Exercise Sheet 2

Prof. Kenny Paterson, F. Günther
M. Backendal, M. Filić

Group Discussion (Vigenère and chosen plaintexts). Take some time to think about and discuss:

(a) What is, in brief, the difference between the Vigenère cipher and the one-time pad?

(b) Chosen-plaintext attacks, where an adversary gets to pick the input to, e.g., a block cipher seem quite hard to mount in practice. Why should we then care for such strong attacks? Can you think of examples of real-world chosen-plaintext attacks?

Suggested focus. Attempt these problems
before class: From Ex01: Remaining parts of Problem 4.
Problem 1, Problem 2 part (a), Problem 3 parts (a) and (b), Problem 4.
in class: Problem 2 part (b), Problem 3 part (c).

Suggested reading. Reading the following sections in the Boneh-Shoup book [1] might help with the problems on this exercise sheet: Sections 4.1.1-4.1.3 (block ciphers) and Sections 4.2.1-4.2.3 (DES).

Problem 1 (Strong PRP security). Let us recall the definition of PRP security of a block cipher E, in which an adversary $\mathcal{A}$ with access to an oracle $\textsc{Fn}$ is asked to distinguish a "real world" ($b = 0$) where $\textsc{Fn}(x)$ returns the evaluation $\mathsf{E}(\mathsf{K}, x)$ under a random key K from a "random world" ($b = 1$) where $\textsc{Fn}(x)$ returns the evaluation $\Pi(x)$ under a randomly sampled permutation $\Pi\colon \{0,1\}^n \to \{0,1\}^n$. The advantage of $\mathcal{A}$ in this game is defined as $\mathbf{Adv}^{\text{PRP}}_{\mathsf{E}}(\mathcal{A}) = 2 \cdot \left|\Pr\left[\text{Game } \mathbf{PRP}(\mathcal{A}, \mathsf{E}) \Rightarrow \mathsf{true}\right] - \frac{1}{2}\right|$, where $\Pr\left[\text{Game } \mathbf{PRP}(\mathcal{A}, \mathsf{E}) \Rightarrow \mathsf{true}\right]$ denotes the probability that the output of Game **PRP** is true.

Game $\mathbf{PRP}(\mathcal{A}, \mathsf{E})$:	Oracle $\textsc{Fn}(x)$:
1 $b \leftarrow\!\!\$\ \{0,1\}$	6 If $b = 0$ then:
2 $\mathsf{K} \leftarrow\!\!\$\ \{0,1\}^k$	7 $\quad y \leftarrow \mathsf{E}_{\mathsf{K}}(x)$
3 $\Pi \leftarrow\!\!\$\ \mathsf{Perms}[\{0,1\}^n]$	8 Else if $b = 1$ then:
4 $b' \leftarrow\!\!\$\ \mathcal{A}^{\textsc{Fn}}()$	9 $\quad y \leftarrow \Pi(x)$
5 Return $b' = b$	10 Return y

Let $\mathsf{E} : \{0,1\}^k \times \{0,1\}^n \to \{0,1\}^n$ be a block cipher.

(a) In the strong PRP (sPRP) security game, an adversary has access to the real block cipher or random permutation not only in the forward "encipher" direction (E), but also in the reverse "decipher" direction (E^{-1}).

1) Write out the definition sPRP security as a coded game and define the adversary's advantage function for the game.

2) Briefly argue why sPRP security implies PRP security. That is, show that sPRP is a *stronger* security notion than PRP.

(b) Let the function $\mathsf{F}_4 : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$ be defined by

$$\mathsf{F}_4(\mathsf{K}, x) = \mathsf{E}(\mathsf{K}, x) \oplus \mathsf{E}(\mathsf{K}, \mathsf{K}).$$

1) Show that F_4 is a block cipher.

2) Show that F_4 does *not* have strong-PRP security.

Problem 2 (Key recovery vs. pseudorandomness). In the lecture we settled for *pseudorandomness* (see Problem 1) as the desired security notion for block ciphers. Let us see that this notion also implies security against *key recovery* attacks (but not the other way around).

We begin by formalizing (target) key recovery security via the game **TKR** shown below. In this game, the adversary $\mathcal{A}$ has access to an oracle Fn which always returns the real evaluation $\mathsf{E}(\mathsf{K}, x)$ under a random key K, and the goal of the adversary is to find the key K. (I.e., instead of distinguishing E from a random permutation, guessing a hidden bit b, the adversary here outputs a key guess K' and wins if $\mathsf{K}' = \mathsf{K}$.) The advantage of $\mathcal{A}$ is accordingly defined as $\mathbf{Adv}^{\mathrm{TKR}}_{\mathsf{E}}(\mathcal{A}) = \Pr[\,\text{Game } \mathbf{TKR}(\mathcal{A}, \mathsf{E}) \Rightarrow \mathsf{true}\,]$.

Game $\mathbf{TKR}(\mathcal{A}, \mathsf{E})$	Oracle $\textsc{Fn}(x)$:
1 $\mathsf{K} \leftarrow\!\!\$\ \{0,1\}^k$	4 $y \leftarrow \mathsf{E}_{\mathsf{K}}(x)$
2 $\mathsf{K}' \leftarrow\!\!\$\ \mathcal{A}^{\textsc{Fn}}()$	5 Return y
3 Return $\mathsf{K}' = \mathsf{K}$	

(a) Let $k, n \geq 1$. Construct a block cipher $\mathsf{E}_{kr} : \{0,1\}^k \times \{0,1\}^n \to \{0,1\}^n$ such that

1) $\mathbf{Adv}^{\mathrm{TKR}}_{\mathsf{E}_{kr}}(\mathcal{A}) \leq 2^{-k}$ for all adversaries $\mathcal{A}$,

2) but $\mathbf{Adv}^{\mathrm{PRP}}_{\mathsf{E}_{kr}}(\mathcal{A}') = 1 - \frac{1}{2^n}$ for an adversary $\mathcal{A}'$ making only a single query to Fn.

Hint: Think of 1) as saying there is no way for an adversary to learn anything about the key K from $\mathsf{E}_{\mathsf{K}}(x)$, i.e., it is a permutation that does not leak any information about K.

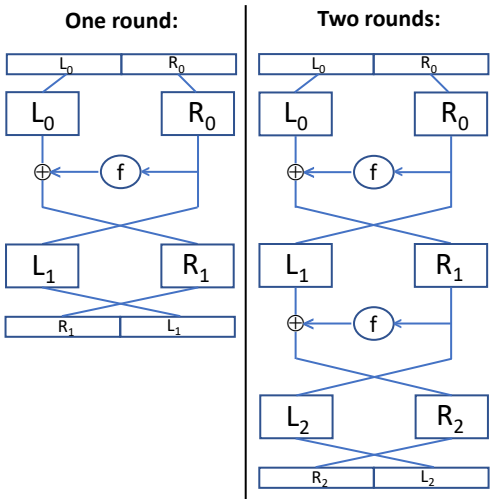
(b) Show that for any block cipher E, PRP security implies TKR security.

Hint: To prove this claim, assume that there is an efficient q-query adversary $\mathcal{A}$ against the TKR security of E and use $\mathcal{A}$ to build an efficient $(q+1)$-query adversary $\mathcal{B}$ against the PRP security of E such that you can relate $\mathbf{Adv}^{\mathrm{TKR}}_{\mathsf{E}}(\mathcal{A})$ to $\mathbf{Adv}^{\mathrm{PRP}}_{\mathsf{E}}(\mathcal{B})$. (The term "$q$-query adversary" refers to an adversary making q *distinct* queries. You may assume that $q \ll 2^n$ and that $\frac{1}{2^n}$ is sufficiently "small" for security.)

Hint: Try with an informal argument first, and then see if you can formalize it.

Problem 3 (Security of the Feistel cipher). A *Feistel cipher* is a common structural approach in the construction of block ciphers. It uses a so called *round function* to iteratively compute permutations of the input.

Consider the following graphical representations of the Feistel cipher for one round and two rounds using round function f. Note that the round function f also takes as input the key K and the round number r, but we omit these inputs in the picture.



Let $f\colon \mathbb{N} \times \{0,1\}^k \times \{0,1\}^m \to \{0,1\}^m$ be an arbitrary round function for some $k, m \in \mathbb{N}$. Let us define a function $\mathsf{F}_{f,\mathtt{t}}$ that runs $\mathtt{t}$ rounds of the Feistel cipher as follows:

Function $\mathsf{F}_{f,\mathtt{t}}(\mathsf{K}, M)$ // $|\mathsf{K}| = k$ and $|M| = 2m$
$L_0 \,\|\, R_0 \leftarrow M$ // $|L_0| = |R_0| = m$
For $r = 1$ to $\mathtt{t}$ do
  $L_r \leftarrow R_{r-1}$; $R_r \leftarrow L_{r-1} \oplus f(r, \mathsf{K}, R_{r-1})$
$C \leftarrow R_{\mathtt{t}} \,\|\, L_{\mathtt{t}}$
Return C

With this definition, the image above shows $\mathsf{F}_{f,1}$ (on the left) and $\mathsf{F}_{f,2}$ (on the right).

(a) Show that $\mathsf{F}_{f,1}$ is a block cipher.

(b) Show that $\mathsf{F}_{f,1}$ is not PRP secure.

(c) Show that $\mathsf{F}_{f,2}$ is not PRP secure. **Hint:** Build an adversary that makes two queries.

What is the advantage achieved by each of the adversaries that you built? The purpose is to attack the Feistel transform and not its round function, so your attacks should work no matter how f is defined.

Note: Luby and Rackoff [2] proved that for an appropriate choice of the round function (f in F) the 3-round Feistel cipher is PRP secure, and the 4-round Feistel cipher is strong-PRP secure.

Note 2: The DES block cipher runs 16 rounds of the Feistel cipher, and is defined as follows.

Function $\mathsf{DES}(\mathsf{K}, M)$ // $|\mathsf{K}| = 56$ and $|M| = 64$
$(\mathsf{K}_1, \ldots, \mathsf{K}_{16}) \leftarrow \text{KeySchedule}(\mathsf{K})$ // $|\mathsf{K}_i| = 48$ for $1 \le i \le 16$
$M \leftarrow \mathsf{IP}(M)$
$L_0 \,\|\, R_0 \leftarrow M$ // $|L_0| = |R_0| = 32$
For $r = 1$ to 16 do
  $L_r \leftarrow R_{r-1}$; $R_r \leftarrow L_{r-1} \oplus f(\mathsf{K}_r, R_{r-1})$
$C \leftarrow \mathsf{IP}^{-1}(R_{16} \,\|\, L_{16})$
Return C

It uses a key-schedule algorithm KeySchedule, round function f, initial permutation IP, and its inverse IP^{-1}.

Problem 4 (Meet-in-the-middle attack). For the DES block cipher (see Problem 3), many specific attacks and weaknesses have been found. The main issue, from the very start, was however its small key size of only 56 bits. To strengthen the security of DES, 3DES was introduced, executing DES three-times in forward, backward, and forward direction (EDE: encrypt-decrypt-encrypt) with two or three different keys.

In this question we will look at why three executions are indeed necessary to gain any reasonable security, by studying the security of (only) double encryption using a block cipher. Throughout, $(E_{\mathsf{K}}, D_{\mathsf{K}})$ denotes a block cipher with block-size n bits and key-size k bits.

(a) We start with exhaustive key search in the single encryption case, formalizing the cost analysis of exhaustive key search from the lectures. Suppose you have available a collection of t distinct plaintext-ciphertext pairs (P_i, C_i), $1 \le i \le t$, such that $C_i = E_{\mathsf{K}}(P_i)$ for each i.

1) Under the assumption that $E_{\mathsf{K}}(\cdot)$ behaves like a random permutation for each choice of key K, show that the expected number of keys K such that $C_1 = E_{\mathsf{K}}(P_1)$ is equal to 2^{k-n}.

2) Extend your analysis to show that the expected number of keys K such that $C_i = E_{\mathsf{K}}(P_i)$ for each $1 \le i \le t$ is approximately equal to 2^{k-tn}.

3) Explain how to use the pairs (P_i, C_i) to carry out an exhaustive key search. Evaluate the cost of the attack in terms of the number of calls needed to the encryption algorithm of the block cipher and the number of plaintext/ciphertext pairs t that are needed to uniquely identify the key. (A full answer should take into account that a single pass of the naive attack may not identify a unique key; t should be large enough to make the expected number of keys small, for example, much less than 1.)

(b) Now consider double encryption using the block cipher $(E_{\mathsf{K}}, D_{\mathsf{K}})$: two k-bit keys $\mathsf{K}_1, \mathsf{K}_2$ are chosen uniformly at random, and the encryption of an n-bit plaintext P is given by $C = E_{\mathsf{K}_2}(E_{\mathsf{K}_1}(P))$. That is, the block cipher is applied with key K_1, and then again with key K_2. You are again given t distinct plaintext-ciphertext pairs (P_i, C_i), $1 \le i \le t$, so that now $C_i = E_{\mathsf{K}_2}(E_{\mathsf{K}_1}(P_i))$ for each i.

1) Under the assumption that $E_{\mathsf{K}}(\cdot)$ behaves like a random permutation for each choice of K, show that the expected number of keys $(\mathsf{K}_1, \mathsf{K}_2)$ such that $C_i = E_{\mathsf{K}_2}(E_{\mathsf{K}_1}(P_i))$ for each i with $1 \le i \le t$ is approximately equal to 2^{2k-tn}. **Hint:** Think of $E_{\mathsf{K}_2}(E_{\mathsf{K}_1}(\cdot))$ as being a block cipher with a $2k$-bit key and apply a result from part (a).

2) How many pairs (P_i, C_i) are needed to uniquely identify the key $(\mathsf{K}_1, \mathsf{K}_2)$ in an exhaustive key search against double encryption? Express your answer in terms of k and n.

3) Let us now consider a meet-in-the-middle attack on double encryption. Let

$$X = \{(E_{\mathsf{K}}(P_1), E_{\mathsf{K}}(P_2), \ldots, E_{\mathsf{K}}(P_t)) : \mathsf{K} \in \{0,1\}^k\}$$

and

$$Y = \{(D_{\mathsf{K}}(C_1), D_{\mathsf{K}}(C_2), \ldots, D_{\mathsf{K}}(C_t)) : \mathsf{K} \in \{0,1\}^k\}.$$

That is, X is the set of all length t vectors whose components are the (single) block cipher encryptions of P_i, and Y is the set of all length t vectors whose components are the (single) block cipher decryptions of C_i. By considering the sets X and Y, show that the set of all candidate keys $(\mathsf{K}_1, \mathsf{K}_2)$ can be found using roughly $t \cdot 2^{k+1}$ block cipher operations and storage of 2^k tn-bit words. **Hint:** Think about matching vectors in the sets X and Y; what data structures can you use to efficiently record this information? What computation other than block cipher operations does your attack need, and what is the cost of that computation?

(c) Provide an assessment of the strength of double encryption, as compared to single encryption.

Acknowledgements. This exercise sheet is in part inspired by (and adapted from) problems by Mihir Bellare, as well as from the book "A Graduate Course in Applied Cryptography" by Dan Boneh and Victor Shoup, and from the book "The Joy of Cryptography" by Mike Rosulek.

References

[1] D. Boneh and V. Shoup. *A Graduate Course in Applied Cryptography.* Online, version 0.6 edition, Jan. 2023.

[2] M. Luby and C. Rackoff. How to construct pseudorandom permutations from pseudorandom functions. *SIAM Journal on Computing*, 17(2):373–386, 1988.