16-124-836
Marcel Zauder

3.1 Nested Encription Scheme

Prove that Σ satisfies one-time secrecy, then so does Σ^2:

The given library of function on the exercise sheet will be called L_2 of Σ^2.
We know that Σ satisfies the one-time secrecy with Σ.Enc(k,m) = c, with library L_1.
It is clear that $L_{OTS-L} \equiv L_{OTS-R}$, which means $Pr[A \diamond L_{OTS-L} \rightarrow 1] \ = \ Pr[A \diamond L_{OTS-R} \rightarrow 1]$, for any A.
The scheme Σ is used to encrypt m_L and m_R into c_L and c_R which are distributed equally. These encrypted ciphertexts are then encrypted again to get c_{L2} and c_{R2} which are still equally distributed. The used Library will be called $L_1' \diamond L_1$.
$L_1' \diamond L_1$ will satisfy one-time secrecy because $L_{OTS-L} \equiv L_{OTS-R}$ with an appropriate Eavesdrop(m_L, m_R).
Because in $L_1' \diamond L_1$ and L_2 the same is done, the produced ciphertexts are distributed the same (equally) and therefore one-time secrecy is given in L_2.

3.2 Negligible Functions

3.2.a

1. $\frac{1}{2^{\frac{\lambda}{2}}}$ Negligible?:
   It is negligible because: $\lim_{\lambda\rightarrow\infty}(p(\lambda) \cdot \frac{1}{2^{\frac{\lambda}{2}}}) \ = \ 0$

2. $\frac{1}{\lambda^2}$ Negligible?:
   No, it is not negligible because for $p(\lambda) = \lambda^2$ we have:
   $\lim_{\lambda\rightarrow\infty}(p(\lambda) \cdot \frac{1}{\lambda^2}) \ = \ \lim_{\lambda\rightarrow\infty}(\lambda^2 \cdot \frac{1}{\lambda^2}) \ = \ 1$

3. $\frac{1}{\lambda^{\frac{1}{\lambda}}}$ Negligible?:
   No, it is not because: $\lim_{\lambda\rightarrow\infty}(p(\lambda) \cdot \frac{1}{\lambda^{\frac{1}{\lambda}}}) \ = \ \lim_{\lambda\rightarrow\infty}(p(\lambda) \cdot \frac{1}{\underbrace{\lambda^{\frac{1}{\lambda}}}_{1}}) \ = \ \lim_{\lambda\rightarrow\infty}(p(\lambda)) \neq 0 \ \forall p(\lambda)$

4. $\frac{1}{\sqrt{\lambda}}$ Negligible?:
   No, it is not negligible because for $p(\lambda) = \sqrt{\lambda}$ we have:
   $\lim_{\lambda\rightarrow\infty}(p(\lambda) \cdot \frac{1}{\sqrt{\lambda}}) \ = \ \lim_{\lambda\rightarrow\infty}(\sqrt{\lambda} \cdot \frac{1}{\sqrt{\lambda}}) \ = \ 1$

5. $\frac{1}{2^{\sqrt{\lambda}}}$ Negligible?:
   It is negligible because: $\lim_{\lambda\rightarrow\infty}(p(\lambda) \cdot \frac{1}{2^{\sqrt{\lambda}}}) \ = \ 0$

3.2.b

- f(), g() are negligible $\Rightarrow$ f() $\cdot$ g() is negligible?
  We know:

$$\begin{aligned}
\lim_{\lambda\to\infty}(p(\lambda)\cdot f(\lambda)) &= 0\\
\lim_{\lambda\to\infty}(p(\lambda)\cdot g(\lambda)) &= 0\\
\Rightarrow \lim_{\lambda\to\infty}(p(\lambda)\cdot(f(\lambda)\cdot g(\lambda))) &= \lim_{\lambda\to\infty}((p(\lambda)\cdot f(\lambda))\cdot g(\lambda))\\
&= \underbrace{\lim_{\lambda\to\infty}(p(\lambda)\cdot f(\lambda))}_{0}\cdot\underbrace{\lim_{\lambda\to\infty}(g(\lambda))}_{0}\\
&= 0 \qquad \square
\end{aligned}$$

- Example s.t. f() and g() are negligible but $\frac{f()}{g()}$ is not:
  If $f() = g() = \frac{1}{2^\lambda}$, both are clearly negligible, but $\frac{f()}{g()} = \frac{\frac{1}{2^\lambda}}{\frac{1}{2^\lambda}} = \frac{2^\lambda}{2^\lambda} = 1$ is clearly not.

3.3 Hashrate

3.3.a CPU with 2GHz

Assuming you have one Intel CPU with 2GHz clock speed, how many cycles per block can one have in case of a single-threaded AVX1 implementation? How much is the hash rate?

- 1Mb = 1'000'000 bytes
- 2Ghz = 1 * 10^9 Hz (cycles/sec)
- From the given paper we can assume that the performance of SHA-256 will be most likely be constant at 12.8 cycles/byte

Therefore we have:

$$\begin{aligned}
12.8\ \frac{cycles}{byte}\times 1'000'000\ bytes &= 12'800'000\ cycles\\
12'800'000\ cycles\ \div\ 2'000'000'000\ \frac{cycles}{sec} &= 0.0064\ sec\\
1\ sec\ \div\ 0.0064\ sec &= 156.25\ hashes\ per\ second
\end{aligned}$$

3.3.b Bitcoin

Current hashrate is 93'241'227 * 10^{12} hashes per second (3.10.2019 2:00)

$$93'241'227\ *\ 10^{12}\ \div\ 156.25\ \approx\ 6\ *10^{17}$$

So $\sim 6\ *10^{17}$ such CPUs are needed to compute the current hash rate of bitcoin.

3.4 A Random Cipher

3.4.a Description

Σ.M = Σ.C = $\{0,1\}^{\kappa}$
Σ.K = $\{0,1\}^{?}$

Σ.KeyGen() = k ← $\{0,1\}^{?}$, Σ.Enc(k,m): c = ??? return c , Σ.Dec(k,c): m = ??? return m

Σ.Enc(k,m)	Σ.Dec(k,c)
c = ???	m = ???
return c	return m

3.4.b Upper Bound

The chance to guess m randomly out of c is:

$$\begin{aligned} P[A(c) \Rightarrow m] &= 1 - (1 - \frac{q}{2^k}) \textit{ invers of guessing q-times false.} \\ &= \frac{q}{2^k} \end{aligned}$$

For $q \rightarrow 2^k$ the probability gets to 1.