

Andrew Johnson

CS 427 Crypto

Homework #2

1) Which of the following are negligible functions? Justify your answers.

$$f(\lambda) = \sqrt{\frac{\lambda}{2\lambda}}, \quad p(\lambda) = \lambda^c \quad \lim_{\lambda \rightarrow \infty} \frac{p(\lambda)}{f(\lambda)} = \lim_{\lambda \rightarrow \infty} \frac{\lambda^c}{\sqrt{\frac{\lambda}{2\lambda}}} = \lim_{\lambda \rightarrow \infty} \frac{\lambda^{c+\frac{1}{2}}}{2^{\lambda/2}} = 0$$

We see that the denominator of the final fraction in our limit will head towards infinity faster than the numerator. This causes the limit to look like, $\frac{1}{\infty} = 0$. Thus this function is negligible \square

$$f(\lambda) = \frac{1}{2^{\log(\lambda^2)}}, \quad p(\lambda) = \lambda^c \quad \lim_{\lambda \rightarrow \infty} \frac{p(\lambda)}{f(\lambda)} = \lim_{\lambda \rightarrow \infty} \frac{\lambda^c}{2^{\log(\lambda^2)}} = \begin{cases} \infty, & \text{if } c > 0 \\ 0, & \text{if } c \leq 0 \end{cases}$$

We see that if we use a polynomial with a power greater than zero then we get something that is not equal to zero, thus making this equation non-negligible \square

$$f(\lambda) = \frac{1}{\lambda^{\log(\lambda)}}, \quad p(\lambda) = \lambda^c \quad \lim_{\lambda \rightarrow \infty} \frac{p(\lambda)}{f(\lambda)} = \lim_{\lambda \rightarrow \infty} \frac{\lambda^c}{\lambda^{\log(\lambda)}} = \lim_{\lambda \rightarrow \infty} \lambda^{c-\log(\lambda)} = 0$$

Since $c < \infty$, and $\log(\lambda) \rightarrow \infty$ as $\lambda \rightarrow \infty$, then our final equation gives $\lambda^{-\infty} = 0$. This makes the given equation negligible \square

$$f(\lambda) = \frac{1}{\lambda^2}, \quad p(\lambda) = \lambda^c \quad \lim_{\lambda \rightarrow \infty} \frac{p(\lambda)}{f(\lambda)} = \lim_{\lambda \rightarrow \infty} \frac{\lambda^c}{\lambda^2} = \lim_{\lambda \rightarrow \infty} \lambda^{c-2} = \begin{cases} \infty, & \text{if } c > 2 \\ -\infty, & \text{if } c < 2 \\ 1, & \text{if } c = 2 \end{cases}$$

We can see that there is no option of this function being able to go to 0. So this function is non-negligible \square

$$f(\lambda) = \frac{1}{2^{\log(\lambda)^2}}, \quad p(\lambda) = \lambda^c \quad \lim_{\lambda \rightarrow \infty} \frac{p(\lambda)}{f(\lambda)} = \lim_{\lambda \rightarrow \infty} \frac{\lambda^c}{2^{\log(\lambda)^2}} = \lim_{\lambda \rightarrow \infty} \frac{2^{c \cdot \log(\lambda)}}{2^{\log(\lambda)^2}} = \lim_{\lambda \rightarrow \infty} 2^{(c-\log(\lambda)) \cdot \log(\lambda)} = 0$$

If we look at $(c - \log(\lambda)) \cdot \log(\lambda)$ as $\lambda \rightarrow \infty$, we see that it goes towards $-\infty$. This is why the limit goes to 0. Proving that this equation is negligible \square

$$f(\lambda) = \frac{1}{\log(\lambda)^2}, \quad p(\lambda) = \lambda^c \quad \lim_{\lambda \rightarrow \infty} \frac{p(\lambda)}{f(\lambda)} = \lim_{\lambda \rightarrow \infty} \frac{\lambda^c}{\log(\lambda)^2} = \begin{cases} \infty, & \text{if } c > 0 \\ 0, & \text{if } c \leq 0 \end{cases}$$

We see that if $c > 0$, then our final function doesn't go to 0. Thus this function is non-negligible \square

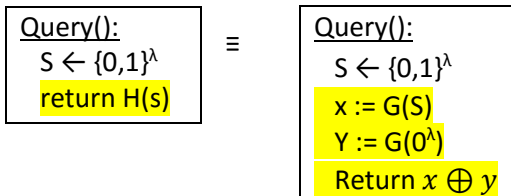
$$f(\lambda) = \frac{1}{\sqrt{\lambda}}, \quad p(\lambda) = \lambda^c \quad \lim_{\lambda \rightarrow \infty} \frac{p(\lambda)}{f(\lambda)} = \lim_{\lambda \rightarrow \infty} \frac{\lambda^c}{\sqrt{\lambda}} = \lim_{\lambda \rightarrow \infty} \lambda^{c-0.5} = \begin{cases} \infty, & \text{if } c > 0.5 \\ -\infty, & \text{if } c < 0.5 \\ 1, & \text{if } c = 0.5 \end{cases}$$

Just like in one of the previous functions. There is no way for this function to go to 0, so this function is non-negligible \square

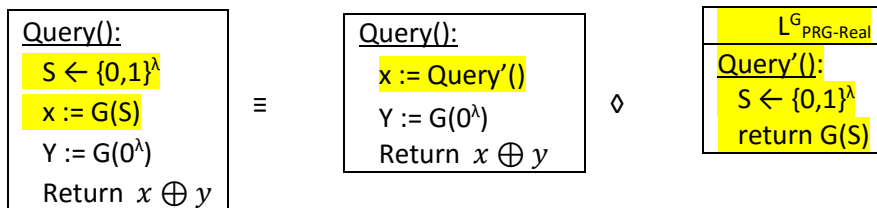
$$f(\lambda) = \frac{1}{2^{\sqrt{\lambda}}}, \quad p(\lambda) = \lambda^c \quad \lim_{\lambda \rightarrow \infty} \frac{p(\lambda)}{f(\lambda)} = \lim_{\lambda \rightarrow \infty} \frac{\lambda^c}{2^{\sqrt{\lambda}}} = \lim_{\lambda \rightarrow \infty} \frac{2^{c \cdot \log(\lambda)}}{2^{\sqrt{\lambda}}} = \lim_{\lambda \rightarrow \infty} 2^{c \cdot \log(\lambda) - \sqrt{\lambda}} = 0$$

We look at $\lambda \rightarrow \infty$ and notice that $c \cdot \log(\lambda) - \sqrt{\lambda} \rightarrow -\infty$. This makes the final equation always 0, thus making our given function negligible \square

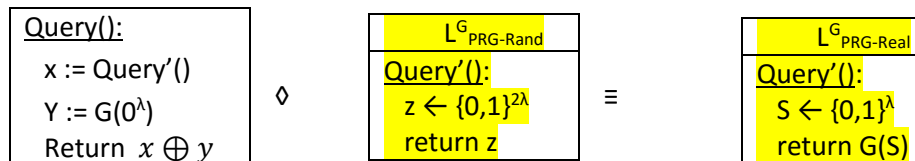
2) We take that we know G is a secure PRG.



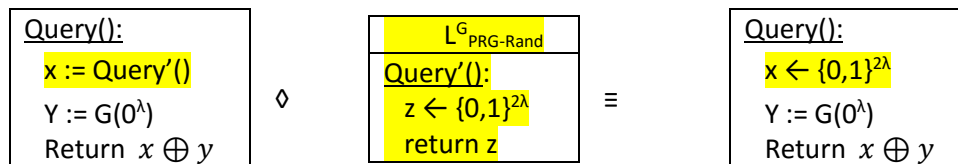
For the first movement we fill in details of H(s).



Factor out the terms of $L^G_{\text{PRG-Real}}$

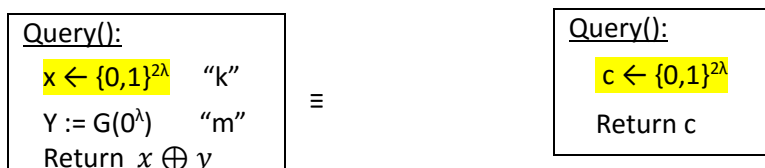


Since G is a secure PRG we can replace $L^G_{\text{PRG-Real}}$ with $L^G_{\text{PRG-Rand}}$



Now we inline our Query'

We can't do the same thing to y since it doesn't pass a parameter that is uniformly chosen. Although it now resembles OTP security of 2λ -bits, where x is our uniformly chosen 2λ -bit code and y will be our plaintext of 2λ -bits, since $G(0^\lambda)$ returns 2λ -bits. Thus we can say



This finishes our proof of security saying that

$$\begin{array}{|l}
 \text{Query():} \\
 S \leftarrow \{0,1\}^\lambda \\
 x := G(S) \\
 Y := G(0^\lambda) \\
 \text{Return } x \oplus y
 \end{array}
 \equiv
 \begin{array}{|l}
 \text{Query():} \\
 c \leftarrow \{0,1\}^{2\lambda} \\
 \text{Return } c
 \end{array}$$

$$L^{H(s)}_{\text{PRG-Real}} \qquad L^{H(s)}_{\text{PRG-Rand}}$$

3) Assuming $L^{H(s)}_{\text{PRG-Real}}$ and $L^{H(s)}_{\text{PRG-Rand}}$ look like the libraries below, respectively,

$$\begin{array}{|l}
 \text{H(s):} \\
 X := G(s) \\
 \text{Return } s \parallel x
 \end{array}
 \qquad
 \begin{array}{|l}
 \text{H(s):} \\
 c \leftarrow \{0,1\}^{3\lambda} \\
 \text{Return } c
 \end{array}$$

All we need to break which of these libraries we are in is to pass 0^λ through H and you should expect a string starting with λ 0's and then the rest of the 3λ -bit string. If you don't then you know you are in the RAND library. With our input there is a $\frac{1}{2^{2\lambda}}$ for REAL and $\frac{1}{2^{3\lambda}}$ for the RAND

So the adversary just has to compare the first λ -bits of c , and return $s == c^{\text{First } \lambda}$.

4) How we can break the function F' . If we input a string so x' is the string of all zeros then the real library will always have an output of $F(k,x) \parallel F(k,x)$. Where the random library would just output any random $2n$ bit string. So REAL has probability of $\frac{1}{2^{2n}}$, and where Rand would have a probability of $\frac{1}{2^{2n}}$. Which are different so F' is insecure.

So the Adversary would call the function with x as any n -bit string and x' be the n -bit string of 0's. Then have it return whether $c == x \parallel x$.