Symmetric Crypto   Lec 2.   (Apr 4 '24)

Outline:

- Recap
- Game based Defn.
- PRG extension (BM '84)
- Hybrid Arguments
- PRFs from PRG (GGM '84)
- Wrap up

RECAP:

$$\underbrace{\text{OWF} \xrightarrow{\text{Goldreich-Levin}} \underset{\text{(1-bit stretch)}}{\text{PRG}}}_{\text{Last class}} \overbrace{\xrightarrow{\text{Blum-Micali}} \underset{(\text{poly}(\lambda)\text{-stretch})}{\text{PRG}} \xrightarrow{\text{GGM}} \text{PRF}}^{\text{Today}}$$

$$\text{PRF} \xrightarrow{\text{LR}} \text{PRP/Block Cipher}$$

Def: A PRG $G : S \to R$ is a deterministic, poly-time algorithm that given a seed $s \in S$ (seed space) as input, outputs $r \in R$ (output space).

G is <u>secure</u> if for all <u>efficient</u> adversaries A,

$$\left| \Pr_{A,s}\left[A(r) = 1 : \begin{array}{l} s \leftarrow\$ S \\ r \leftarrow G(s) \end{array} \right] - \Pr_{A,r}\left[A(r) = 1 : r \leftarrow\$ R \right] \right| \leq \mathrm{negl}(\lambda)$$

Here, the probability space is over random choice of s, r, and randomness of A.

<u>Last Lecture</u>: Secure PRG $G : \{0,1\}^n \to \{0,1\}^{n+1}$ with 1-bit stretch from a OWF using Hard core bits. (GL)

<u>Today</u>: Given a secure PRG:

$G : \{0,1\}^n \to \{0,1\}^{n+1}$, we build another PRG, $G' : \{0,1\}^n \to \{0,1\}^{\ell(n)}$, where ℓ is a poly. $(\ell(n) > n+1)$

Game based definition of PRG security :-

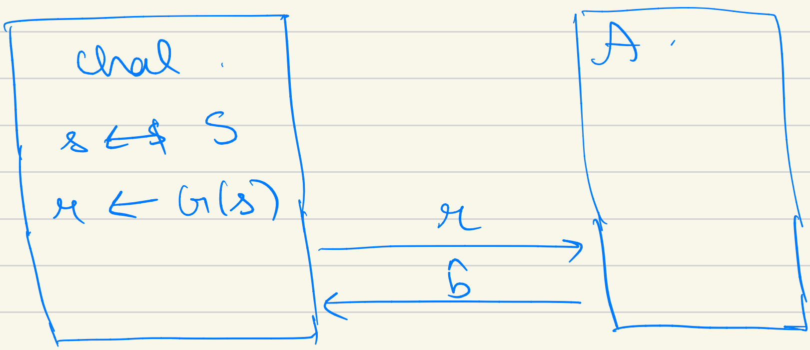
(easier to work with)

In the above definition, the adversary A needs to act as a distinguisher b/w two distributions :

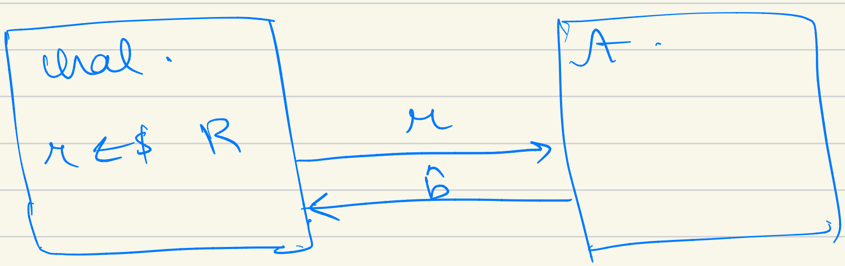
$$D_0 : \left\{ \begin{array}{l} s \xleftarrow{\$} S \\ r \leftarrow G(s) \end{array} \right\} \quad \text{vs.} \quad \left\{ r \xleftarrow{\$} R \right\} : D_1$$

We can reframe this as a game b/w A and a challenger:-

Experiment 0 : $\approx D_0$



Experiment 1 : $\approx D_1$



Let W_0: Event that A outputs 1 in Exp 0.
W_1: " " " " " " Exp 1.

Then, we define:

PRGAdv[A, G] = Advantage of A in the PRG security game for G

$$= |\Pr(W_0) - \Pr(W_1)|.$$

where the probability space is over the random choices of the Challenger and A.

G is secure if, $\forall$ efficient adversaries A,

$$\text{PRGAdv}[A, G] \leq \text{negl}(\lambda).$$

↳ Identical to the distribution-based defn. above.

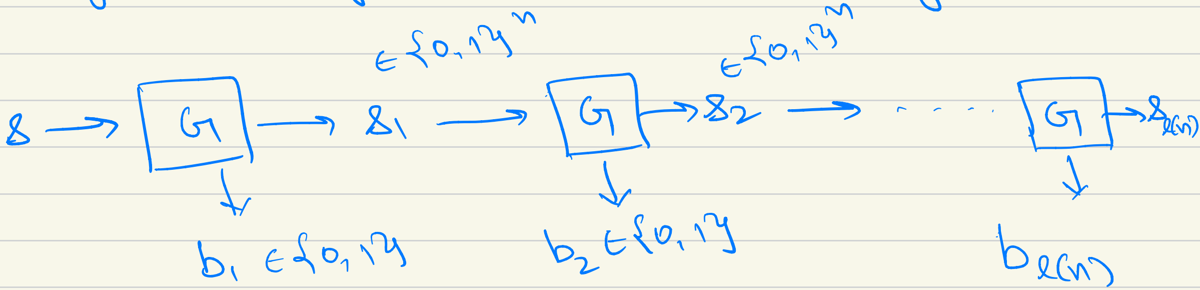
PRG Extension :- (Blum-Micali '84)

Turing Turing

Let $G : \{0,1\}^n \to \{0,1\}^{n+1}$ be a secure PRG.
We will construct $G' : \{0,1\}^n \to \{0,1\}^{\ell(n)}$

$G: \{0,1\}^n \longrightarrow \underbrace{\{0,1\}^n}_{ES} \times \underbrace{\{0,1\}}_{1 \text{ extra bit}}$

* We can sequentially compose G, to get many random-looking bits.



Output $\underbrace{(b_1, b_2, \ldots, b_{\ell(n)})}$ $\in \{0,1\}^{\ell(n)}$

Formally,

$G'(s \in \{0,1\}^n)$:

$s_0 = s$

for each $i \in \{1, 2, \ldots, \ell(n)\}$:

$(s_i, b_i) \leftarrow G(s_{i-1})$

O/P $(b_1, \ldots b_{\ell(n)})$.

Is G' a secure PRG?

1.) Efficient? Let $t(n)$ be the runtime of G. Then, G' runtime is: $\ell(n) * t(n) + O(\ell(n))$: poly. ✓

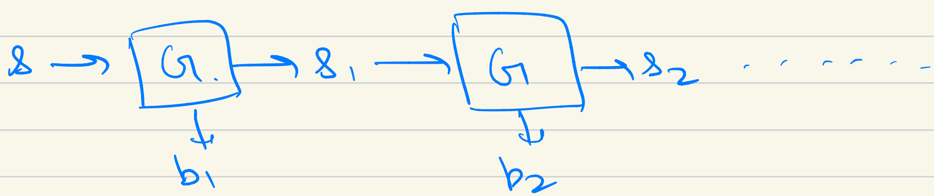
2.) Secure ?   Informal : G secure $\Rightarrow$ G' secure.

Thm. For every adv A playing the PRG game for G', $\exists$ an adversary B that plays the PRG game for G, s.t.

$$\underbrace{\mathrm{PRGAdv}[A, G']}_{\text{negl}} = \underbrace{\ell(n)}_{\text{poly}} \cdot \underbrace{\mathrm{PRGAdv}[B, G]}_{\text{negl if } G \text{ is secure}}.$$

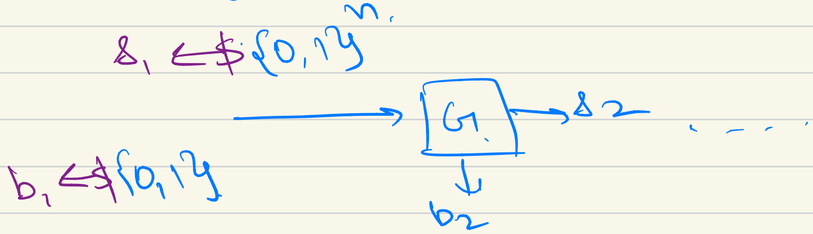
$\Leftarrow$

Proof : Informally :-



(i) For random s, the o/p of G looks random, so, we can replace (s_1, b_1) by random elements :-

$\approx_c$

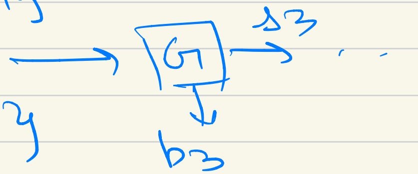


(ii) now, s_1 is random, so we can

replace (s_2, b_2) by random elements:-

$s_2 \xleftarrow{\$} \{0,1\}^n$.

$\approx_c$



$b_1 \xleftarrow{\$} \{0,1\}$, $b_2 \xleftarrow{\$} \{0,1\}$

and so on. (we can do this for each PRG on the chain.)

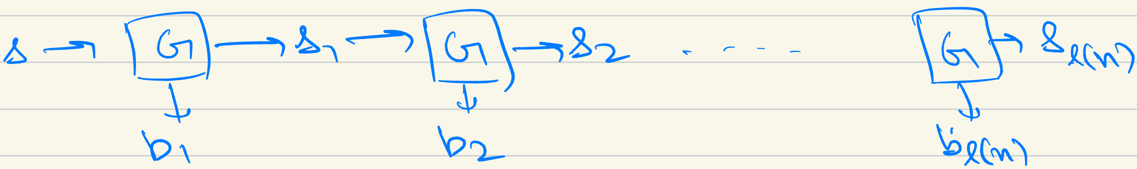
HOW to FORMALIZE the above intuition?

<u>Hybrid Arguments</u>:

Recall we defined 2 games, Exp0, Exp1 in the PRG security defn.

<u>Exp 0</u>: $\equiv H_0$

Chal samples $s \xleftarrow{\$} \{0,1\}^n$ and gives $r = G'(s)$ to A:



send $r = (b_1, b_2, \ldots, b_{\ell(n)})$ to A.

We'll now define a sequence of Hybrid games, $H_0 = \text{Exp}_0, H_1, \ldots H_{\ell(n)}$,

where we'll slightly change the Challenger's behavior in each hybrid.

H_1 : Hybrid 1 : Chal samples $s_1, b_1 \leftarrow\$$.

$$\begin{tikzcd} s_1 \leftarrow\$ \{0,1\}^n \arrow[r] & \boxed{G} \arrow[r] \arrow[d] & s_2 \arrow[r] & \boxed{G} \arrow[r] \arrow[d] & \cdots & \boxed{G} \arrow[r] \arrow[d] & s_{\ell(n)} \\ & b_2 & & b_3 & & b_{\ell(n)} & \end{tikzcd}$$

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

give $\mu = (b_1, b_2, \ldots, b_{\ell(n)})$ to A.

(Informally, no adv should be able to distinguish b/w H_0, H_1 due to security of PRG G.)

⋮

H_j : Hybrid j :

$$\begin{tikzcd} s_j \leftarrow\$ \{0,1\}^n \arrow[r] & \boxed{G} \arrow[d] & \cdots & \boxed{G} \arrow[r] \arrow[d] & s_{\ell(n)} \\ & {} & & {} & \end{tikzcd}$$

$(b_1 \leftarrow\$, b_2 \leftarrow\$ \ldots, b_j \leftarrow\$ \{0,1\}, b_{j+1}, \ldots, b_{\ell(n)})$

⋮

$H_{\ell(n)}$:

$(b_1, b_2 - - - - - - - - - -, b_{\ell(n)} \leftarrow\$ \{0,1\})$

$\equiv$ Exp 1 in PRG security defn.

For $i \in \{0, 1, \ldots, \ell(n)\}$, define p_i as the probability that A outputs 1 in Hybrid game H_i.

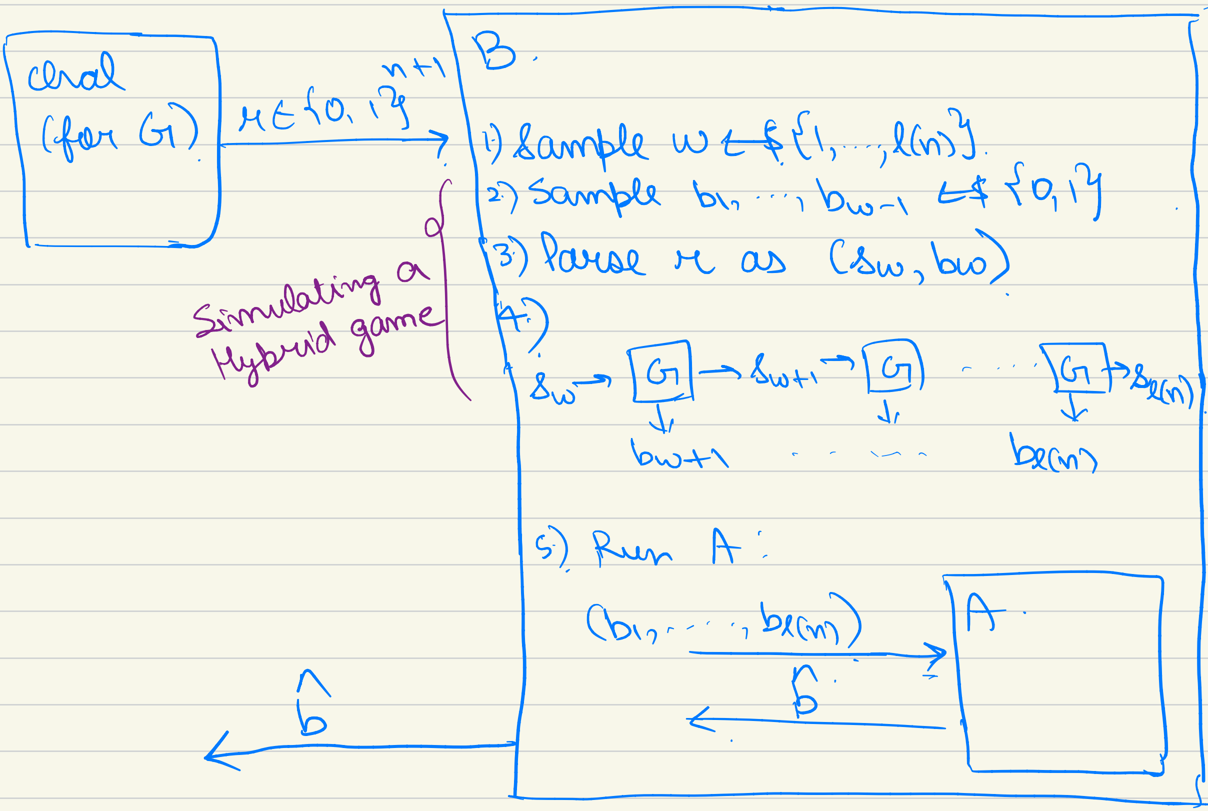
By defn.,

$$\text{PRGAdv}[A, G'] = \left|\Pr[W_0] - \Pr[W_1]\right|$$

1 in Exp0   1 in Exp1

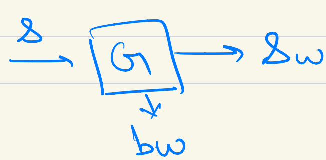
$$= |p_0 - p_{\ell(n)}|$$

Now, we'll construct the adv B playing the PRG security game for G.



In the game that B is playing,

Exp0 : $r = G(s)$ for random seed s.



i.e. $b_1, \ldots, b_{w-1}$: random, but

$b_w, \ldots, b_{\ell(n)}$: generated as in H_{w-1}.

$\Rightarrow$ this is Hybrid H_{w-1} from A's perspective.

Exp 1 : $r \xleftarrow{\$} R$. $\quad r = (\overset{\$}{s_w}, \overset{\$}{b_w})$

$\Rightarrow$ $b_1, \ldots, b_w$: random.

$\Rightarrow$ B 'identically simulates' hybrid H_w to A.

This means,

$$\Pr(\underbrace{W_{0,B}}_{\text{event that B outputs 1 in Exp}_0} \mid w = j) = \overbrace{P_{j-1}}^{\Pr \text{ A outputs 1 in } H_{j-1}} \quad \text{and}$$

* B outputs whatever A outputs
* B is simulating H_{j-1} in the event $W_{0B} / w = j$, so, A outputs 1 with prob. p_{j-1}.

$$\Pr(W_{1B} \mid w = j) = P_j \quad \text{for all } j \ldots$$

So,

$$\text{PRGAdv}[B, G] = |\Pr[W_{0B}] - \Pr[W_{1B}]|.$$

By total probability :-

$$= \left| \sum_{j=1}^{\ell(n)} \Pr(W_{0B} \mid w=j) * \overbrace{\Pr(w=j)}^{\frac{1}{\ell(n)}} - \sum_{j=1}^{\ell(n)} \Pr(W_{1B} \mid w=j) * \Pr(w=j) \right|$$

Since w is sampled uniformly from $\{1, \ldots, \ell(n)\}$,

$$= \frac{1}{\ell(n)} \left| \begin{array}{l} p_0 + \cancel{p_1} + \cancel{\ldots} \ldots + \cancel{p_{\ell(n)-1}} \\ - \cancel{p_1} - \cancel{p_2} \ldots - \cancel{p_{\ell(n)-1}} - p_{\ell(n)} \end{array} \right|$$

$$= \frac{1}{\ell(n)} \left| p_0 - p_{\ell(n)} \right|$$

$$= \frac{1}{\ell(n)} \cdot \text{PRGAdv}[A, G'].$$

i.e. $\text{PRGAdv}[A, G'] = \ell(n) \cdot \text{PRGAdv}[B, G]$.

so, if G is secure, meaning $\text{PRGAdv}[B, G]$ is $\text{negl}(\lambda)$ $\forall B$,
then, G' must also be secure
bc $\ell(n) \cdot \text{negl}(\lambda)$ is $\text{negl}(\lambda)$.

Hence, G' is a secure PRG.

PRFs : (pseudo random functions)

PRF $F: K \times X \to Y$: deterministic, efficient algorithm.

Key space   Input space   Output space.

Informally, for a random key k, $F(k, \cdot)$ should look like a random function from X to Y.

$Func[X, Y]$ = space of all functions from X to Y.

e.g. if $K = \{0,1\}^{128} = X = Y$,

$\#$ Keys $= 2^{128} = \#$ PRFs.

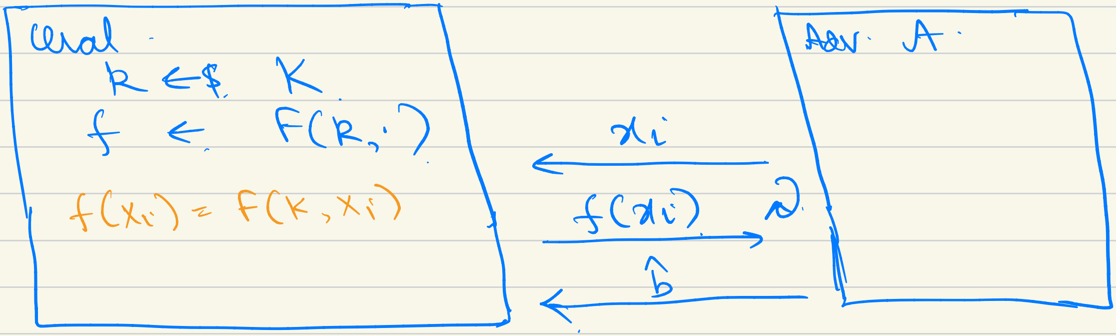
But $\#$ functions in $Func[X, Y] = |Y|^{|X|}$

$= (2^{128})^{2^{128}}$.

i.e. $\underbrace{Func[X, Y]}$ $\gg$ $\underbrace{|K|}$.

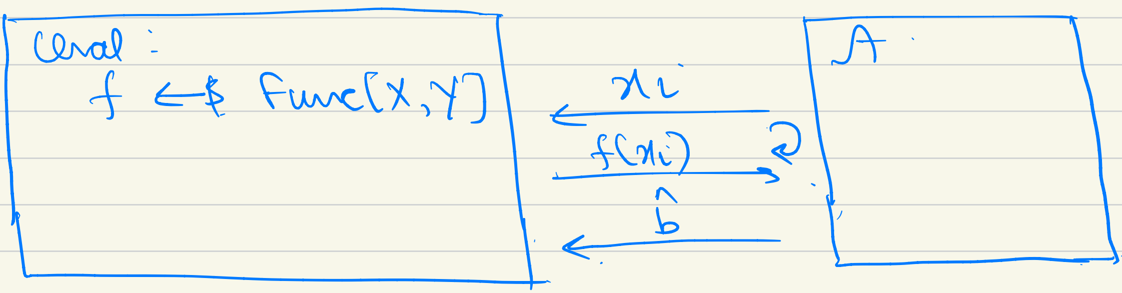
PRF security game :

Exp 0 :



Adv: can make poly. # queries, on arbitrary x_i.

Exp 1 :



Let W_b : Event that A outputs 1 in Exp b.

Advantage of A w.r.t. PRF F :

$$\text{PRFAdv}[A, F] = \left| \Pr[W_0] - \Pr[W_1] \right|$$

A is called a Q-query adversary if it makes upto Q queries to the chal.

A PRF F is secure if $\forall$ efficient adversaries $\mathcal{A}$,

$$\text{PRFAdv}[\mathcal{A}, F] \leq \text{negl}(\lambda).$$

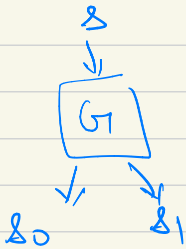
PRF from PRG. (Goldreich, Goldwasser, Micali '84)

Turing! Turing!

Given a PRG $G: S \to S \times S$, we can construct a PRF:

"length - doubling"

visualize G as:



lets say,

$s_0 = G_0(s)$

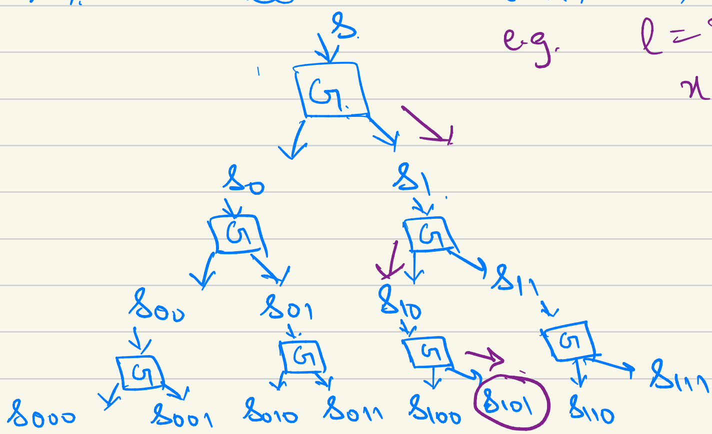
$s_1 = G_1(s)$.

we'll make PRF $F: \underbrace{S}_{} \times \underbrace{\{0,1\}^{\ell}}_{} \to S$ as follows:-

Key space $= S$. Bit strings, lets say $\ell = \lambda$.

$F(\underline{s}, x)$: let $x = (x_1, \ldots, x_\ell)$:

e.g. $\ell = 3$ and $x = \{1, 0, 1\}$



i.e. $F(s, 101) = s_{101}$

$$= G_1(G_0(G_1(s)))$$
$$(x_3 \quad x_2 \quad x_1)$$

* For $x = x_1, \ldots, x_\ell$, traverse the path in the above tree of evaluations

Formally,
$F(s, (x_1 \ldots x_\ell))$:

$t \leftarrow s$
for i in $\{1, \ldots, \ell\}$:
$\quad t \leftarrow G_{x_i}(t)$
o/p t.

Is F a secure PRF?

1.) Efficiency : ℓ evals of G. ✓
$= \text{poly}(\lambda)$

2.) security :

Thm: For every Q-query PRF adv. A, we can construct a PRG adv B, s.t.

$$\text{PRFAdv}[A, F] = \ell \cdot Q \cdot \text{PRGAdv}[B, G].$$

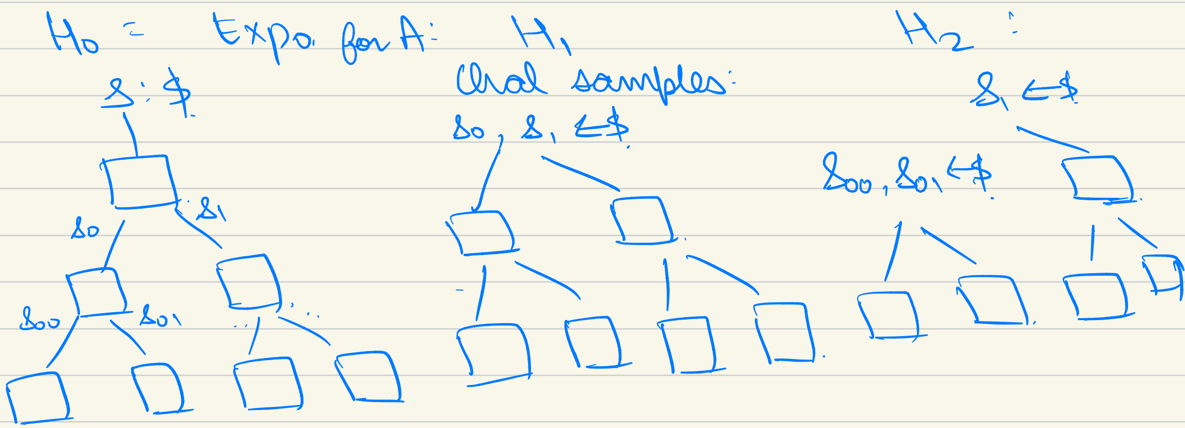
i.e. G is a secure PRG. $\Rightarrow$ F is a secure PRF.

Proof Sketch:

Given A: an adv. for the PRF game for F, we'll construct B: an adv. for the PRG game for G.

We'll use the Hybrid argument!

Naively, we could replace each PRG o/p by random, one-by-one:



note, there are 2^{i-1} PRGs on level i of tree.

$\Rightarrow$ # Hybrids: $1 + 2 + 2^2 + \dots + 2^{\ell-1}$

$\Rightarrow 2^{\ell} - 1$. : This is a problem...

Now, we'll construct $\mathcal{B}$, an adversary for the PRG game against G:

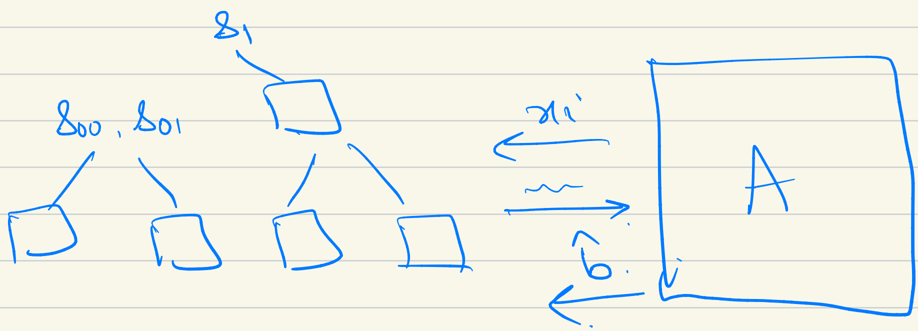
Chal (for G) $\xrightarrow{r}$

$\mathcal{B}$:

sample $w \xleftarrow{\$} \{1, 2, \ldots, 2^{\ell}-1\}$

e.g. $w = 2$: sample $s_1 \xleftarrow{\$} S$.

use r as (s_{00}, s_{01}):



use this tree to answer A's queries on $x_1, x_2, \ldots$

Output A's output.

$\hat{b}$

Simulate a Hybrid...

Analysis of B's advantage:-

By similar argument as that for PRG_{+poly} construction,

$$\text{PRGAdv}[B, G] = \left| \Pr(\overset{\overset{\text{1 in Exp0}}{\uparrow}}{W_{0B}}) - \Pr(\overset{\overset{\text{1 in Exp1}}{\uparrow}}{W_{1B}}) \right|$$

$$\left| \sum_{j=1}^{2^\ell - 1} \Pr(W_{0B} \mid w = j) * \Pr(w = j) - \sum_{j=1}^{2^\ell - 1} \Pr(W_{1B} \mid w = j) * \underbrace{\Pr(w = j)}_{\frac{1}{2^\ell - 1}} \right|$$

(skipped in class :)

Also, In Exp 0 of B,

B identically simulates H_{j-1}

conditioned on $w = j$.

In Exp 1 for B, it identically simulates H_j, conditioned on $w = j$.

Let p_j : Probability that A outputs 1 in H_j.

Then,

$$\text{PRGAdv}[B, G] = \frac{1}{2^\ell - 1} \left| \begin{array}{l} p_0 + p_1 \quad \ldots + p_{2^\ell - 2} \\ \quad - p_1 \quad \ldots \quad - p_{2^\ell - 1} \end{array} \right|$$

$$= \frac{1}{\underset{\nwarrow\ 2^\lambda - 1}{2^\ell - 1}} \text{PRFAdv}[A, F].$$

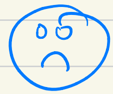
Issue: Recall, $l = \lambda$.

Even if $\text{PRFAdv}[A, F]$ is non-negligible,

B's advantage is still negligible!

$\Rightarrow$ B does NOT break G's security.

$\Rightarrow$ This proves NOTHING about F's security . . .



$\boxed{\text{SOLN}}$: Note, $\mathcal{A}$ is Q-query bounded, where $Q \stackrel{?}{=} \text{poly}(\lambda)$.

i.e. $\mathcal{B}$ only needs to simulate PRGs in the paths of these Q queries.

$\Rightarrow$ There will be max Q such PRGs in each level.

$\Rightarrow$ Just need $l \cdot Q$ hybrids !!

Full proof in book (Sec. 4.6)

Symmetric Crypto - Summary:

Next class

$$\begin{tikzcd}[column sep=large]
\text{OWF} \arrow[r, bend left, "\text{GL}"] & \text{PRG}_{+1} \arrow[l, bend left, "\text{By defn.}"] \arrow[r, bend left, "\text{BM84}"] & \text{PRG}_{+\text{poly}} \arrow[l, bend left, "\text{Truncate}"] \arrow[r, bend left, "\text{GGM84}"] & \text{PRF} \arrow[l, bend left, "\text{Deterministic Counter Mode.}"] \arrow[r, bend left, "\text{LR88}"] & \text{PRP/Block Cipher} \arrow[l, bend left, "\text{Switching Lemma}"]
\end{tikzcd}$$

Block Cipher: $\downarrow$ PRF but a permutation