

Homework 1 Solutions

Problems

1. (40 Points) Consider the following variant of one-time perfect security, where Eve can obtain two ciphertexts encrypted under the same key, called **two-time perfect security**

We say that an encryption scheme is two-time perfectly secure if $\forall m_{11}, m_{12}, m_{21}, m_{22} \in \mathcal{M}$, the following distributions are identical:

- $\mathcal{D}_1 := \{c_1 := \text{Enc}(k, m_{11}), c_2 := \text{Enc}(k, m_{12}); k \leftarrow \text{KeyGen}()\}$
- $\mathcal{D}_2 := \{c_1 := \text{Enc}(k, m_{21}), c_2 := \text{Enc}(k, m_{22}); k \leftarrow \text{KeyGen}()\}$

Describe an attack demonstrating that one-time pad does **not** satisfy this security definition.

Solution. Consider an adversary who chooses $m_{11}, m_{12}, m_{21}, m_{22}$ such that $m_{11} = m_{12}$, while $m_{21} \neq m_{22}$. We then have that:

$$\begin{aligned} \Pr[c_1 = c_2 | \mathcal{D}_1] &= \Pr[m_{11} \oplus k = m_{12} \oplus k | \mathcal{D}_1] \\ &= \Pr[m_{11} \oplus k = m_{11} \oplus k | \mathcal{D}_1] \\ &= \Pr[m_{11} = m_{11} | \mathcal{D}_1] \\ &= 1 \end{aligned}$$

We also have:

$$\begin{aligned} \Pr[c_1 = c_2 | \mathcal{D}_2] &= \Pr[m_{21} \oplus k = m_{22} \oplus k | \mathcal{D}_2] \\ &= \Pr[m_{21} = m_{22} | \mathcal{D}_1] \\ &= 0 \end{aligned}$$

Therefore, $\Pr[c_1 = c_2 | \mathcal{D}_1] \neq \Pr[c_1 = c_2 | \mathcal{D}_2]$, and so the distributions are not equal.

Note. There are many different ways to attack this scheme. They should all have the same general form of showing that, for some chosen messages,

$$\Pr[A | \mathcal{D}_1] \neq \Pr[A | \mathcal{D}_2]$$

for some predicate A .

2. (60 Points) Consider an alternative security definition, called **key-privacy security**, that captures the following: a ciphertext should not reveal any information about the *key* that it was encrypted under. In other words, an adversary that tries to guess the key after seeing a ciphertext should do no better than random guessing. We formalize this into a security definition below.

We say that an encryption scheme satisfies *key-privacy security* if for all adversaries \mathcal{A} and for all messages $m \in \mathcal{M}$, we have

$$\Pr_{k \leftarrow \text{KeyGen}()} [\mathcal{A}(\text{Enc}(k, m)) = k] = \frac{1}{|\mathcal{K}|}.$$

- (a) (30 Points) Prove that key-privacy security does **not** imply one-time perfect security. Do so by defining an encryption scheme, proving that it satisfies key-privacy, and then proving that it does not satisfy one-time perfect security.
- (b) (30 Points) Prove that one-time perfect security does **not** imply key-privacy security. Do so by defining an encryption scheme, proving that it satisfies one-time perfect security, and then proving that it does not satisfy key-privacy.

Solution.

- (a) Consider the following encryption scheme:

$\text{KeyGen}(1^\lambda)$	$\text{Enc}(k, m)$	$\text{Dec}(k, c)$
$k \leftarrow \{0, 1\}^\lambda$	return m	return c
return k		

We begin by proving that the scheme satisfies correctness, and is therefore a valid encryption scheme. We have that $\forall k \in |\mathcal{K}|, \forall m \in \mathcal{M}$:

$$\begin{aligned} & \Pr[\text{Dec}(k, \text{Enc}(k, m)) = m] \\ &= \Pr[\text{Dec}(k, m) = m] \\ &= \Pr[m = 1] = 1 \end{aligned}$$

and so the scheme satisfies correctness.

We next prove that the scheme satisfies key-privacy. Consider an arbitrary adversary \mathcal{A} . We have that:

$$\begin{aligned} & \Pr_{k \leftarrow \text{KeyGen}()} [\mathcal{A}(\text{Enc}(k, m)) = k] \\ &= \sum_{k \in |\mathcal{K}|} \Pr[\mathcal{A}(\text{Enc}(k, m)) = k \mid k = \text{KeyGen}()] \cdot \Pr[k = \text{KeyGen}()] \quad (\text{law of total probability}) \\ &= \sum_{k \in |\mathcal{K}|} \Pr[\mathcal{A}(m) = k \mid k = \text{KeyGen}()] \cdot \Pr[k = \text{KeyGen}()] \quad (\text{construction}) \\ &= \sum_{k \in |\mathcal{K}|} \Pr[\mathcal{A}(m) = k \mid k = \text{KeyGen}()] \cdot \frac{1}{|\mathcal{K}|} \quad (\text{construction}) \\ &= \frac{1}{|\mathcal{K}|} \sum_{k \in |\mathcal{K}|} \Pr[\mathcal{A}(m) = k \mid k = \text{KeyGen}()] \\ &= \frac{1}{|\mathcal{K}|} \sum_{k \in |\mathcal{K}|} \Pr[\mathcal{A}(m) = k] \quad (\text{independence}) \\ &= \frac{1}{|\mathcal{K}|} \cdot 1 = \frac{1}{|\mathcal{K}|} \end{aligned}$$

The key transformation in the argument above is the one marked “independence.” As \mathcal{A} ’s view does not contain *any* information about k , the probability that it outputs k must be independent

from the value k takes in the experiment. We give the full proof here, but in the future proofs will automatically use the fact that $\Pr_{x \leftarrow \{0,1\}^\lambda}[\mathcal{A}(y) = x] = \frac{1}{2^\lambda}$ when y contains no information about x .

Next, consider the messages $m_0 = 0^\lambda$ and $m_1 = 1^\lambda$. We have that:

$$\Pr[\text{ct} = 0^\lambda \mid D_0] = 1 \neq 0 = \Pr[\text{ct} = 0^\lambda \mid D_1]$$

and so $D_0 \neq D_1$, and the scheme does not satisfy one-time perfect security.

- (b) We give two different ways to do this proof, the first under bullet i and the second under bullet ii. Let $s = (s_1, \dots, s_n)$ be an n -bit string. For any $1 \leq i \leq j \leq n$, the notation $s[i, \dots, j]$ refers to the contiguous substring (s_i, \dots, s_j) . For two strings x and y let $x||y$ denote their concatenation.
- Consider the following encryption scheme for messages of length λ :

$\text{KeyGen}(1^\lambda)$	$\text{Enc}(k, m)$	$\text{Dec}(k, c)$
$k \leftarrow \{0,1\}^{\lambda+1}$	$c := m \oplus k[1 \dots \lambda]$	return $k[1 \dots \lambda] \oplus c[1 \dots \lambda]$
return k	return $c' = c k_{\lambda+1}$	

We begin by proving that the scheme satisfies correctness and is therefore a valid encryption scheme. We have that $\forall k \in |\mathcal{K}|, \forall m \in \mathcal{M}$:

$$\begin{aligned} & \Pr[\text{Dec}(k, \text{Enc}(k, m)) = m] \\ &= \Pr[\text{Dec}(k, (m \oplus k[1 \dots \lambda])) || k_{\lambda+1}) = m] \\ &= \Pr[k[1 \dots \lambda] \oplus (m \oplus k[1 \dots \lambda]) = m] \\ &= \Pr[m = m] = 1 \end{aligned}$$

and so the scheme satisfies correctness.

We next prove that the scheme satisfies one-time perfect security.

We have that $\forall m_0, m_1 \in \{0,1\}^\lambda$, for any given ciphertext c in D_0 :

$$\begin{aligned} & \Pr[c' = c : \begin{array}{l} k \leftarrow \text{KeyGen}() \\ c' \leftarrow \text{Enc}(k, m_0) \end{array}] \\ &= \Pr[c' = c : \begin{array}{l} k \leftarrow \{0,1\}^{\lambda+1} \\ c' := (m_0 \oplus k[1 \dots \lambda]) || k_{\lambda+1} \end{array}] \\ &= \Pr[(m_0 \oplus k[1 \dots \lambda]) || k_{\lambda+1} = c : k \leftarrow \{0,1\}^{\lambda+1}] \\ &= \Pr[(m_0 \oplus c[1 \dots \lambda]) || c_{\lambda+1} = k : k \leftarrow \{0,1\}^{\lambda+1}] \\ &= \frac{1}{2^{\lambda+1}} \end{aligned}$$

An identical analysis applies for D_1 . We therefore have that $D_0 = D_1$, and so the scheme satisfies one-time perfect security.

Next, consider the following key-privacy adversary \mathcal{A} :

$\mathcal{A}(c)$
$k' \leftarrow \{0,1\}^\lambda$
return $k' c_{\lambda+1}$

We can now analyze \mathcal{A} 's advantage in the key-privacy game:

$$\begin{aligned}
& \Pr[\mathcal{A}(\text{Enc}(k, m)) = k : k \leftarrow \text{KeyGen}()] \\
&= \Pr[\mathcal{A}(c || k_{\lambda+1}) = k : k \xleftarrow{\$} \{0, 1\}^{\lambda+1}] && \text{(construction)} \\
&= \Pr[k' || k_{\lambda+1} = k : \begin{array}{l} k \xleftarrow{\$} \{0, 1\}^{\lambda+1} \\ k' \xleftarrow{\$} \{0, 1\}^\lambda \end{array}] && \text{(definition of } \mathcal{A} \text{)} \\
&= \Pr[k' = k[1 \dots \lambda] : \begin{array}{l} k \xleftarrow{\$} \{0, 1\}^{\lambda+1} \\ k' \xleftarrow{\$} \{0, 1\}^\lambda \end{array}] \\
&= \frac{1}{2^\lambda} \neq \frac{1}{2^{\lambda+1}} = \frac{1}{|\mathcal{K}|}
\end{aligned}$$

And so the scheme does *not* satisfy key-privacy security.

- ii. Consider the one-time pad encryption scheme. We have already shown that OTP is correct and satisfies one-time perfect security, and so all that remains to be shown is that it does not satisfy key-privacy security. Key privacy security requires that:

$$\forall \mathcal{A}, \forall m \in \mathcal{M}, \Pr_{k \leftarrow \text{KeyGen}()} [\mathcal{A}(\text{Enc}(k, m)) = k] = \frac{1}{|\mathcal{K}|}$$

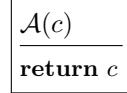
Therefore, to prove that OTP does *not* satisfy key-privacy security we must show:

$$\exists \mathcal{A}, \text{ s.t. } \exists m \in \mathcal{M} \text{ s.t. } \Pr_{k \leftarrow \text{KeyGen}()} [\mathcal{A}(\text{Enc}(k, m)) = k] \neq \frac{1}{|\mathcal{K}|}$$

or equivalently:

$$\exists m \in \mathcal{M} \text{ s.t. } \exists \mathcal{A} \text{ s.t. } \Pr_{k \leftarrow \text{KeyGen}()} [\mathcal{A}(\text{Enc}(k, m)) = k] \neq \frac{1}{|\mathcal{K}|}$$

In other words, the adversary we choose can depend on the message we choose when attacking the scheme. Let $m = 0^\lambda$, and consider the following adversary:



We then have that:

$$\begin{aligned}
& \Pr_{k \leftarrow \text{KeyGen}()} [\mathcal{A}(\text{Enc}(k, m)) = k] \\
&= \Pr_{k \leftarrow \text{KeyGen}()} [\mathcal{A}(k \oplus 0^\lambda) = k] \\
&= \Pr_{k \leftarrow \text{KeyGen}()} [\mathcal{A}(k) = k] \\
&= \Pr_{k \leftarrow \text{KeyGen}()} [k = k] \\
&= 1 \neq \frac{1}{|\mathcal{K}|}
\end{aligned}$$

and so OTP does not satisfy key-privacy security.