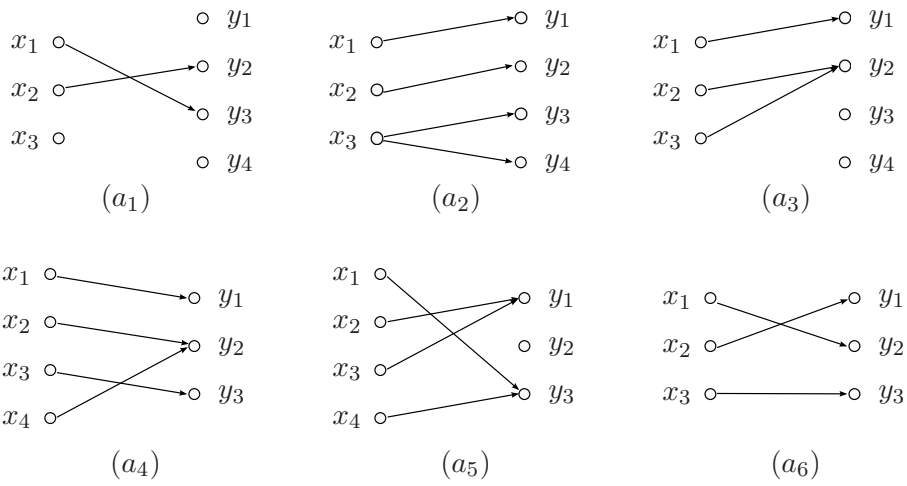
Problem Set 1
Introduction to Modern Cryptography
Coursera-Course (Summer 2018)

Bauhaus-Universität Weimar, Chair of Media Security
Problem Session: Eik List, Course: Prof Dan Boneh, Stanford University.
URL: `http://www.uni-weimar.de/de/medien/professuren/mediensicherheit/teaching/`

Due Date: Friday, 25 May 2018, 1:30 PM, via email to `eik.list(at)uni-weimar.de`.
Note: *For all tasks, explain your solutions in brief in your own words. We can not accept solutions without a clear explanation or calculation. The use of LaTeX is recommended; a LaTeX template file can be found on the website of the problem session.*

Question 1 – Recap: Functions and Permutations (7 Points)
Recall autonomously the terms function, injectivity, surjectivity, bijectivity, and permutation. Hereafter, let $\mathcal{X} = \{x_1, x_2, \ldots, x_n\}$ and $\mathcal{Y} = \{y_1, y_2, \ldots, y_m\}$ be two sets.

a) For each of the following six illustrations, briefly state if they represent a function or not; if yes, if they are injective, surjective, and/or bijective.



b) Let $F : \mathcal{X} \to \mathcal{Y}$ be a function. Can F be bijective if $n < m$? Can F be bijective if $n > m$?

c) Let F and G be arbitrary permutations over $\mathcal{X}$. Let $H(x) = G(F(x))$, for all $x \in \mathcal{X}$. Show *or* disprove: H is also a permutation over $\mathcal{X}$.

d) Let $F : \mathcal{X} \to \mathcal{X}$ be a function and let G be a permutation over $\mathcal{X}$. Show *or* disprove: $H(x) = G(F(x))$, for all $x \in \mathcal{X}$, is a permutation over $\mathcal{X}$.

e) Let $\{0,1\}^n$ be the set of all n-bit strings. Let F be a permutation over $\{0,1\}^n$. Show *or* disprove: for all inputs $x \in \{0,1\}^n$, the output of F, $y = F(x)$, has the same hamming weight as x.

Question 2 – Distinguishers (8 Points)
We write $x \twoheadleftarrow \mathcal{X}$ to say that some x has been chosen independently uniformly at random from some set $\mathcal{X}$. A random function $F : \mathcal{X} \to \mathcal{Y}$ returns for each input $x \in \mathcal{X}$ some random output $y \twoheadleftarrow \mathcal{Y}$ chosen uniformly at random and independently from all previous inputs. However, functions are deterministic: two calls to $F(x)$ with the same input x will always yield the same output y.
In the following, we say $F, F' : \{0,1\}^n \to \{0,1\}^n$ are independent random functions over n-bit strings. Below, you shall evaluate the security of some given functions G that use F and F' internally. G represents the real world. The ideal world is another random function $\$: \{0,1\}^n \to \{0,1\}^n$ that outputs random n-bit strings. We call G a secure PRF if its outputs can not be distinguished from that of \$. For each of the constructions G below, briefly specify an efficient distinguisher $\mathbf{A}$ that can distinguish the outputs of G from the outputs of \$ and specify the advantage of $\mathbf{A}$ for $q = 2$ queries. $\mathbf{A}$ can choose the inputs x, and is given $G(x)$ for each query, but has no access to neither F nor F'. If no such efficient distinguisher exists, briefly explain why.

a) $G(x) := \text{revert}(F(x))$, where $\text{revert}(\cdot)$ reverts the bit order.

b) $G(x) := F(0^n)$.

c) $G(x) := F(x) \oplus F'(x)$, where $\oplus$ denotes bitwise XOR.

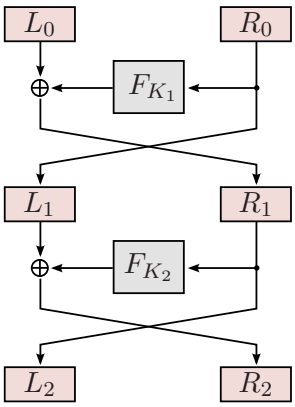
d) $G(x) := F(x) \oplus F(x \oplus 1)$.

Hint: For details about distinguishers and advantages, please see the final page of this problem set.

Question 3 – Two-Round Feistel (4 Points)
Luby and Rackoff's theorem showed that a three-round Feistel network is a secure PRP. Let's see what goes wrong if we use only a **two-round** Feistel network $E : (\{0,1\}^k)^2 \times \{0,1\}^{2n} \to \{0,1\}^{2n}$ that uses a secure PRF $F : \{0,1\}^k \times \{0,1\}^n \to \{0,1\}^n$ with two independent secret keys K_1 and K_2. K_1 is used for the first, and K_2 for the second round. We write $2n$-bit plaintexts as (L_0, R_0) and compute the $2n$-bit ciphertexts (L_2, R_2) as follows:

$$L_1 = R_0 \qquad R_1 = F_{K_1}(R_0) \oplus L_0$$
$$L_2 = R_1 \qquad R_2 = F_{K_2}(R_1) \oplus L_1.$$

The encryption process is illustrated below. Describe an efficient attack that can distinguish the two-round Feistel cipher from a random function. Good distinguishers do not need more than two chosen-plaintext queries.

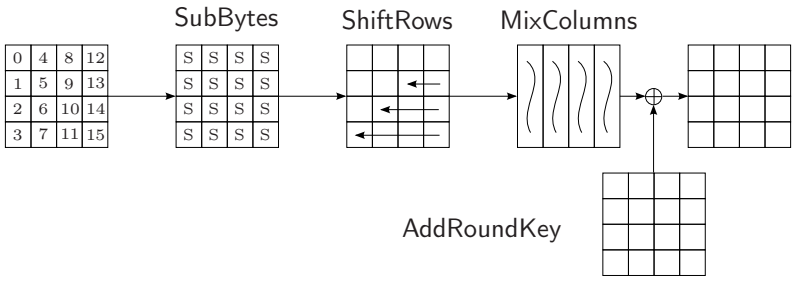


Question 4 – Multiple Encryption (4 Points)
Let $E : \{0,1\}^k \times \{0,1\}^n \to \{0,1\}^n$ be a secure block cipher with a k-bit key. Alice wants to construct a cipher with very high security. Therefore, she defines a cipher $\mathcal{E} : \{0,1\}^{3k} \times \{0,1\}^n \to \{0,1\}^n$ that encrypts a message M by using E three times in a row, under independent secret keys $K_1, K_2, K_3 \in \{0,1\}^k$:

$$C = \mathcal{E}_{K_1,K_2,K_3}(M) = E_{K_3}(E_{K_2}(E_{K_1}(M)))$$

Describe a key-recovery attack on $\mathcal{E}$ that takes significantly less time than testing all 2^{3k} keys. Specify your attack time and memory complexities.

Question 5 – AES (4 Points)
The AES is the standard block cipher nowadays: each of its 10 rounds consists of SubBytes, ShiftRows, MixColumns and AddRoundKey (except for the last round). Below you find an illustration of one round. The input on the left also shows the order of the bytes in the state.



This task shows what would happen if an operation missed. Let $P, P' \in \{0,1\}^{128}$ be two plaintexts that differ only in their byte 0 (here encoded in hexadecimal):

$$P = \texttt{0x0001020304050607 08090a0b0c0d0e0f},$$
$$P' = \texttt{0xff01020304050607 08090a0b0c0d0e0f}.$$

In the following, both plaintexts were encrypted with three different versions of the AES:

1. An AES version where MixColumns was omitted in every round.

2. An AES version where ShiftRows was omitted in every round.

3. A complete AES version.

Below you find three ciphertext pairs, encoded in hexadecimal. For all pairs, C_i is the ciphertext of P and C_i' that of P'. Find out which ciphertext pair (C_i, C_i') was generated by which AES version from above and explain your answer briefly.

i	C_i	C_i'
1	0x231446982181c5ca6476f957632cbd2a	0xbbac089c1cc8ff7e83f147177b488316
2	0xceee58e9437269400fc7e2466b25564c	0xb7ee58e9437269400fc7e2466b25564c
3	0x2b45b04bce63e0c1e750ca0c876248a9	0xb6012b6ece63e0c1e750ca0c876248a9

Question 6 – Block-Cipher Modes (4 Points)
Alice wants to send the two-block (32 characters) message

$$M = (M_1, M_2) = \texttt{Send␣to␣Bob␣100,-␣EUR␣from␣ Alice}$$

with $M_1, M_2 \in \{0,1\}^n$ encoded as 8-bit ASCII string encrypted to her bank. Alice chooses an initial value $IV \in \{0,1\}^n$, encrypts M with AES-128 in some mode and her secret key, and transmits the resulting ciphertext (IV, C_1, C_2) to her bank.
An adversary Eve intercepts Alice's ciphertext. Instead of the original text, Eve wants to replace it with a manipulated ciphertext $C' = (IV', C_1', C_2')$ for the message

$$M' = (M_1', M_2') = \texttt{Send␣to␣Eve␣400,-␣EUR␣from␣Alice}$$

a) Specify a possible ciphertext (IV', C_1', C_2') for M' when the used mode is Counter mode.

b) Specify a possible ciphertext (IV', C_1', C_2') for M' when the used mode is CBC.

Hint: Note that Eve can freely choose a new initial value IV'.

Question 7 – Simple MACs (1+2+2 Points)
For each of the following MACs, briefly explain why they are secure or describe an efficient forgery attack, i.e., how to efficiently forge the authentication code for a message. Prior, you can ask for the authentication of up to three chosen messages.

a) $MAC_K(M) := \bigoplus_{i=1}^{m} E_K(M_i)$.

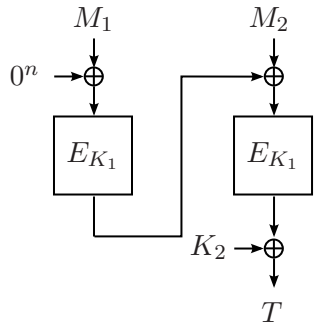
b) $MAC_K(M) := \bigoplus_{i=1}^{m} E_K(M_i \oplus \langle i \rangle)$, where $\langle i \rangle$ denotes the n-bit representation of i.

c) $MAC_K(M) := \bigoplus_{i=1}^{m} E_{K \oplus \langle i \rangle}(M_i)$.

Question 8 – CBC-MAC' (4 Points)
Assume that $E : \{0,1\}^k \times \{0,1\}^n \to \{0,1\}^n$ is a secure block cipher and $K_1 \in \{0,1\}^k$ a secret key. Let $K_2 \in \{0,1\}^n$ be a second independent key. We consider the following variant of CBC-MAC, called CBC-MAC'. Given an m-block message $M = (M_1, \ldots, M_m)$, CBC-MAC' computes an authentication tag as follows:

$$\begin{aligned} C_0 &= 0^n, \\ C_i &= E_{K_1}(C_{i-1} \oplus M_i), \qquad \text{for } 1 \le i \le m, \\ \text{CBC-MAC}'_{K_1,K_2}(M) &:= C_m \oplus K_2, \end{aligned}$$

Describe an efficient forgery attack that can predict the tag for a message with CBC-MAC'. You may ask for the tags of at most three (other) messages before. Note that you can vary the lengths of your chosen messages. Below, you find an illustration for a two-block message.



Recap on Distinguishers

A distinguisher $\mathbf{A}$ is an efficient algorithm that has access to a fixed number of oracles that it can ask queries to and obtain responses from. W.l.o.g., $\mathbf{A}$ never asks duplicate queries or queries that are exactly equal to an earlier response from one of the oracles. The task of $\mathbf{A}$ is to distinguish between two worlds: $\mathcal{D}_{\text{real}}$ and $\mathcal{D}_{\text{ideal}}$. Each world represents a distribution of outputs from the oracles. At the beginning of the experiment, one of the worlds is chosen secretly and uniformly at random. After $\mathbf{A}$ has finished its interactions with the oracles, $\mathbf{A}$ has to output a bit b that represents a guess of the world that $\mathbf{A}$ interacted with. The advantage of $\mathbf{A}$ is given by

$$\underset{\mathbf{A}}{\Delta}(\mathcal{D}_{\text{real}}, \mathcal{D}_{\text{ideal}}) \stackrel{\text{def}}{=} \left|\Pr\left[\mathbf{A}^{\mathcal{D}_{\text{real}}} \Rightarrow 1\right] - \Pr\left[\mathbf{A}^{\mathcal{D}_{\text{ideal}}} \Rightarrow 1\right]\right|,$$

where $\mathbf{A}^x$ means that $\mathbf{A}$ interacted with the world x, and $\mathbf{A} \Rightarrow 1$ means that $\mathbf{A}$'s output was $b = 1$, i.e., $\mathbf{A}$ chose the real world. The probabilities are taken over all random choices of the distributions.

Example: To evaluate the PRF-security of some keyed function G, $\mathbf{A}$ interacts with a challenger $\mathcal{C}$. $\mathcal{C}$ chooses some random secret key K and a secret bit $b \twoheadleftarrow \{0,1\}$. Then, $\mathbf{A}$ can ask queries x_1, x_2, ... If $b = 1$, then $\mathcal{C}$ answers the queries by $\mathbf{A}$ using the real construction and $\mathbf{A}$ obtains real answers: $y_1 = G_K(x_1)$, $y_2 = G_K(x_2)$, etc. If $b = 0$, then $\mathcal{C}$ answers the queries by $\mathbf{A}$ using the ideal construction \$ that only samples the answers at random: $y_1 \twoheadleftarrow \{0,1\}^n$,

$y_2 \twoheadleftarrow \{0,1\}^n$, etc. **A** wins if it guesses the world correctly just from observing the answers. It can guess, but in the long term, it will guess exactly as many times correctly as it guesses wrongly and has advantage 0. So, **A** can have a significant advantage only if it can find something in the real answers that allows it to distinguish both worlds. Otherwise, G is a called secure in the PRF model.