}{ccc} \mathcal{A} & & \mathcal{C} \\ & \xleftarrow{\quad z = f(x) \quad} & x \leftarrow \mathcal{X} \\ & \xrightarrow{\quad x' \quad} & \\ & & \text{win}: f(x') = z \end{array}$$

Non-trivial: If f is OWF, then so is $f'(x) = 0 \| f(x)$. Thus is not a PRG!

Ex. Prove it.

Also: If f is OWF, then so is $f'(x) = x[1] \| f(x)$

$x[1]$ = 1st bit of x

Ex. Prove it.

HARD-CORE BIT: Is a PREDICATE $h: \mathcal{X} \to \{0,1\}$ s.t. given $f(x)$ it

is hard to compute $h(x)$ (i.e.
$(h(x), f(x)) \approx_c (U_1, f(x))$).

FACT. Every f commits on h.

$$G(s) = f(s) \| h(s)$$

PRG assuming f is ONE-WAY
PERMUTATION

CPA - SECURITY

Want : Build SKE (Enc, Dec) s.t.

- $|K| << |m|$
- Can encrypt more than 1 msg -

Recall : $Enc(K, m) = G(K) \oplus m$ this achieves $|K| << |m|$. However, if we reuse the key :

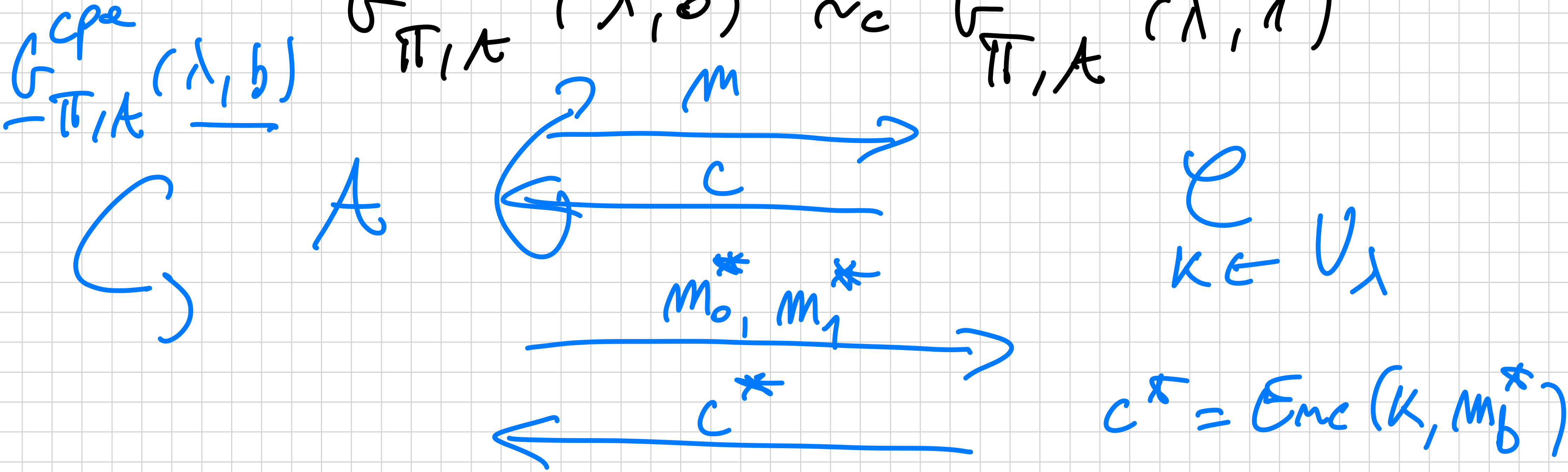
$$c_1 = G(K) \oplus m_1 \quad ; \quad c_2 = G(K) \oplus m_2$$

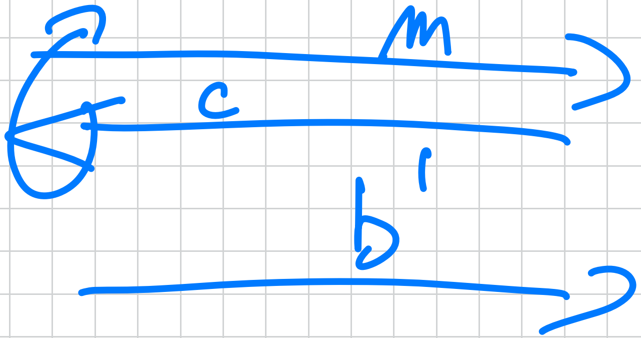
$$c_1 \oplus c_2 = m_1 \oplus m_2$$

If A knows a single pair (m_1, c_1) future plaintexts are exposed forever.

DEF (CPA SECURITY) We say that $(Enc, Dec) = \Pi$ is CPA secure if:

$$G^{cpa}_{\Pi,A}(\lambda, 0) \approx_c G^{cpa}_{\Pi,A}(\lambda, 1)$$





$c = Enc(k, m)$

Ex. The above is impossible if Enc is DETERMINISTIC!