PRGs

Intuition: A Pseudorandom Generator (PRG) takes a *small, uniformly random seed*, and stretches it into a longer string that is *not* uniformly random, but is indistinguishable from random.

PRGs

Definition (PRG)

Let ℓ be a polynomial, and let G be a deterministic polynomial-time algorithm such that for any n and any input $s \in \{0,1\}^n$, $G(s)$ is a string of length $\ell(n)$. We say that G is a pseudorandom generator if the following conditions hold:

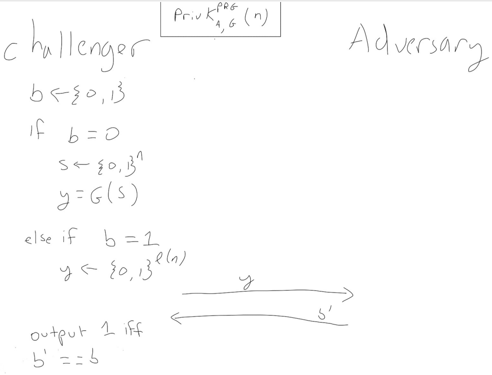
1. Expansion: for every n it holds that $\ell(n) > n$.
2. Pseudorandomness: for any PPT algorithm $\mathcal{A}$, there is a negligible function negl(n) such that $\Pr[\mathsf{PrivK}^{\mathsf{prg}}_{\mathcal{A},\mathsf{G}}(n) = 1] \leq \frac{1}{2} + \mathsf{negl(n)}$

PRGs

Definition (PRG)

Let ℓ be a polynomial, and let G be a deterministic polynomial-time algorithm such that for any n and any input $s \in \{0,1\}^n$, $G(s)$ is a string of length $\ell(n)$. We say that G is a pseudorandom generator if the following conditions hold:

1. Expansion: for every n it holds that $\ell(n) > n$.
2. Pseudorandomness: for any PPT algorithm $\mathcal{A}$, there is a negligible function negl(n) such that $\Pr[\mathrm{PrivK}^{\mathrm{prg}}_{\mathcal{A},\mathrm{G}}(n) = 1] \leq \frac{1}{2} + \mathrm{negl(n)}$



PRGs

Definition (PRG)

Let ℓ be a polynomial, and let G be a deterministic polynomial-time algorithm such that for any n and any input $s \in \{0,1\}^n$, $G(s)$ is a string of length $\ell(n)$. We say that G is a pseudorandom generator if the following conditions hold:

1. Expansion: for every n it holds that $\ell(n) > n$.
2. Pseudorandomness: for any PPT algorithm $\mathcal{A}$, there is a negligible function negl(n) such that $\Pr[\mathrm{PrivK}^{\mathrm{prg}}_{\mathcal{A},\mathrm{G}}(n) = 1] \leq \frac{1}{2} + \mathrm{negl(n)}$

G is *NOT* pseudorandom if: $\exists$ a PPT algorithm $\mathcal{A}$ and some polynomial $p(\cdot)$, s.t. $\Pr[\mathrm{PrivK}^{\mathrm{prg}}_{\mathcal{A},\mathrm{G}}(n) = 1] > \frac{1}{2} + \frac{1}{p(n)}$

An Insecure PRG

$G(s_1 \cdots s_n)$:
Let $s_{n+1} = \bigoplus_{i \in \{1,\ldots,n\}} s_i$
Output $s_1 \cdots s_{n+1}$

An Insecure PRG

$G(s_1 \cdots s_n)$:
Let $s_{n+1} = \bigoplus_{i \in \{1,\dots,n\}} s_i$
Output $s_1 \cdots s_{n+1}$

$\mathcal{A}$ receives $y = s_1 \cdots s_{n+1}$ from the challenger, and has to guess whether $b = 0$ (i.e. $y = G(s)$) or $b = 1$ (i.e. $y \leftarrow \{0,1\}^{n+1}$).

An Insecure PRG

$G(s_1 \cdots s_n)$:
Let $s_{n+1} = \bigoplus_{i \in \{1,\ldots,n\}} s_i$
Output $s_1 \cdots s_{n+1}$

$\mathcal{A}$ receives $y = s_1 \cdots s_{n+1}$ from the challenger, and has to guess whether $b = 0$ (i.e. $y = G(s)$) or $b = 1$ (i.e. $y \leftarrow \{0,1\}^{n+1}$).

$\underline{\mathcal{A}}$:
Compute $\hat{s}_{n+1} = \bigoplus_{i \in \{1,\ldots,n\}} s_i$.
If $\hat{s}_{n+1} = s_{n+1}$, output 0
Else, output 1.

An Insecure PRG

$G(s_1 \cdots s_n)$:
Let $s_{n+1} = \bigoplus_{i \in \{1,\ldots,n\}} s_i$
Output $s_1 \cdots s_{n+1}$

$\mathcal{A}$ receives $y = s_1 \cdots s_{n+1}$ from the challenger, and has to guess whether $b = 0$ (i.e. $y = G(s)$) or $b = 1$ (i.e. $y \leftarrow \{0,1\}^{n+1}$).

$\underline{\mathcal{A}:}$
Compute $\hat{s}_{n+1} = \bigoplus_{i \in \{1,\ldots,n\}} s_i$.
If $\hat{s}_{n+1} = s_{n+1}$, output 0
Else, output 1.

$$\begin{aligned}\Pr[\mathsf{PrivK}^{\mathsf{prg}}_{\mathcal{A},\mathsf{G}}(n) = 1] &= \Pr[b' == b] \\ &= \Pr[b' = 0 \wedge b = 0] + \Pr[b' = 1 \wedge b = 1]\end{aligned}$$

An Insecure PRG

$G(s_1 \cdots s_n)$:
Let $s_{n+1} = \bigoplus_{i \in \{1,\ldots,n\}} s_i$
Output $s_1 \cdots s_{n+1}$

$\mathcal{A}$ receives $y = s_1 \cdots s_{n+1}$ from the challenger, and has to guess whether $b = 0$ (i.e. $y = G(s)$) or $b = 1$ (i.e. $y \leftarrow \{0,1\}^{n+1}$).

$\underline{\mathcal{A}}$:
Compute $\hat{s}_{n+1} = \bigoplus_{i \in \{1,\ldots,n\}} s_i$.
If $\hat{s}_{n+1} = s_{n+1}$, output 0
Else, output 1.

$$\begin{aligned}
\Pr[\mathrm{PrivK}^{\mathrm{prg}}_{\mathcal{A},\mathrm{G}}(n) = 1] &= \Pr[b' == b] \\
&= \Pr[b' = 0 \wedge b = 0] + \Pr[b' = 1 \wedge b = 1] \\
&= \Pr[b' = 0 \mid b = 0] \cdot \frac{1}{2} + \Pr[b' = 1 \mid b = 1] \cdot \frac{1}{2}
\end{aligned}$$

An Insecure PRG

$G(s_1 \cdots s_n)$:
Let $s_{n+1} = \bigoplus_{i \in \{1,\ldots,n\}} s_i$
Output $s_1 \cdots s_{n+1}$

$\mathcal{A}$ receives $y = s_1 \cdots s_{n+1}$ from the challenger, and has to guess whether $b = 0$ (i.e. $y = G(s)$) or $b = 1$ (i.e. $y \leftarrow \{0,1\}^{n+1}$).

$\underline{\mathcal{A}}$:
Compute $\hat{s}_{n+1} = \bigoplus_{i \in \{1,\ldots,n\}} s_i$.
If $\hat{s}_{n+1} = s_{n+1}$, output 0
Else, output 1.

$$\begin{aligned}
\Pr[\mathsf{PrivK}^{\mathsf{prg}}_{\mathcal{A},\mathsf{G}}(n) = 1] &= \Pr[b' == b] \\
&= \Pr[b' = 0 \wedge b = 0] + \Pr[b' = 1 \wedge b = 1] \\
&= \Pr[b' = 0 \mid b = 0] \cdot \frac{1}{2} + \Pr[b' = 1 \mid b = 1] \cdot \frac{1}{2} \\
&= 1 \cdot \frac{1}{2} + \frac{1}{2} \cdot \frac{1}{2}
\end{aligned}$$

An Insecure PRG

$G(s_1 \cdots s_n)$:
Let $s_{n+1} = \bigoplus_{i \in \{1,\ldots,n\}} s_i$
Output $s_1 \cdots s_{n+1}$

$\mathcal{A}$ receives $y = s_1 \cdots s_{n+1}$ from the challenger, and has to guess whether $b = 0$ (i.e. $y = G(s)$) or $b = 1$ (i.e. $y \leftarrow \{0,1\}^{n+1}$).

$\underline{\mathcal{A}}$:
Compute $\hat{s}_{n+1} = \bigoplus_{i \in \{1,\ldots,n\}} s_i$.
If $\hat{s}_{n+1} = s_{n+1}$, output 0
Else, output 1.

$$\begin{aligned}
\Pr[\mathrm{PrivK}^{\mathrm{prg}}_{\mathcal{A},\mathrm{G}}(n) = 1] &= \Pr[b' == b] \\
&= \Pr[b' = 0 \wedge b = 0] + \Pr[b' = 1 \wedge b = 1] \\
&= \Pr[b' = 0 \mid b = 0] \cdot \frac{1}{2} + \Pr[b' = 1 \mid b = 1] \cdot \frac{1}{2} \\
&= 1 \cdot \frac{1}{2} + \frac{1}{2} \cdot \frac{1}{2} = \frac{3}{4} = \frac{1}{2} + \frac{1}{4}
\end{aligned}$$

Pseudorandom vs. Random

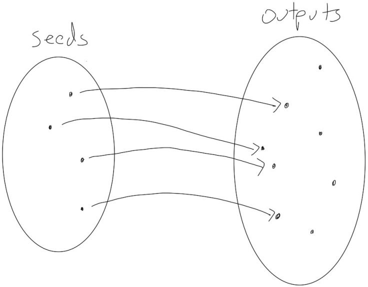
Even when G is pseudorandom, it is very far from random.

Pseudorandom vs. Random

Even when G is pseudorandom, it is very far from random.
Consider G that doubles its input length: $G(s_1 \cdots s_n) = r_1, \ldots, r_{2n}$.

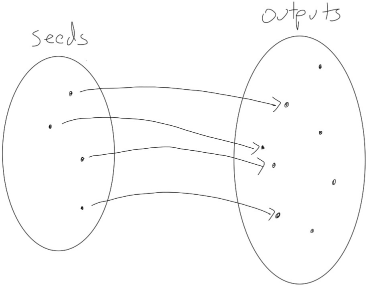
Pseudorandom vs. Random

Even when G is pseudorandom, it is very far from random.
Consider G that doubles its input length: $G(s_1 \cdots s_n) = r_1, \ldots, r_{2n}$.
Recall, G is deterministic, so it can only map each input seed to a single output value.



Pseudorandom vs. Random

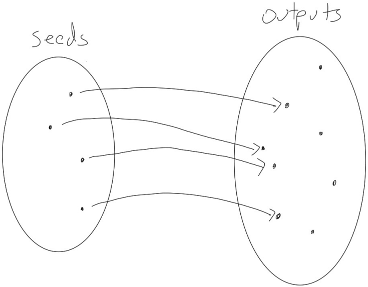
Even when G is pseudorandom, it is very far from random.
Consider G that doubles its input length: $G(s_1 \cdots s_n) = r_1, \ldots, r_{2n}$.
Recall, G is deterministic, so it can only map each input seed to a single output value.



For input of length n:
How many different input seeds are there?
How many different outputs does G have (maximum)?
How many strings of length $2n$ are there?
If you choose $y \leftarrow \{0,1\}^{2n}$ (i.e. uniformly at random),
what is the probability that you choose an output of G?

Pseudorandom vs. Random

Even when G is pseudorandom, it is very far from random.
Consider G that doubles its input length: $G(s_1 \cdots s_n) = r_1, \ldots, r_{2n}$.
Recall, G is deterministic, so it can only map each input seed to a single output value.



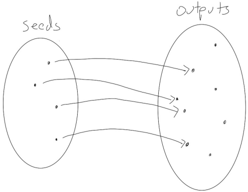
For input of length n:

How many different input seeds are there?	2^n
How many different outputs does G have (maximum)?	2^n
How many strings of length $2n$ are there?	2^{2n}

If you choose $y \leftarrow \{0,1\}^{2n}$ (i.e. uniformly at random),
what is the probability that you choose an output of G? $\frac{2^n}{2^{2n}} = 2^{n-2n} = 2^{-n}$

Pseudorandom vs. Random

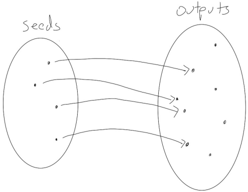
Even when G is pseudorandom, it is very far from random.
Consider G that doubles its input length: $G(s_1 \cdots s_n) = r_1, \ldots, r_{2n}$.
Recall, G is deterministic, so it can only map each input seed to a single output value.



If we allow an exponential time adversary, *every* PRG is insecure!

Pseudorandom vs. Random

Even when G is pseudorandom, it is very far from random.
Consider G that doubles its input length: $G(s_1 \cdots s_n) = r_1, \ldots, r_{2n}$.
Recall, G is deterministic, so it can only map each input seed to a single output value.

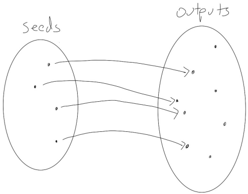


If we allow an exponential time adversary, *every* PRG is insecure!

$\underline{\mathcal{A}:}$
Let $S = \emptyset$
For each $s \in \{0,1\}^n$
    $S = S \cup G(s)$.
If $y \in S$ output 0
else output 1.

Pseudorandom vs. Random

Even when G is pseudorandom, it is very far from random.
Consider G that doubles its input length: $G(s_1 \cdots s_n) = r_1, \ldots, r_{2n}$.
Recall, G is deterministic, so it can only map each input seed to a single output value.



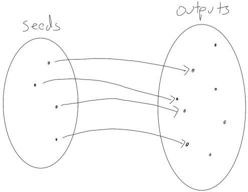
If we allow an exponential time adversary, *every* PRG is insecure!

$\underline{\mathcal{A}}$:
Let $S = \emptyset$
For each $s \in \{0,1\}^n$
$\quad S = S \cup G(s)$.
If $y \in S$ output 0
else output 1.

$$\Pr[\mathrm{PrivK}^{\mathrm{prg}}_{\mathcal{A},\mathrm{G}}(n) = 1] = \Pr[b' == b]$$
$$= \Pr[b' = 0 \wedge b = 0] + \Pr[b' = 1 \wedge b = 1]$$

Pseudorandom vs. Random

Even when G is pseudorandom, it is very far from random.
Consider G that doubles its input length: $G(s_1 \cdots s_n) = r_1, \ldots, r_{2n}$.
Recall, G is deterministic, so it can only map each input seed to a single output value.



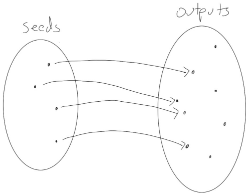
If we allow an exponential time adversary, *every* PRG is insecure!

$\underline{\mathcal{A}:}$
Let $S = \emptyset$
For each $s \in \{0,1\}^n$
$\quad S = S \cup G(s)$.
If $y \in S$ output 0
else output 1.

$$\begin{aligned}
\Pr[\mathrm{PrivK}^{\mathrm{prg}}_{\mathcal{A},\mathrm{G}}(n) = 1] &= \Pr[b' == b] \\
&= \Pr[b' = 0 \wedge b = 0] + \Pr[b' = 1 \wedge b = 1] \\
&= \Pr[b' = 0 \mid b = 0] \cdot \frac{1}{2} + \Pr[b' = 1 \mid b = 1] \cdot \frac{1}{2}
\end{aligned}$$

Pseudorandom vs. Random

Even when G is pseudorandom, it is very far from random.
Consider G that doubles its input length: $G(s_1 \cdots s_n) = r_1, \ldots, r_{2n}$.
Recall, G is deterministic, so it can only map each input seed to a single output value.



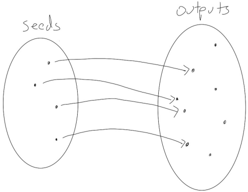
If we allow an exponential time adversary, *every* PRG is insecure!

$\underline{\mathcal{A}}$:
Let $S = \emptyset$
For each $s \in \{0,1\}^n$
    $S = S \cup G(s)$.
If $y \in S$ output 0
else output 1.

$$\begin{aligned}
\Pr[\mathsf{PrivK}^{\mathsf{prg}}_{\mathcal{A},\mathsf{G}}(n) = 1] &= \Pr[b' == b] \\
&= \Pr[b' = 0 \wedge b = 0] + \Pr[b' = 1 \wedge b = 1] \\
&= \Pr[b' = 0 \mid b = 0] \cdot \frac{1}{2} + \Pr[b' = 1 \mid b = 1] \cdot \frac{1}{2} \\
&= 1 \cdot \frac{1}{2} + (1 - 2^{-n}) \cdot \frac{1}{2}
\end{aligned}$$

Pseudorandom vs. Random

Even when G is pseudorandom, it is very far from random.
Consider G that doubles its input length: $G(s_1 \cdots s_n) = r_1, \ldots, r_{2n}$.
Recall, G is deterministic, so it can only map each input seed to a single output value.



If we allow an exponential time adversary, *every* PRG is insecure!

$\underline{\mathcal{A}:}$
Let $S = \emptyset$
For each $s \in \{0,1\}^n$
    $S = S \cup G(s)$.
If $y \in S$ output 0
else output 1.

$$\begin{aligned}
\Pr[\mathsf{PrivK}^{\mathsf{prg}}_{\mathcal{A},\mathsf{G}}(n) = 1] &= \Pr[b' == b] \\
&= \Pr[b' = 0 \wedge b = 0] + \Pr[b' = 1 \wedge b = 1] \\
&= \Pr[b' = 0 \mid b = 0] \cdot \frac{1}{2} + \Pr[b' = 1 \mid b = 1] \cdot \frac{1}{2} \\
&= 1 \cdot \frac{1}{2} + (1 - 2^{-n}) \cdot \frac{1}{2} = 1 - \frac{1}{2} 2^{-n} = 1 - 2^{-n-1}
\end{aligned}$$