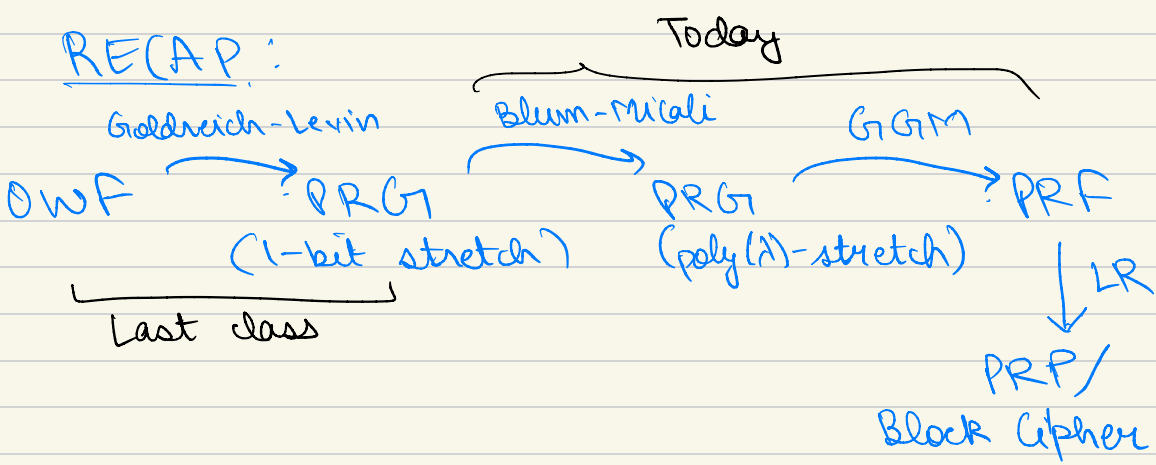
Symmetric Crypto Lec 2 (Apr 1, 2026)

Outline:

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

RECAP:



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,

$$\left| \Pr_{A,s}\left[A(r) = 1 : \begin{matrix} s \xleftarrow{\$} S \\ r \leftarrow G(s) \end{matrix} \right] - \Pr_{A,r}\left[A(r) = 1 : r \xleftarrow{\$} R \right] \right| \leq \text{negl}(n)$$

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)

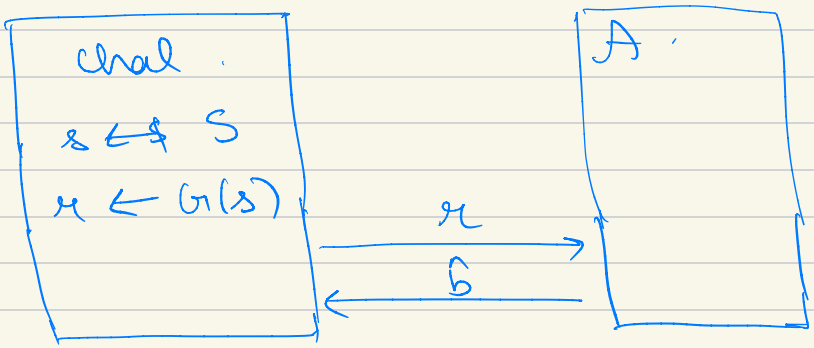
In the above definition, the adversary A needs to act as a distinguisher b/w two distributions :

$$\{ s \leftarrow\$ S ; r \leftarrow G(s) \} \text{ vs. } \{ r \leftarrow\$ R \} : D$$

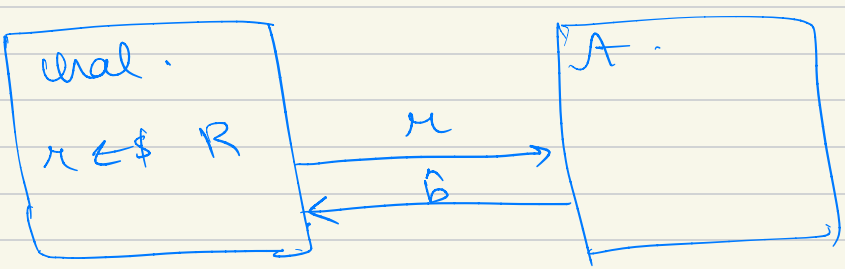
D_0 : D_1

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, (i.e PPT)

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

↳ Identical to the distribution-based defn. above.

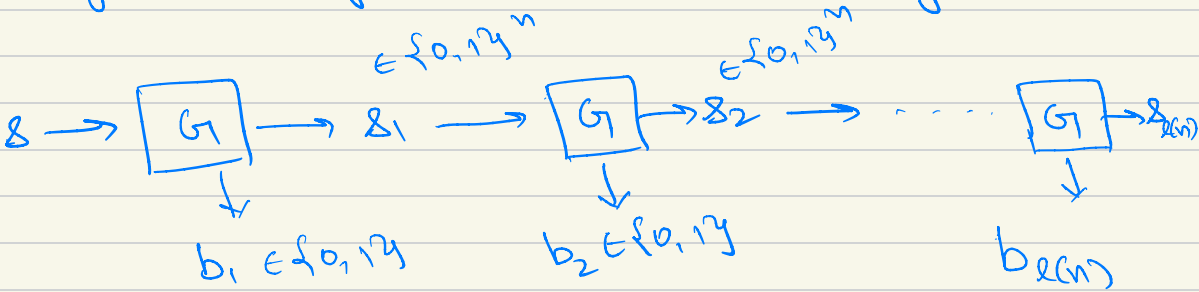
PRG Extension :- (Blum-Micali '84) (Yao Yao)

Let $G : \{0,1\}^n \to \{0,1\}^{n+1}$ be a secure PRG.

We will construct $G' : \{0,1\}^n \to \{0,1\}^{l(n)}$

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

* We can sequentially compose G_1 to get many random-looking bits.



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

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. ✓

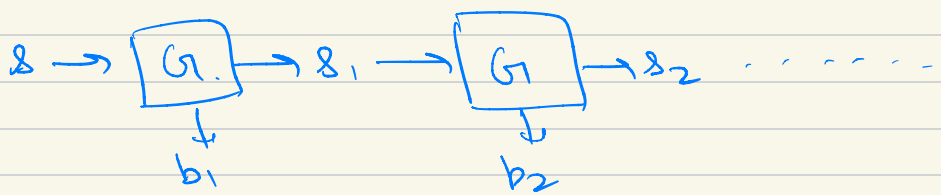
2.) Secure ? Informal : G_1 secure $\Rightarrow$ G_1' secure.

Thm. For every adv A playing the PRG game for G_1', we can construct adv B that plays the PRG game for G_1, s.t.

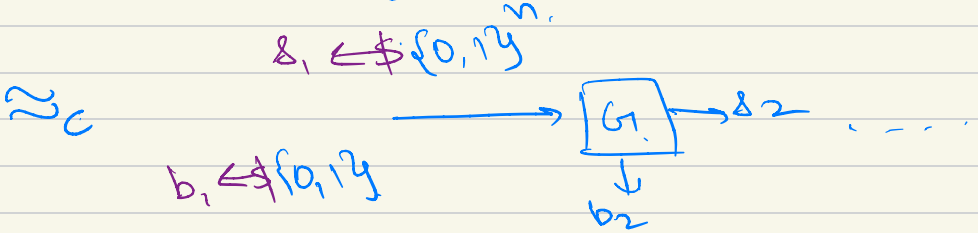
$$\underbrace{PRGAdv[A, G_1']}_{\text{negl.}} = \underbrace{l(n)}_{\text{poly}} \cdot \underbrace{PRGAdv[B, G_1]}_{\text{negl if } G_1 \text{ is secure}}.$$

$\leq$

Proof : Informally :-



(i) For random s, the o/p of G_1 looks random, so, we can replace (s_1, b_1) by random elements :-



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

replace (s_2, b_2) by random elements:

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

$\approx_c$ $\longrightarrow$ [G] $\rightarrow s_3 \ldots$

$b_1 \xleftarrow{\$} \{0,1\}$, $b_2 \xleftarrow{\$} \{0,1\}$ $\quad$ [G] $\rightarrow b_3$

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

HOW to FORMALIZE the above intuition?

Hybrid Arguments:

Recall, we defined 2 games in the PRG security defn:

Exp 0:

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

$s \rightarrow$ [G] $\rightarrow s_1 \rightarrow$ [G] $\rightarrow s_2 \cdots \rightarrow$ [G] $\rightarrow s_{\ell(n)}$

b_1 $\quad$ b_2 $\quad$ $b_{\ell(n)}$

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

and Exp 1:

Oral samples $r \leftarrow\$ \{0,1\}^{\ell(n)}$, i.e.

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

Hybrids: Something in between Exp 0 and Exp 1 ... 'Hybrid' of the two!

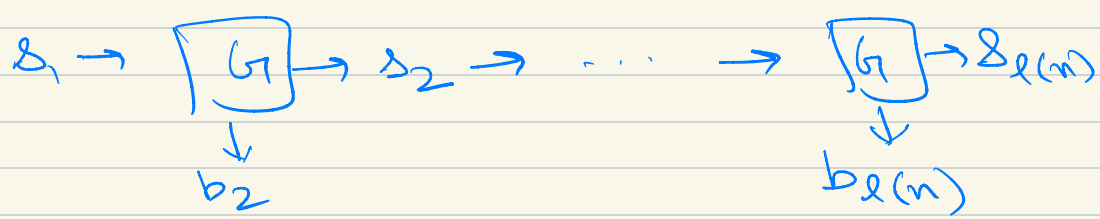
Exp 0 = Hybrid H_0

Hybrid H_1:

Instead of sampling $s \leftarrow\$ \{0,1\}^n$, (as in H_0)

Oral samples $s_1 \leftarrow\$ \{0,1\}^n$ and $b_1 \leftarrow\$ \{0,1\}$

Then, computes the rest as in H0:



Oral then gives $(b_1, b_2, \ldots, b_{\ell(n)})$ to A

b_1: Random, like Exp 1

$b_2, \ldots, b_{\ell(n)}$: based on PRG chain, like H_0

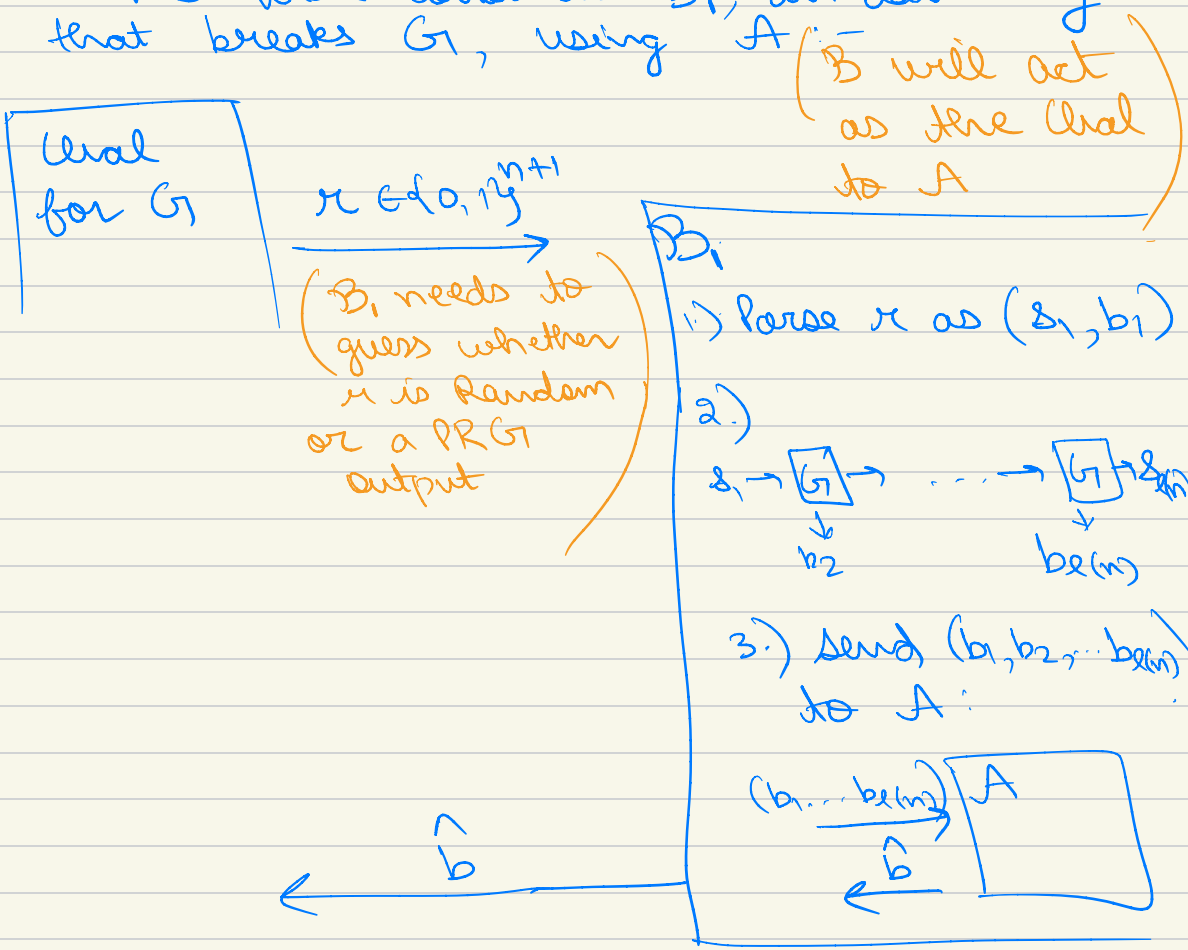
'Hybrid'

i.e. $Pr(A \text{ outputs } 1 \text{ in } H_0)$ is different from $Pr(A \text{ outputs } 1 \text{ in } H_1)$

Claim: If A can distinguish b/w H_0 and H_1, we can break PRG security of G_1

Proof:

We will construct B_1, an adversary that breaks G_1, using A.



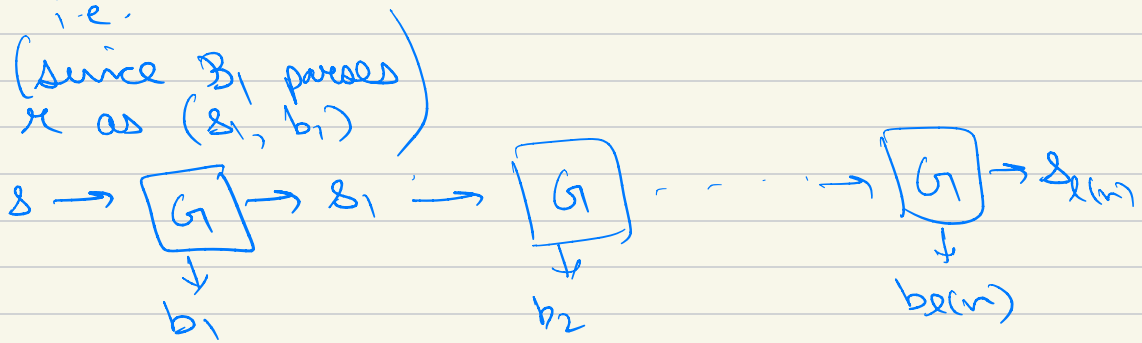
Let p_0 = Probability that A outputs 1 in Hybrid H_0

p_1 = " " " " in H_1.

Now, in the game B_1 is playing,

Exp 0: $r = G_1(s)$ for some s.

i.e.
(since B_1 passes r as (s_1, b_1))



The string $(b_1, \cdots b_{\ell(n)})$ sent to A by B_1 in this case is just like H_0.

Exp 1: $r \xleftarrow{\$} R$ i.e. Both s_1, b_1 are uniformly random; just like H_1!!

This means, since B_1 just outputs what A outputs,

$$Pr(B_1 \text{ outputs } 1 \text{ in Exp } 0) = Pr(A \text{ outputs } 1 \text{ in } H_0) = p_0$$

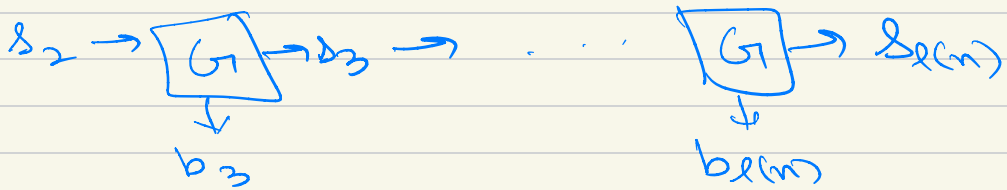
Similarly, $Pr(B_1 \text{ outputs } 1 \text{ in Exp } 1) = p_1$

Hence, $PRGAdv(\mathcal{B}_1, G) = |p_1 - p_0|$. QED.

We'll now define more hybrids, replacing $b_2, \ldots, b_{\ell(n)}$ by random values, one-by-one:

Hybrid H_2:

Oral samples $s_2 \leftarrow\!\$ \{0,1\}^n$, and $b_1, b_2 \leftarrow\!\$ \{0,1\}$ and computes:



H_j: first j bits are random...

$H_{\ell(n)}$:

$(b_1, \ldots, b_{\ell(n)}) \leftarrow\!\$ \{0,1\}$

$\equiv$ Exp 1 in PRG security defn for G'.

$\forall i \in \{0, \ldots, \ell(n)\}$, $p_i = \text{Prob}\left(\begin{matrix}\text{A outputs}\\ 1 \text{ in } H_i\end{matrix}\right)$

Similar to Claim 1 above, we could make a different adversary B_i that breaks G_1 if

A can distinguish b/w hybrids H_{i-1} and H_i ... for all i...

If $\ell(n)$ was constant $\Rightarrow$ constant # Hybrids,

this is enough.

But, if $\ell(n) = \text{poly}(n)$, this is NOT enough!

(There are contrived schemes which are insecure, but one could define a series of hybrids and prove that each one is indistinguishable from the next !!

See 2021/088 on eprint!

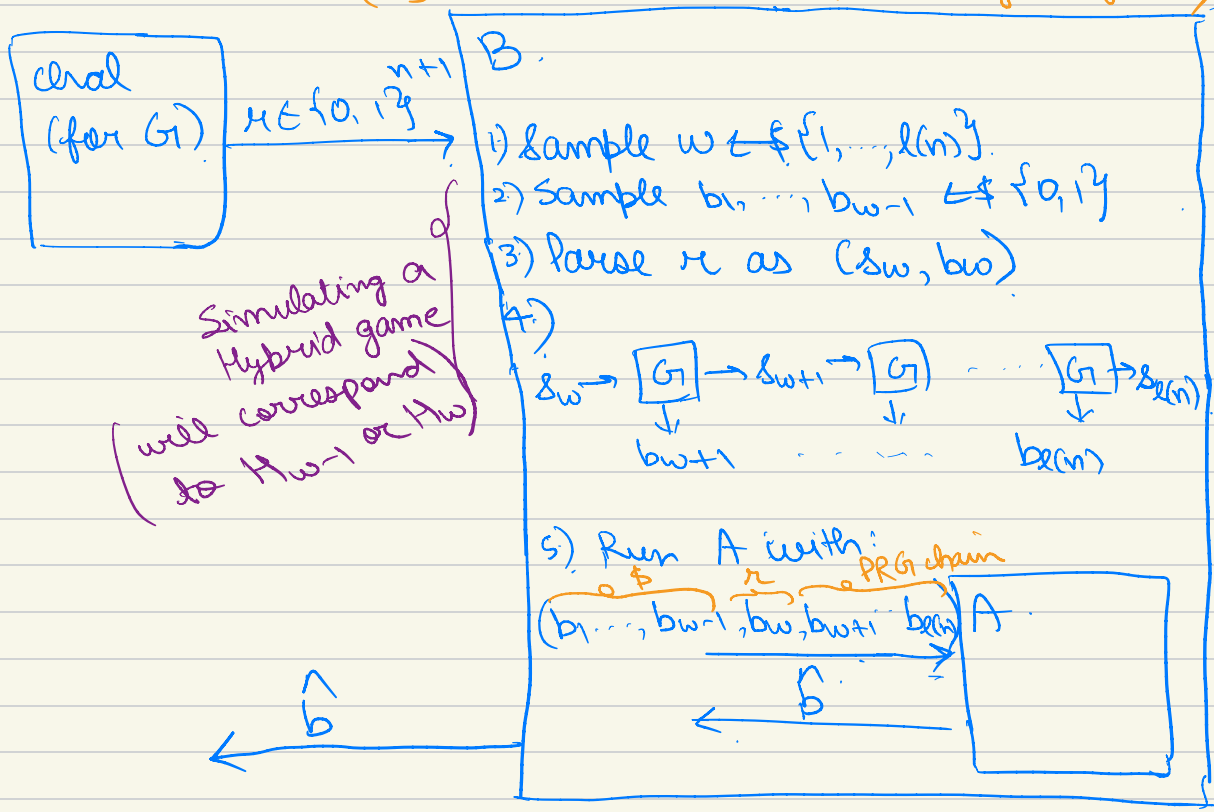
Hence, we'll follow a slightly different approach to prove security...

By defn.,

$$\mathrm{PRG\,Adv}[A, G'] = \left| \Pr_{\text{Exp0}}[W_0] - \Pr_{\text{Exp1}}[W_1] \right|$$

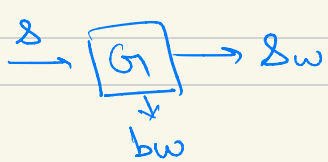
$$= \left| p_0 - p_{\ell(n)} \right|.$$

Now, we'll construct the adv B playing the PRG security game for G.
(B will act as chal to A in 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, $Pr(W_{0B}) = Pr(A \text{ outputs } 1 \text{ in } H_{w-1})$

$\rightarrow Pr(A \text{ outputs } 1 \text{ in } H_{j-1})$

$Pr(W_{0,B} \mid w = j) = p_{j-1}$ and

event that B outputs 1 in Exp_0

* 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}.

$Pr(W_{1B} \mid w = j) = p_j$ for all j...

So,

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

By total probability :-

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

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

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

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

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

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

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

Hence, G' is a secure PRG.

PRFs : (pseudo random functions)

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

- K : key space
- X : input space
- Y : output space.

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

$\text{Func}[X, Y]$ = space of all functions from X to Y.

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

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

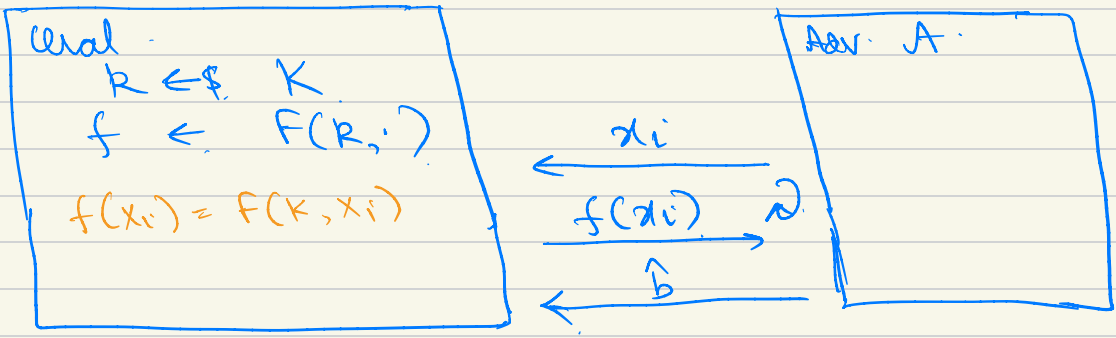
But # functions in $\text{Func}[X, Y] = |Y|^{|X|}$

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

i.e. $\text{Func}[X, Y] \not\approx |K|$.

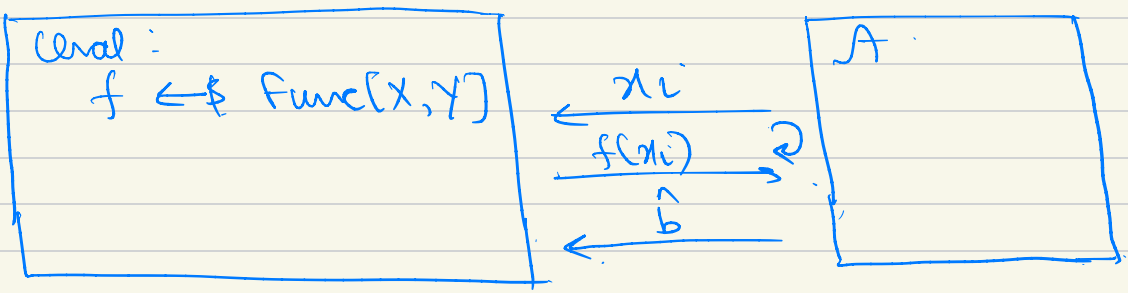
PRF security game :

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 :

$$PRFAdv[A, F] = |Pr[W_0] - Pr[W_1]|$$

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

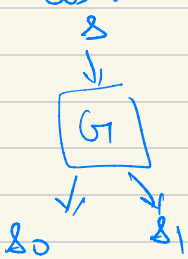
A PRF F is secure if $\forall$ efficient (PPT) adversaries A,

$$PRFAdv[A, F] \leq negl(\lambda).$$

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

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

visualize G as: "length-doubling"



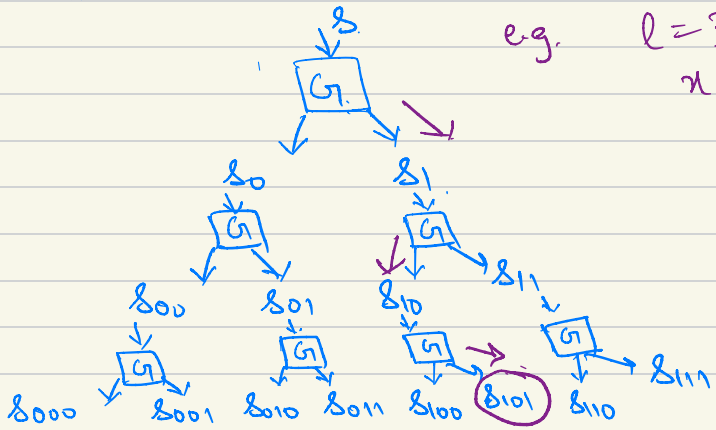
lets say, $s_0 = G_0(s)$

$s_1 = G_1(s)$.

we'll make PRF $F: \underbrace{S}_{\text{Key space} = S} \times \underbrace{\{0,1\}^l}_{\text{Bit strings, lets say } l = \lambda} \to S$ as follows:-

$F(s, x)$: let $x = (x_1, \ldots, x_l)$:

eg. $l = 3$ and $x = \{1, 0, 1\}$



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

$= G_1(G_0(G_1(s)))$

$(x_3 \; x_2 \; x_1)$

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

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

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

Is F a secure PRF?

1.) Efficiency: ℓ evals of G_1. ✓
$= \text{poly}(\lambda)$

2.) security:

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

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

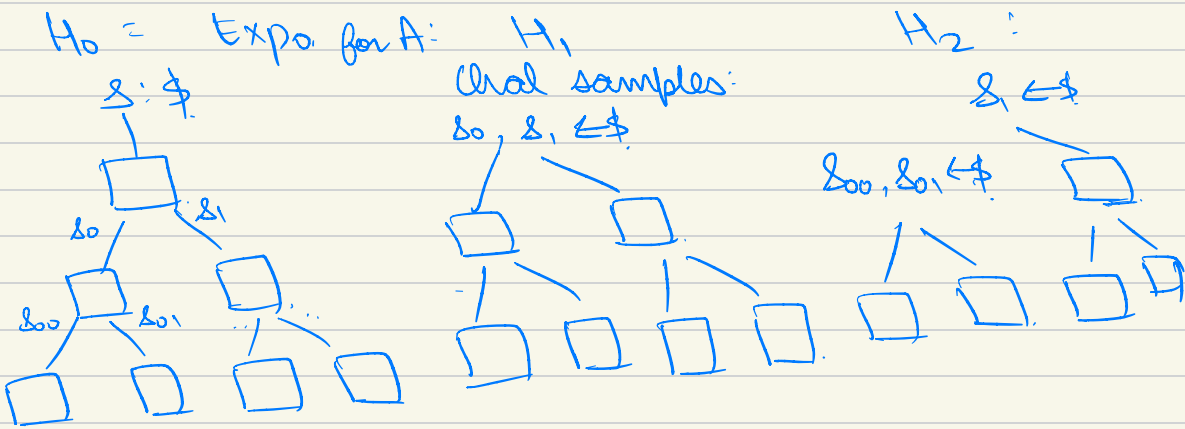
i.e. G is a secure PRG. $\Rightarrow$ F is a secure PRF.

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:

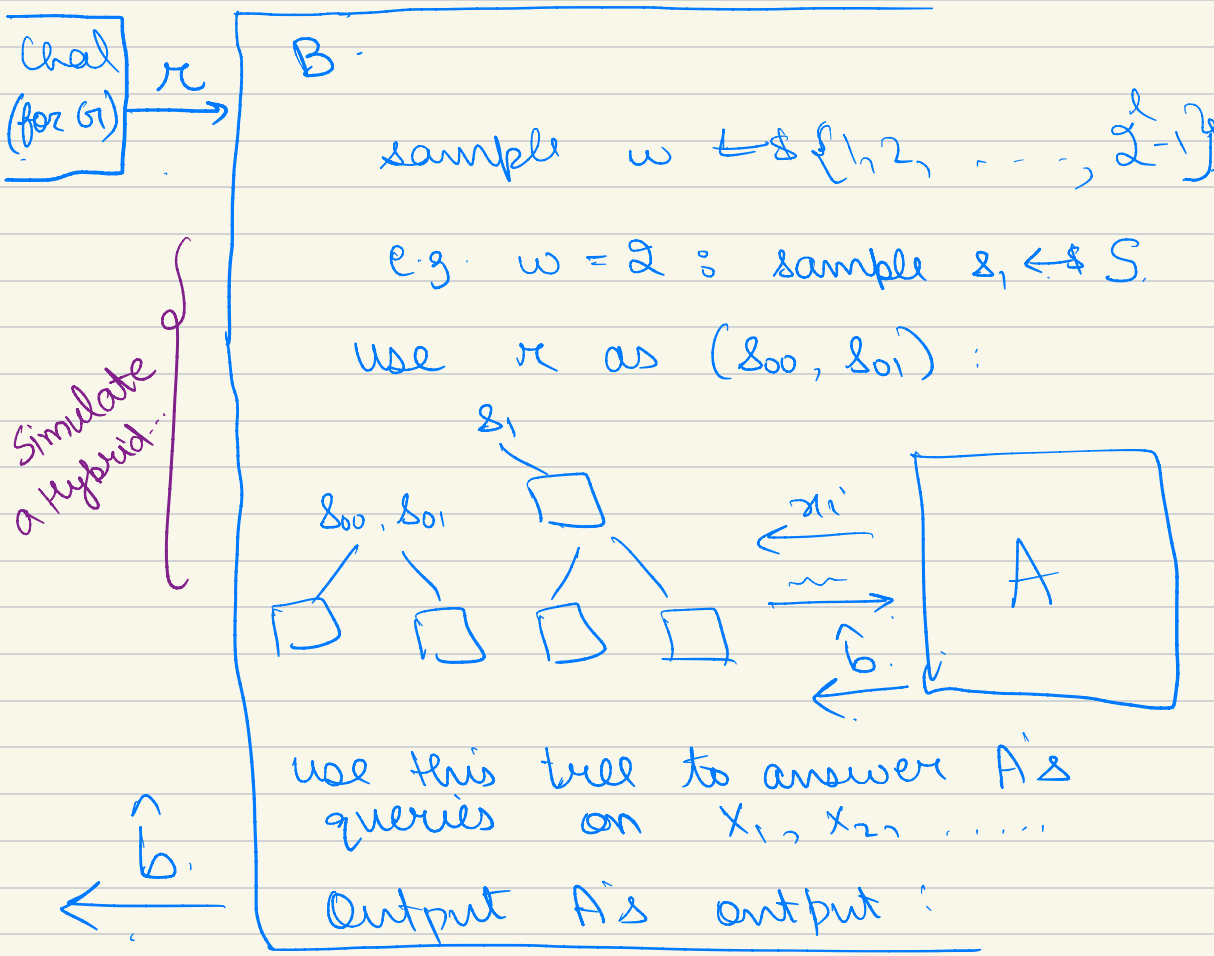


note, there are 2^{i-1} PRGs on level i of tree.

$\Rightarrow$ # Hybrids: $1 + 2 + 2^2 + \ldots + 2^{l-1}$

$\Rightarrow 2^l - 1$. ∴ This is a problem...

Now, we'll construct B, an adversary for the PRG game against G_1:



Analysis of B's advantage :-

By similar argument as that for $PRG_{1\text{-poly}}$ construction,

1 in Exp0 — 1 in Exp1

$$PRGAdv[B, G] = \left| Pr(W_{0B}) - Pr(W_{1B}) \right|$$

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

$Pr(w = j) = \frac{1}{2^\ell - 1}$

(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,

$$PRGAdv[B, G] = \frac{1}{2^\ell - 1} \left| p_0 + p_1 \cdots + p_{2^\ell - 2} - p_1 \cdots - p_{2^\ell - 1} \right|$$

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

$2^\ell - 1$

Issue: Recall, $l = \lambda$.

Even if $\mathrm{PRFAdv}[A, F]$ 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...

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

i.e. 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 $2 \cdot Q$ hybrids !!

Full proof in book (sec. 4.6)

Symmetric Crypto - Summary :