5.1 A Searching Adversary

5.1.1 What is the advantage? Is it negligible?

Because $G : \{0,1\}^\lambda \to \{0,1\}^{\lambda+l}$ is injective, every key s has exactly one expanded key. Therefore $2^{\lambda+l} - 2^\lambda$ expanded keys cannot be the result of this PRG.

For $L^G_{PRG-real}$ we get:

$$Pr(A \diamond L^G_{PRG-real} \Rightarrow 1) \;=\; 1$$

Because the distinguisher uses the same injective function as the PRG and checks all possible inputs s', it is obvious that it will always return 1.

For $L^G_{PRG-rand}$ we get:

$$\begin{aligned} Pr(A \diamond L^G_{PRG-rand} \Rightarrow 1) &= 1 - Pr(A \diamond L^G_{PRG-rand} \Rightarrow 0) \\ &= 1 - \frac{2^{\lambda+l} - 2^\lambda}{2^{\lambda+l}} \\ &= 1 - 1 + \frac{2^\lambda}{2^{\lambda+l}} \\ &= \frac{2^\lambda}{2^{\lambda+l}} \\ &= \frac{1}{2^l} \end{aligned}$$

Because the library $L^G_{PRG-real}$ can return all possible keys in $\{0,1\}^{\lambda+l}$, especially the $2^{\lambda+l} - 2^\lambda$ keys which cannot be returned by the PRG because of its injectivity.

For the advantage we will get:

$$\begin{aligned} Bias(A) &= \mid Pr(A \diamond L^G_{PRG-real} \Rightarrow 1) - Pr(A \diamond L^G_{PRG-rand} \Rightarrow 1) \mid \\ &= \mid 1 - \frac{1}{2^l} \mid \\ &= \frac{2^l - 1}{2^l} \end{aligned}$$

We can see that the limit ($\lambda \to \infty$) of this function will not get to 0 and is therefore not negligible.

5.1.2 Does this contradict to the concept of PRG?

No, because the distinguisher is not indistinguishable with a polynomial distinguisher and therefore this does not contradict the concept ot PRG.

5.1.2 What if G is not injective?

If G is not injective, the possible key space of the extended keys could be even smaller. Therefore we have the same problem as above. Therefore its advantage will be approximately the same for the said distinguisher.

16-124-836
Marcel ZAUDER

5.2 Lucky Gambler

For $L^G_{PRG-real}$ we get:

$$Pr(A \diamond L^G_{PRG-real} \Rightarrow 1) \;=\; \frac{1}{2^\lambda}$$

For $L^G_{PRG-real}$ we get:
x possible: $\frac{2^\lambda}{2^{\lambda+l}} = \frac{1}{2^l}$

$$\begin{aligned} Pr(A \diamond L^G_{PRG-rand} \Rightarrow 1) \;&=\; Pr(A \diamond L^G_{PRG-rand_{x\ possible}} \Rightarrow 1) + Pr(A \diamond L^G_{PRG-rand_{x\ impossible}} \Rightarrow 1) \\ &=\; \frac{1}{2^l} \cdot \frac{1}{2^\lambda} \cdot 1 + 0 \\ &=\; \frac{1}{2^{\lambda+l}} \end{aligned}$$

For the advantage we will get:

$$\begin{aligned} Bias(A) \;&= \mid Pr(A \diamond L^G_{PRG-real} \Rightarrow 1) - Pr(A \diamond L^G_{PRG-rand} \Rightarrow 1) \mid \\ &= \mid \frac{1}{2^\lambda} - \frac{1}{2^{\lambda+l}} \mid \\ &=\; \frac{2^l - 1}{2^{\lambda+l}} \\ \Rightarrow \lim_{\lambda\to\infty} (p(x) \cdot \frac{2^l - 1}{2^{\lambda+l}}) \;&=\; 0 \quad \forall p(x) \end{aligned}$$

Therefore it is negligible.

5.3 PRGs

5.3.A

$L^A_{PRG-real}$
$\underline{\text{QUERY}_A():}$
$x\|y\|z = G(s)$
return $G(x)\|G(z)$

This is our starting library $L^A_{PRG-real}$.

$\underline{\text{QUERY}_A():}$
$x\|y\|z = \text{QUERY}_G()$
return $G(x)\|G(z)$

$\diamond$

$L^G_{PRG-real}$
$\underline{\text{QUERY}_G():}$
$s \leftarrow \{0,1\}^\lambda$
return $G(s)$

We are using the library $L^G_{PRG-real}$ to make a hybrid library.

$\underline{\text{QUERY}_A():}$
$x\|y\|z = \text{QUERY}_G()$
return $G(x)\|G(z)$

$\diamond$

$L^G_{PRG-rand}$
$\underline{\text{QUERY}_G():}$
$r \leftarrow \{0,1\}^{3\lambda}$
return r

Because we know G is secure we can replace the $L^G_{PRG-real}$ with $L^G_{PRG-rand}$.

$\underline{\text{QUERY}_A():}$
$x\|y\|z \leftarrow \{0,1\}^{3\lambda}$
$x' = G(x)$
$z' = G(z)$
return $x'\|z'$

This subroutine of $L^G_{PRG-rand}$ was inlined. New variables were introduced to proceed further.

$\underline{\text{QUERY}_A():}$
$x' = \text{QUERY}_G()$
$z' = \text{QUERY}_G()$
return $x'\|z'$

$\diamond$

$L^G_{PRG-real}$
$\underline{\text{QUERY}_G():}$
$s \leftarrow \{0,1\}^\lambda$
return $G(s)$

We are using the library $L^G_{PRG-real}$ to make a hybrid library. Therefore we do not need x and z and could eliminate the first step of $\text{QUERY}_A()$.

$\underline{\text{QUERY}_A():}$
$x' = \text{QUERY}_G()$
$z' = \text{QUERY}_G()$
return $x'\|z'$

$\diamond$

$L^G_{PRG-rand}$
$\underline{\text{QUERY}_G():}$
$r \leftarrow \{0,1\}^{3\lambda}$
return r

Because we know G is secure we can replace the $L^G_{PRG-real}$ with $L^G_{PRG-rand}$.

$\underline{\text{QUERY}_A():}$
$x' \leftarrow \{0,1\}^{3\lambda}$
$z' \leftarrow \{0,1\}^{3\lambda}$
return $x'\|z'$

The subroutine of $L^G_{PRG-rand}$ was inlined.

$L^A_{PRG-rand}$
$\underline{\text{QUERY}_A():}$
$r \leftarrow \{0,1\}^{6\lambda}$
return r

Concatenating 3λ uniform bits with 3λ uniform bits has the same effect as sampling 6λ uniform bits. This results in the forming of a chain of indistinguishable libraries resulting in $L^A_{PRG-rand}$, which implies that A is a secure PRG.

5.3.B

$L^B_{PRG-real}$
QUERY$_B()$:
$x\|\|y\|\|z = G(s)$
return $x\|\|y$

This is our starting library $L^B_{PRG-real}$.

QUERY$_B()$:
$x\|\|y\|\|z =$ QUERY$_G()$
return $x\|\|y$

$\diamond$

$L^G_{PRG-real}$
QUERY$_G()$:
$s \leftarrow \{0,1\}^\lambda$
return $G(s)$

We are using the library $L^G_{PRG-real}$ to make a hybrid library.

QUERY$_B()$:
$x\|\|y\|\|z =$ QUERY$_G()$
return $x\|\|y$

$\diamond$

$L^G_{PRG-rand}$
QUERY$_G()$:
$r \leftarrow \{0,1\}^{3\lambda}$
return r

Because we know G is secure we can replace the $L^G_{PRG-real}$ with $L^G_{PRG-rand}$.

QUERY$_B()$:
$x\|\|y\|\|z \leftarrow \{0,1\}^{3\lambda}$
return $x\|\|y$

The subroutine of $L^G_{PRG-rand}$ was inlined.

QUERY$_B()$:
$x \leftarrow \{0,1\}^\lambda$
$y \leftarrow \{0,1\}^\lambda$
$z \leftarrow \{0,1\}^\lambda$
return $x\|\|y$

Choosing 3λ uniformly random bits and then splitting them up in 3 equal parts has exactly the same effect as choosing λ uniformly random bits three times. Additionally we can see that we do not need z anymore and can remove it from the PRG.

$L^B_{PRG-rand}$
QUERY$_B()$:
$r \leftarrow \{0,1\}^{2\lambda}$
return r

In the end we can combine the concetination of x and y so we only have one variable r which is chosen uniformly random out of 2λ. This results in the forming of a chain of indistinguishable libraries resulting in $L^B_{PRG-rand}$, which implies that B is a secure PRG.

5.3.C

We will consider the following distinguisher:

A
$x\|\|y =$ QUERY$()$
return $x \stackrel{?}{=} y$

It is obvious that $Pr(A \diamond L^C_{PRG-real} \Rightarrow 1) = 1$, because C is concatenating two bit strings which are generated from the same s. On the other hand for $Pr(A \diamond L^C_{PRG-rand} \Rightarrow 1)$ outputs only 1 if $L^C_{PRG-rand}$ outputs a string with the same first and second half.
The probability for this is $\frac{1}{2^{3\lambda}}$. The advantage is therefore:

$$Bias(A) \ = \mid Pr(A \diamond L^C_{PRG-real} \Rightarrow 1) - Pr(A \diamond L^C_{PRG-rand} \Rightarrow 1) \mid = \ 1 - \frac{1}{2^{3\lambda}}$$

, which is clearly not negligible for $\lambda \to \infty$.

5.3.D

We will consider the following distinguisher:

A
$x \| y = \text{QUERY}()$
return $G(o^\lambda) \stackrel{?}{=} y$

It is obvious that $Pr(A \diamond L^D_{PRG-real} \Rightarrow 1) = 1$, because D is producing a bit-string which second half is equal to $G(o^\lambda)$. On the other hand for $Pr(A \diamond L^D_{PRG-rand} \Rightarrow 1)$ outputs only 1 if $L^D_{PRG-rand}$ outputs a string with the second half equal to $G(o^\lambda)$.
The probability for this is $\frac{1}{2^{3\lambda}}$. The advantage is therefore:

$$Bias(A) \ = \mid Pr(A \diamond L^D_{PRG-real} \Rightarrow 1) - Pr(A \diamond L^D_{PRG-rand} \Rightarrow 1) \mid = \ 1 - \frac{1}{2^{3\lambda}}$$

, which is clearly not negligible for $\lambda \to \infty$.

5.3.E

$L^E_{PRG-real}$
$\text{QUERY}_F()$:
$x = G(s)$
$y = G(o^\lambda)$
return $x \oplus y$

This is our starting library $L^E_{PRG-real}$.

$\text{QUERY}_F()$:
$x = G(s)$
return x

Because we are using an "XOR" for the return value, which does not affect the uniform distribution of our PRG, we can exclude it. Therefore we can see directly that this library is secure. Therefore E is a secure PRG.

5.3.F

$L^F_{PRG-real}$
$\text{QUERY}_F()$:
$x = G(s_L)$
$y = G(s_R)$
return $x \oplus y$

This is our starting library $L^F_{PRG-real}$.

$\text{QUERY}_F()$:
$x = \text{QUERY}_G()$:
$y = \text{QUERY}_G()$:
return $x \oplus y$

$\diamond$

$L^G_{PRG-real}$
$\text{QUERY}_G()$:
$s \leftarrow \{0,1\}^\lambda$
return $G(s)$

We are using the library $L^G_{PRG-real}$ to make a hybrid library.

$\text{QUERY}_F()$:
$x = \text{QUERY}_G()$:
$y = \text{QUERY}_G()$:
return $x \oplus y$

$\diamond$

$L^G_{PRG-rand}$
$\text{QUERY}_G()$:
$r \leftarrow \{0,1\}^{3\lambda}$
return r

Because we know G is secure we can replace the $L^G_{PRG-real}$ with $L^G_{PRG-rand}$.

$\underline{\text{QUERY}_F():}$
$r \leftarrow \{0,1\}^{3\lambda}$
$r' \leftarrow \{0,1\}^{3\lambda}$
return $r \oplus r'$

The subroutine of $L^G_{PRG-rand}$ was inlined.

$L^F_{PRG-rand}$
$\underline{\text{QUERY}_F():}$
$r \leftarrow \{0,1\}^{3\lambda}$
return r

Because using two random generated bit strings in an XOR is the same as generating one random bit string we can change this in the PRG. Therefore we have shown that $L^F_{PRG-real}$ and $L^F_{PRG-rand}$ are equivalent and therefore $L^F_{PRG-real}$ is secure.

5.3.H

$L^H_{PRG-real}$
$\underline{\text{QUERY}_H():}$
$x = G(s_L)$
$y = G(s_R)$
return $x \| y$

This is our starting library $L^H_{PRG-real}$.

$\underline{\text{QUERY}_H():}$
$x = \text{QUERY}_G():$
$y = \text{QUERY}_G():$
return $x \| y$

$\diamond$

$L^G_{PRG-real}$
$\underline{\text{QUERY}_G():}$
$s \leftarrow \{0,1\}^{\lambda}$
return $G(s)$

We are using the library $L^G_{PRG-real}$ to make a hybrid library.

$\underline{\text{QUERY}_H():}$
$x = \text{QUERY}_G():$
$y = \text{QUERY}_G():$
return $x \| y$

$\diamond$

$L^G_{PRG-rand}$
$\underline{\text{QUERY}_G():}$
$r \leftarrow \{0,1\}^{3\lambda}$
return r

Because we know G is secure we can replace the $L^G_{PRG-real}$ with $L^G_{PRG-rand}$.

$\underline{\text{QUERY}_H():}$
$r \leftarrow \{0,1\}^{3\lambda}$
$r' \leftarrow \{0,1\}^{3\lambda}$
return $r \| r'$

The subroutine of $L^G_{PRG-rand}$ was inlined.

$L^H_{PRG-rand}$
$\underline{\text{QUERY}_F():}$
$r \leftarrow \{0,1\}^{6\lambda}$
return r

Because using two random generated 3λ-bit strings and concatenating them is the same as generating one random 6λ-bit string we can change this in the PRG. Therefore we have shown that $L^F_{PRG-real}$ and $L^F_{PRG-rand}$ are equivalent and therefore $L^F_{PRG-real}$ is secure.

5.4 Breaking a PRG Candidate

5.4.1 Proof that G is not a secure PRG for any function f

We consider the following distinguisher:

A
$s \| x = \text{QUERY}()$
return $f(s) \stackrel{?}{=} x$

It is obvious that $Pr(A \diamond L^G_{PRG-real} \Rightarrow 1) = 1$, because x is computed by $G(s)$ from s. Therefore x and $G(s)$ will be always the same in this situation.
For $Pr(A \diamond L^G_{PRG-rand} \Rightarrow 1)$ the x part has to be the same as the output of $G(s)$. This will only be the case in 1 out of $2^{\lambda+l}$ possibilities ($f(s)$ outputs a $2^{\lambda+l}$-bit string).
The advantage is therefore:

$$Bias(A) \ = \mid Pr(A \diamond L^G_{PRG-real} \Rightarrow 1) - Pr(A \diamond L^G_{PRG-rand} \Rightarrow 1) \mid = \ 1 - \frac{1}{2^{\lambda+l}}$$

, which is clearly not negligible for $\lambda \to \infty$.

5.4.2 Proof that G is not a secure PRG for an algorithm V

We consider the following distinguisher:

V
$x = \text{QUERY}()$
return $G(G^{-1}(x)) \stackrel{?}{=} x$

Further we define:

$$p := P[V(G(s)) = s] \ > \ negl(\lambda)$$

For a random $G(s)$ library we get:

$G(s)_{random}$
$r \leftarrow \{0,1\}^{\lambda+l}$
return r

In order for V to return a valid output, r has to be in $G(x)$. If it is not the case there will be no valid return value. The probability for r being in $G(x)$ is: $Pr(r \in G(x)) = \frac{2^\lambda}{2^{\lambda+l}} = \frac{1}{2^l}$.
The advantage of V is therefore:

$$\begin{aligned} Bias(V) \ &= \mid Pr(V \diamond L^G_{PRG-real} \Rightarrow 1) - Pr(V \diamond L^G_{PRG-rand} \Rightarrow 1) \mid \\ &= \ p - \frac{1}{2^l} \cdot p \\ &= \ \frac{2^l - 1}{2^l} \cdot p \\ \lim_{\lambda \to \infty} (Bias(V)) \ &\neq \ 0 \qquad\qquad for\ any\ l > 0 \end{aligned}$$

Therefore the advantage is not negligible and G is not a secure PRG.