Cryptography

Academic Year 2024-2025

Homework 1

Michele Dinelli, ID 0001132338

October 17, 2024

Exercise 1.

We want to prove that any instance of an encryption scheme Π^G composed by three spaces $\mathcal{K}, \mathcal{M}, \mathcal{G}$ such that $\Pi^G = (Gen, Enc, Dec)$ with $Enc(k,m) = G(k) \oplus m$ and G is a pseudorandom generator is not perfectly secure. Let's check if for Π^G holds

$$|\mathcal{K}| \geq |\mathcal{M}|$$

If G is a pseudorandom generator then it means it's a deterministic algorithm that given as input $s \in \{0,1\}^n$ outputs a string $G(s) \in \{0,1\}^{\ell(|s|)}$ where ℓ is a polynomial defined as $\ell : \mathbb{N} \mapsto \mathbb{N}$. It's noticeable that G accepts at maximum 2^n inputs and because of that generates at maximum 2^n strings of length $l(|s|)$ while the number of possible strings of length $\ell(|s|)$ is $2^{l(|s|)}$. Since it is required for a pseudorandom generator that $\forall n \in \mathbb{N},\ \ell(n) > n$ we conclude that regarding Π^G we have

$$|\mathcal{K}| < |\mathcal{M}| \text{ since}$$
$$|\mathcal{K}| = 2^n \text{ and } |\mathcal{M}| = 2^{\ell(n)}$$

To prove that Π^G is not perfectly secure, we must rely on the property of pseudorandom generators that ensures the output length $\ell(n)$ exceeds the input length n, where $\ell : \mathbb{N} \to \mathbb{N}$ is a polynomial such that $\forall n \in \mathbb{N},\ \ell(n) > n$.

Exercise 2.

The exercise consists of considering the following functions and demonstrating that none of them are pseudorandom generators.

$$G_1(x) = x \cdot \oplus_{i=1}^{|x|} x_i \qquad G_2(x) = F(0^{|x|}, x) \qquad G_3(x) = F(x,x) \cdot x$$

- $G_1(x)$ is not a pseudorandom generator because it is easily distinguishable from a true random source. To prove that $G_1(x)$ is not a pseudorandom generator we have to define a distinguisher D such that $|Pr(D(s) = 1) - Pr(D(G(r)) = 1)$ is not negligible. We define D as follows:

$$\begin{aligned}
&D(x): \\
&w \leftarrow \oplus_{i=1}^{|x-1|} x_i; \\
&z = x|_{|x-1|}; \\
&\text{Return 1 if } (w \cdot z) = x
\end{aligned}$$

  Observing that

$$
\begin{aligned}
\Pr(D(G_1(r)) = 1) &= 1 \\
\Pr(D(s) = 1) &= \frac{1}{2^n} \\
|\Pr(D(G_1(r)) = 1) - \Pr(D(s) = 1)| &= 1 - \frac{1}{2^n} \\
&\text{which is not negligible}
\end{aligned} \tag{1}
$$

- $G_2(x)$ uses a pseudorandom function to produce its output. $F(k, x)$ should take k among all strings of length $|x|$ randomly. Defining $F(k, x)$ *with* $k = 0^{|x|}$ means that the key is fixed and that is not ideal. Furthermore, if F is a pseudorandom function than it is length preserving i.e $F(k, x)$ is defined iff $|k| = |x|$ and in that case $|F(k, x)| = |x|$. Just observing that we can conclude that $G_2(x)$ is not a pseudorandom function since its expansion factor does not satisfy $\ell(n) > n,\ \forall n \in \mathbb{N}$. $G_2(x)$ does not expand x in any way ($G_2(x) \in \{0, 1\}^{|x|}$).

- $G_3(x)$ is not a pseudorandom function because from the output of $G_3(x)$ could be immediately extracted the original input x from the last n bits. To prove that $G_3(x)$ is not a pseudorandom generator we have to define a distinguisher D such that $|Pr(D(s) = 1) - Pr(D(G(r)) = 1)$ is not negligible. We define D as follows:

$$
\begin{aligned}
&D(x): \\
&n \leftarrow \ell^{-1}(|x|); \\
&z \leftarrow \text{last } n \text{ bits of } x; \\
&o \leftarrow \mathcal{O}(z); \\
&\text{Return 1 if } o = x|_n
\end{aligned}
$$

where $\mathcal{O}$ is an oracle for F. D extracts the last $n = |x|$ bits of $G_3(x)$ and query an oracle on F with them. Then check if the output of the oracle is equal to the first n bits of x. If are equal D can distinguish between a true random and $G_3(x)$ in a similar way to eq. 1.

$$
\begin{aligned}
\Pr(D(G_3(r)) = 1) = 1 \text{ if } D \text{ receives } G_3(x) \text{ so the oracle behaves like } F(x, x)& \\
\Pr(D(s) = 1) &= \frac{1}{2^n} \\
|\Pr(D(G_3(r)) = 1) - \Pr(D(s) = 1)| &= 1 - \frac{1}{2^n} \\
&\text{which is not negligible}
\end{aligned} \tag{2}
$$

The exercise 2 asks also to prove that none of the following binary functions is a pseudorandom function.

$$
F_1(k, x) = k \oplus x \qquad F_2(k, m) = G(m)|_{|k|} \qquad F_3(x) = G(k)|_{|m|}
$$

- $F_1(k, x)$ is not a pseudorandom function because a distinguisher D can trivially recover the key xoring the output of $F_1(k, x)$ with x i.e $k = F_1(k, x) \oplus x$. To prove that $F_1(k, x)$ is not a pseudorandom function we have to define a distinguisher D such that $|\Pr(D^{F_k(\cdot)}(1^n) = 1) - \Pr(D^{f(\cdot)}(1^n) = 1)|$ is not negligible.

$$
\begin{aligned}
&D(1^n): \\
&m_0 \leftarrow 0^n; \\
&m_1 \leftarrow 1^n; \\
&o_0, o_1 \leftarrow \mathcal{O}(m_0), \mathcal{O}(m_1); \\
&w \leftarrow o_0 \oplus o_1; \\
&\text{Return 1 if } w = 1^n
\end{aligned}
$$

D can query the oracle $\mathcal{O}$ twice with $m_0 = 0^n$ and $m_1 = 1^n$ obtaining o_1 and o_2. Then D checks if $o1 \oplus o2 = 1^n$ returning 1 if it is the case since $(k \oplus 0^n) \oplus (k \oplus 1^n) = 1^n$.

$$
\begin{aligned}
\Pr(D^{F_k(\cdot)}(1^n) = 1) &= 1 \\
\Pr(D^{f(\cdot)}(1^n) = 1) &= \frac{1}{2^n} \\
|\Pr(D^{F_k(\cdot)}(1^n) = 1) - \Pr(D^{f(\cdot)}(1^n) = 1)| &= 1 - \frac{1}{2^n} \\
\text{which is not negligible}
\end{aligned}
\tag{3}
$$

- $F_2(k, x)$ outputs a string of length $|m|$ since $|k| = |m|$ for pseudorandom functions so the key is kind of useless. To prove that $F_2(k, x)$ is not a pseudorandom function we have to define a distinguisher D such that $|\Pr(D^{F_k(\cdot)}(1^n) = 1) - \Pr(D^{f(\cdot)}(1^n) = 1)|$ is not negligible.

$$
\begin{aligned}
&D(1^n): \\
&m \leftarrow 1^n \\
&o \leftarrow \mathcal{O}(m); \\
&g \leftarrow G(m); \\
&\text{Return 1 if } o = g
\end{aligned}
$$

D query an oracle with an arbitrary message m then query $G(m)$ alone since we assume it is public for Kerckhoffs' principle. D compares the output of the oracle with $G(m)$, if are equals then D distinguishes with very high probability $F_2(k, m)$ from a true random string s. What could happen is that $G(m)$ outputs the same value of a true random f but it is very unlikely.

$$
\begin{aligned}
\Pr(D^{F_k(\cdot)}(1^n) = 1) &= 1 \\
\Pr(D^{f(\cdot)}(1^n) = 1) &= \frac{1}{2^n} \\
|\Pr(D^{F_k(\cdot)}(1^n) = 1) - \Pr(D^{f(\cdot)}(1^n) = 1)| &= 1 - \frac{1}{2^n} \\
\text{which is not negligible}
\end{aligned}
\tag{4}
$$

- $F_3(k, x)$ could not be a pseudorandom function because returns always a pseudorandom generator applied to the key ignoring the message. As long as $|m|$ remains constant, $F_2(k, x)$ will always output the same truncated portion of $G(k)$, which is trivial to distinguish from a truly random function. To prove that $F_3(k, x)$ is not a pseudorandom function we have to define a distinguisher D such that $|\Pr(D^{F_k(\cdot)}(1^n) = 1) - \Pr(D^{f(\cdot)}(1^n) = 1)|$ is not negligible.

$$\begin{aligned}
&D(1^n):\\
&m_0, m_1 \leftarrow 1^n, 0^n;\\
&o_0, o_1 \leftarrow \mathcal{O}(m_0), \mathcal{O}(m_1);\\
&\text{Return 1 if } o_0 = o_1
\end{aligned}$$

A distinguisher D could query an oracle two times with 2 arbitrary messages m_0 and m_1, then checks if the results are equals returning 1 and distinguishing $F_3(k, x)$ from a true random f with very high probability in a similar way to eq. 3 and 4.

Exercise 3.

Given $\Pi = (Gen, Enc, Dec)$, $\Pi_{H,J} = (Gen, Enc, Dec)$, H, G two permutations (bijective and inversible) and $Enc_{H,J}(k, m)$ and$Dec_{H,J}(k, c)$ are defined as follows.

$$Enc_{H,J}(k, m) = J(Enc(k, H(m))) \qquad Dec_{H,J}(k, c) = H^{-1}(Dec(k, J^{-1}(c)))$$

It is required to prove that if Π is correct and secure against passive attacks, then $\Pi_{H,J}$ is also correct and secure against passive attacks. On the correctness of $\Pi_{H,J}$ we can observe that:

$$\begin{aligned}
c = J(Enc(k, H(m))) &\text{ encrypt m}\\
&\text{substitute c in the following}\\
Dec_{H,J}(k, c) = H^{-1}(Dec(k, J^{-1}(c))) &\text{ obtaining}\\
Dec_{H,J}(k, c) = H^{-1}(Dec(k, J^{-1}(J(Enc(k, H(m)))))) &\\
&\text{J is bijective so we can rewrite as}\\
Dec_{H,J}(k, c) = H^{-1}(Dec(k, (Enc(k, H(m))))) &\\
\text{since } Dec(k, (Enc(k, H(m))))) = H(m) &\text{ by the correctness of } \Pi\\
Dec_{H,J}(k, c) = H^{-1}(H(m)) &\text{ which is } m \text{ since } H \text{ is bijective.}
\end{aligned}$$

$\Pi_{H,J}$ is correct.

On the security against passive attacks of $\Pi_{H,J}$. Let's proceed with a reduction proof: the goal is to show that if the transformed scheme $\Pi_{H,J}$ can be broken, then the original scheme Π can be broken. We want to prove that

$$\Pi \text{ correct and secure against } eav \;\Rightarrow \Pi_{H,J} \text{ correct and secure against } eav$$

We can try to build an adversary that succeeds in breaking $\Pi_{H,J}$ and use it as a subroutine to build an adversary that succeeds in breaking Π.

$$\begin{gathered}
\forall B \in PPT.\neg BRK(B, \Pi) \Rightarrow \forall A \in PPT.\neg BRK(A, \Pi_{H,J})\\
\Downarrow\\
\exists A \in PPT.BRK(A, \Pi_{H,J}) \Rightarrow \exists B \in PPT.BRK(B, \Pi)
\end{gathered}$$

Let's look at the experiment $\mathsf{PrivK}^{eav}_{B,\Pi}$ defined using $\mathcal{A}$ as a subroutine (pseudocode 1 and pseudocode 2). The adversary $\mathcal{B}$ interacts with the original encryption scheme Π, but it internally uses $\mathcal{A}$ to distinguish between encrypted messages. $\mathcal{B}$ transforms the messages by applying the bijection H, and it transforms the ciphertexts by applying J^{-1} before passing them to $\mathcal{A}$. Thus, $\mathcal{B}$ simulates the environment of $\Pi_{H,J}$ for $\mathcal{A}$, making $\mathcal{A}$ think it is interacting with $\Pi_{H,J}$ when in fact it is interacting with Π. The adversary $\mathcal{B}$ succeeds in breaking the security of Π whenever $\mathcal{A}$ succeeds in breaking $\Pi_{H,J}$.

Algorithm 1 $\mathsf{PrivK}^{eav}_{A,\Pi_{H,J}}$

$k \leftarrow Gen(1^n)$
$m_0, m_1 \leftarrow \mathcal{A}(1^n)$
if $|m_0| \neq |m_1|$ **then**
  return 0
end if
$b \leftarrow \{0,1\}$
$c \leftarrow Enc_{\Pi_{H,J}}(k, m)$
$b^* \leftarrow \mathcal{A}(c)$
return $\neg(b^* \oplus b)$

Algorithm 2 $\mathsf{PrivK}^{eav}_{B,\Pi}$

$k \leftarrow Gen_{\Pi}(1^n)$
$m_0, m_1 \leftarrow \mathcal{B}(1^n)$
$m_0, m_1 \leftarrow H(m_0), H(m_1)$
if $|m_0| \neq |m_1|$ **then**
  return 0
end if
$b \leftarrow \{0,1\}$
$c \leftarrow J^{-1}(Enc_{\Pi}(k, m_b))$ ▷ Notice H has been applied to m_b
$b^* \leftarrow \mathcal{A}(c)$
return $\neg(b^* \oplus b)$

Π is assumed to be secure against passive attacks i.e $Pr(\mathsf{PrivK}^{eav}_{B,\Pi}) = \frac{1}{2} + \epsilon(n)$ where $\epsilon(n)$ is negligible, the same for the experiment $\mathsf{PrivK}^{eav}_{A,B}$ since the two experiments are basically the same. Since the existence of an adversary $\mathcal{A}$ that breaks $\Pi_{H,J}$ implies the existence of an adversary $\mathcal{B}$ that breaks Π, we conclude that if Π is secure against passive attacks, then $\Pi_{H,J}$ must also be secure against passive attacks.

On the security against cpa attacks of $\Pi_{H,J}$. If Π is probabilistic, then $\Pi_{H,J}$ can also be considered secure against CPA attacks, assuming that the transformations H and J do not undermine that property.