Solutions to *Applied Cryptography* (Version 0.6, Jan. 2023) by Boneh & Shoup

Edgard Lima <edgard.lima@gmail.com>

https://www.linkedin.com/in/edgardlima/

https://edgardlima.com

https://github.com/Zer0Leak/AppliedCryptographyBonehBookSolutions

2025

Contents

2 Encyption **1**

3 Stream ciphers **5**

4 Block ciphers **11**

Chapter 2

Encyption

2.1 (multiplicative one-time pad). We may also define a "multiplication mod p" variation of the one-time pad. This is a cipher E = (E, D), defined over (K, M, C), where K := M := C := 1, . . . , p-1, where p is a prime. Encryption and decryption are defined as follows:

$$E(k, m) := k \cdot m \bmod p \qquad D(k, c) := k^{-1} \cdot c \bmod p$$

Here, k^{-1} denotes the multiplicative inverse of k modulo p. Verify the correctness property for this cipher and prove that it is perfectly secure.

Answer: Auxiliary

Notice that $k^{-1} \cdot k \bmod p = 1$ (multiplicative inverse of k modulo p)

Given p is prime, k^{-1} is unique. Let's prove.
Suppose there is x and y, such that $k \cdot x \bmod p = 1 = k \cdot y \bmod p$. Also, make sure x and y are reduced by mod$p$. *i.e.* $0 < x, y < p$
Since p is prime and $0 < k < p$, we can divide both sides by k.
$x \bmod p = y \bmod p$
$x = y$

Answer: Correctess

Notice that $k^{-1} \cdot k \bmod p = 1$ (multiplicative inverse of k modulo p)

$c = Enc(k, m) := k \cdot m \bmod p$
$Dec(k, Enc(k, m)) = k^{-1} \cdot c \bmod p$
$Dec(k, Enc(k, m)) = k^{-1} \cdot (k \cdot m \bmod p) \bmod p$
$Dec(k, Enc(k, m)) = k^{-1} \cdot k \cdot m \bmod p \bmod p$
$Dec(k, Enc(k, m)) = k^{-1} \cdot k \cdot m \bmod p$

$Dec(k, Enc(k, m)) = 1 \cdot m \bmod p$
Since $1 \leqslant m \leqslant p - 1$, then
$Dec(k, Enc(k, m)) = m$

Answer: Perfectly Secure

$m^{-1} \cdot m \bmod p = 1$ (multiplicative inverse of m modulo p)

$k \cdot m \bmod p = c$
$k \cdot m \cdot m^{-1} \bmod p = c \cdot m^{-1} \bmod p$
$k \bmod p = c \cdot m^{-1} \bmod p$
Since m^{-1} is unique, for every $c \in \mathcal{C}$, and for all message $m \in \mathcal{M}$
$N_c = |\{k \in \mathcal{K} : E(k, m) = c\}| = 1$
This is perfectly secure according to **Theorem 2.1 (ii)**

2.2 (A good substitution cipher). Consider a variant of the substitution cipher $\mathcal{E} = (E, D)$ defined in Example 2.3 where every symbol of the message is encrypted using an independent permutation. That is, let $\mathcal{M} = \mathcal{C} = \Sigma^L$ for some a finite alphabet of symbols Σ and some L. Let the key space be $\mathcal{K} = S^L$ where S is the set of all permutations on Σ. The encryption algorithm E(k, m) is defined as:

$$E(k, m) := k[0](m[0]), k[1](m[1]), ..., k[L-1](m[L-1])$$

Show that $\mathcal{E}$ is perfectly secure.

Answer

The encryption decryption of each symbol is independent. At each index there is an independent **substitution cipher**.

Therefore, we can reduce to prove that $\mathcal{M} = \mathcal{C} = \Sigma$, and $\mathcal{K} = S$, *i.e.*, m and c has length 1, and $|\mathcal{K}| = |\Sigma|!$ is perfectly secure.

$P_r[Enc(k, m) = c] = P_r[k(m) = c] = 1/|\Sigma|$ for all $m \in \mathcal{M}$ and all $c \in \mathcal{C}$ and an uniform distribution of $\mathcal{K}$

Therefore it is perfectly secure directly from the **Definition 2.1 (perfect security)**

2.3 (A broken one-time pad). Consider a variant of the one time pad with message space $\{0,1\}^L$ where the key space $\mathcal{K}$ is restricted to all L-bit strings with an even number of 1's. Give an efficient adversary whose semantic security advantage is 1

Answer

The adversary, $\mathcal{A}$, choose $m_0 := 0^L$, and $m_1 := 0^{L-1}1$

If the cipher text c has an even parity it outputs $\hat{b} = 0$ (because it was exactly the parity of the key)

Otherwise, cipher text c has an odd parity, it outputs $\hat{b} = 1$. Because the number of 1's will be the number of 1's in the key, that is even, minus one if the key has a 1 at index $L-1$, or plus one, if the key has a 0 at index $L-1$.

Chapter 3

Stream ciphers

3.1 (Semantic security for random messages). One can define a notion of semantic security for random messages. Here, one modifies Attack Game 2.1 so that instead of the adversary choosing the messages m_0 , m_1 , the challenger generates m_0 , m_1 at random from the message space. Otherwise, the definition of advantage and security remains unchanged.

(a) Suppose $\mathcal{E} = (\mathsf{E}, \mathsf{D})$ is defined over $(\mathcal{K}, \mathcal{M}, \mathcal{C})$, where $\mathcal{M} = \{0,1\}^L$. Assuming that $\mathcal{E}$ is semantically secure for random messages, show how construct a new cipher $\mathcal{E}'$ that is secure in the ordinary sense. You new cipher should be defined over $(\mathcal{K}', \mathcal{M}', \mathcal{C}')$, where $\mathcal{K}' = \mathcal{K}$ and $\mathcal{M}' = \mathcal{M}$.

(b) Give an example of a cipher that is semantically secure for random messages but that is not semantically secure in the ordinary sense.

Answer: (a)

$E(k, m)$ is secure for a random m.
Make $k = k'$

Definition of E':
- Generate r from random $\{0,1\}^L$
- $(r, E'(k', m')) := E(k, m' \oplus r) := E(k, m) = c'$

Notice that $m = m \oplus r$ is random, so E is secure for it.
Notice that, if the adversary knows r and c', it doesn't help to get m, because it is encrypted with k.

Definition of D':
- $m' := (r, D'(k', c')) := D(k, c') \oplus r := D(k, E(k, m)) \oplus r := m \oplus r$

Answer: (b)

Consider E such that $c \in \{0,1\}^{L+1}$. The function E extends the bit-string c by appending a single 0 bit at the end of c in the message is exactly 0^L. Otherwise it appends 1.
The chance of the adversary in this game is:
$1.0 \times 2^{L-1} + \frac{1}{2^L-1} \times \frac{2^L-1}{2^L} + negl(L) = 2^{-(L-1)} + negl(L)$
Notice that 2^{-L} is negligible so $2 \times 2^{L-1}$ is too, and therefore is $2^{-(L-1)} + negl(L)$.

Now the only thing the (ordinary sense) adversary needs to do is choose $m_0 := 0^L$ and $m_1 \neq m_0$. And now the chance becomes 1.0 because if c end with 0, m_0 was chosen, otherwise was m_1.

3.2 (Encryption chain. Let $\mathcal{E} = (E, D)$ be a cipher defined over $(\mathcal{K}, \mathcal{M}, \mathcal{C})$ where $\mathcal{K} = \mathcal{M}$. Let $\mathcal{E}' = (E', D')$ be a cipher where encryption is defined as . Show that if $\mathcal{E}$ is semantically secure then so is $\mathcal{E}'$.

Answer

3.6 (Another malleability example).

Let us give another example illustrating the malleability of stream ciphers. Suppose you are told that the stream cipher encryption of the message "attack at dawn" is `6c73d5240a948c86981bc294814d` (the plaintext letters are encoded as 8-bit ASCII and the given ciphertext is written in hex). What would be the stream cipher encryption of the message "attack at dusk" under the same key?

Answer

In a stream cipher the key string of the same length is the same for the same seed.
So, if m_0 and m_1 has the same length:
$c_0 = s \oplus m_0$ and $c_1 = s \oplus m_1$, thus
$c_0 \oplus m_0 = s = c_1 \oplus m_1$
$c_1 = c_0 \oplus m_0 \oplus m_1$
m_0 and m_1 are equal except for the last 3 letters, so we only need to compute these XORs.
$c_1[10] = c_0[10] \oplus m_0[10] \oplus m_1[10] = 94 \oplus$ 'a' $\oplus$ 'u' $= 94 \oplus 61 \oplus 75$
Do the same for the last 2 letters.
c_1 = `6c73d5240a948c86981bc2808548`

3.20 (Nested PRG construction). Let $G_0 : \mathcal{S} \to \mathcal{R}_1$ and $G_1 : \mathcal{R}_1 \to \mathcal{R}_2$ be two secure PRGs. Show that $G(s) := G_1(G_0(s))$ mapping $\mathcal{S}$ to $\mathcal{R}_2$ is a secure PRG.

Answer

Let's prove by contraposition.
To simplify, let's define:
$H_0 = G_1(G_0(s))$, where $s \xleftarrow{r} \mathcal{S}$
$H_1 = G_1(r_1)$, where $r_1 \xleftarrow{r} \mathcal{R}_1$
$H_2 = r_2$, where $r_2 \xleftarrow{r} \mathcal{R}_2$
If $\mathcal{B}$ breaks $G_1(G_0(r))$, then:

$|\Pr[\mathcal{B}(H_0) = 1] - \Pr[\mathcal{B}(H_2) = 1]| = \epsilon$, where ϵ is non-negligible.
$|\Pr[\mathcal{B}(H_0) = 1] - \Pr[\mathcal{B}(H_1) = 1]| + |\Pr[\mathcal{B}(H_1) = 1] - \Pr[\mathcal{B}(H_2) = 1]| \geq \epsilon$
So, at least one of the two terms of the sum is $\geq \epsilon/2$.

If $|\Pr[\mathcal{B}(H_0) = 1] - \Pr[\mathcal{B}(H_1) = 1]| \geq \epsilon/2$, then adversay $\mathcal{B}$ can break G_0.
It is easy to see. The challenger sends $G_0(s)$ or r_1 to $\mathcal{B}$. $\mathcal{B}$ computes G_1 of this input and just distinguishes between the two cases.

If $|\Pr[\mathcal{B}(H_1) = 1] - \Pr[\mathcal{B}(H_2) = 1]| \geq \epsilon/2$, then G_1, by definition is not secure.
So, if $G_1(G_0(s))$ is not secure, then at least one of G_0 or G_1 is not secure.

3.22 (Bad seeds). Show that a secure PRG $G : \{0,1\}^n \to R$ can become insecure if the seed is not uniformly random in S.

(a) Consider the PRG $G' : \{0,1\}^{n+1} \to R \times \{0,1\}$ defined as $G'(s_0 \| s_1) = (G(s_0), s_1)$. Show that G' is a secure PRG assuming G is secure.

(b) Show that G' becomes insecure if its random seed $s_0 \| s_1$ is chosen so that its last bit is always 0.

(c) Construct a secure PRG $G'' : \{0,1\}^{n+1} \to R \times \{0,1\}$ that becomes insecure if its seed s is chosen so that the parity of the bits in s is always 0.

Answer: (a)

We are assuming s_0 and s_1 are uniformly random and independent
Let's prove by contraposition. *I.e.* if G' is not secure, then G is not secure.
There is an adversary $\mathcal{A}$ that breaks G'. let's wrap it to build an adversary $\mathcal{B}$ that breaks G.
The challenger send either $G(s_0)$ or s_0 to $\mathcal{B}$. $\mathcal{B}$ appends a random bit s_1 to the input and send it to $\mathcal{A}$.
$\mathcal{B}$ just outputs whatever $\mathcal{A}$ outputs, and has the same non-negligible advantage to distinguish the two cases.

Answer: (b)

Just build an adversary $\mathcal{A}$ that checks the last bit of the input. If the last bit is 0, it outputs 0, otherwise 1.
The experiment with G' has probability 0 to output 1, while the experiment with a random string has probability 1/2 to output 1.
$\Pr[\mathcal{A}(G'(s_0 \| s_1)) = 1] - \Pr[\mathcal{A}(r) = 1] = 1/2$, which is not negligible.

Answer: (c)

$G''(s) = (G(s), \text{parity}(s))$
So, it is broken essentially with the same adversary of question (b).

Chapter 4

Block ciphers

Problem 7(a) https://crypto.stanford.edu/~dabo/courses/cs255_winter25/hw_and_proj/hw2.pdf

Answer

$2D((k_1, k_2), c) = D(k_2, D(k_1, c))$

Because of **meet in the middle attack**, this construction is not secure.
Given a known pair (m, c), the adversary can build a hash table of 2^{128} entries of $E(k_1', m)$ for all possible k_1' values. $E(k_1', c)$ is the hash table key, while k_1' is the value. It takes time $O(|K|)$ to build.

Then, for each possible k_2', it can compute $D(k_2', c)$ and check if it matches any entry in the table. If it is present in table, then this pair (k_1', k_2') is a candidate key pair. It is a false positive with probability $1/2^{128}$. And we can check it by just choosing another random message and checking if it encrypts to the expected ciphertext.

This brings down the security from 2^{256} to $2 \cdot 2^{128} = 2^{129}$.

4.1 (Exercising the definition of a secure PRF). Let F be a secure PRF defined over $(\mathcal{K}, \mathcal{X}, \mathcal{Y})$, where $\mathcal{K} = \mathcal{X} = \mathcal{Y} = \{0,1\}^n$.

(a) Show that $F_1(k, x) = F(k, x)||0$ is not a secure PRF.

(b) Show that $F_2(k, (x, y)) := F(k, x) \oplus F(k, y)$ is insecure.

(c) Prove that $F_3(k, x) := F(k, x) \oplus x$ is a secure PRF.

Answer: (a)

Let's use the attach game 4.2
$PRFadv[\mathcal{A}, \mathcal{F}] := |\Pr[W_0] - \Pr[W_1]|$
The Adversary just need to send one x input block.
It outputs 0 when the last of y is 0, otherwise 1.
In Experiment 0, $\Pr[\hat{b} = 1] = 1$, in Experiment 1, $\Pr[\hat{b} = 1] = 1/2$.
So, $PRFadv[\mathcal{A}, \mathcal{F}] = |1 - 1/2| = 1/2$, which is non-negligible.

Answer: (b)

Let's use the attach game 4.2
$PRFadv[\mathcal{A}, \mathcal{F}] := |\Pr[W_0] - \Pr[W_1]|$
The Adversary just need to send on input blocks (x, y) with any $x = y$.
The challenger gives back c.
If the $c = 0^n$, the adversary outputs 0, otherwise 1.
In Experiment 0, $\Pr[\hat{b} = 1] = 1$, in Experiment 1, $\Pr[\hat{b} = 1] = 1/2^n$.
So, $PRFadv[\mathcal{A}, \mathcal{F}] = |1 - 1/2^n|$, which is non-negligible, ranging from $1/2$ to almost 1 for large n.

Answer: (c)

Let's prove by contraposition. *I.e.* if F_3 is not secure, then F is not secure.
There is an adversary $\mathcal{A}$ that breaks F_3. let's wrap it to build an adversary $\mathcal{B}$ that breaks F.
$\mathcal{B}$ computes $x \xleftarrow{r} \mathcal{R}$ and send x to the challenger, and receives c.
Then, it computes $c \oplus x$ and send the result to $\mathcal{A}$. Notice that x is truly random so, only if c is not truly random, $c \oplus x$ is not truly random.
Finally, $\mathcal{B}$ just outputs whatever $\mathcal{A}$ outputs, and has the same non-negligible advantage to distinguish the two cases.
$\mathcal{A}$ can distinguish between the two cases with non-negligible advantage.
Since $\mathcal{B}$ just forwards the input and output of $\mathcal{A}$, it has the same non-negligible advantage to distinguish the two cases.

4.4 (Truncating PRFs). Let F be a PRF whose range is $\mathcal{Y} = \{0,1\}^n$. For some $\ell < n$ consider the PRF F' with a range $\mathcal{Y}' = \{0,1\}^\ell$ defined as: $F'(k,x) := F(k,x)[0 \ldots \ell - 1]$. That is, we truncate the output of $F(k,x)$ to the first ℓ bits. Show that if F is a secure PRF then so is F'

Answer

Let's prove by contraposition. *I.e.* if F' is not secure, then F is not secure.
There is an adversary $\mathcal{A}$ that breaks F'. let's wrap it to build an adversary $\mathcal{B}$ that breaks F.
The adversary $\mathcal{B}$ send enough inputs to the challenger and receives back the corresponding outputs. When it receives two outputs where the first ℓ bits are different it can stop quering and use this pair to challenge $\mathcal{A}$.
$\mathcal{B}$ truncates the outputs to the first ℓ bits and send them to $\mathcal{A}$.
Finally, $\mathcal{B}$ just outputs whatever $\mathcal{A}$ outputs, and has the same non-negligible advantage to distinguish the two cases.

4.5 (Two-key Triple-DES). Consider the following variant of the $3\mathcal{E}$ construction that uses only two keys: for a block cipher (E, D) with key space $\mathcal{K}$ define $3\mathcal{E}'$ as $E((k_1, k_2), m) := E(k_1, E(k_2, E(k_1, m)))$. Show that this block cipher can be defeated by a meet in the middle attack using $O(|\mathcal{K}|)$ evaluation of E and D and using $O(|\mathcal{K}|)$ encryption queries to the block cipher challenger. Further attacks on this method are discussed in [112, 105].

Answer

First, let's fix a $E_{k1}(m)$ we want to reach. Let's choose 0^{64}.

Now, let's compute a table, for all candidates k_2, and name it k_2'. It will be a hash table, mapping $E(k_2', 0^{64})$ to k_2'. This takes time $O(|\mathcal{K}|)$.

Now, let's find messages of interest for our attack. For all possible k_1, named k_1', we will compute $m_i = D(k_1', 0^{64})$. This takes time $O(|\mathcal{K}|)$. Store this pair (m_i, k_{1i}') in a list. This gives us a list of approximatelly distinguish $|\mathcal{K}|$ messages.

That's Because the probability of $k_1' \neq k_2'$ satisfying $D(k_1', 0^{64}) = D(k_2', 0^{64})$ is $1/2^{64}$. So, the expected number of collisons is $= \binom{K}{2} \cdot \frac{1}{B} = \frac{2^{56}(2^{56}-1)}{2} \cdot \frac{1}{2^{64}} \approx \frac{2^{112}}{2^{65}} = 2^{47}$. So, we expect the we will have $2^{56} - 2^{47} \approx 0.99805 \cdot 2^{56}$ distinguish messages.

Let's ask the challenger to encrypt them. This will give us $c_i := E(k_1, E(k_2, E(k_1, m_i)))$. This also takes time $O(|\mathcal{K}|)$. Add c_i to the pair (m_i, k_i'), making it a tuple (m_i, c_i, k_{1i}'). Remember that k_{1i}' is just a candidate for real k_1 used to generate c_i.

Now, for (m_i, c_i, k_{1i}') tuple, compute $D(k_{1i}', c_i)$, and check if the result is present in the initial hash table (takes $O(1)$ each query). *I.e.* We are checking if there exists k_{2i}' such that $E(k_{2i}', 0^{64}) = D(k_{1i}', c_i)$.

If so, we have found a candidate pair (k_{1i}', k_{2i}'). This is the *meet* point. We add it to a list of candidates.

... continues in next page...

Notice this pair may be a false positive. Actually, the probability (k'_{1i}, k'_{2i}) be false positive is $(2^{56}-1)/2^{64} \approx 2^{-8}$ (-1 is the right one), i.e. the number of entries in the hash table divided by the size of the block space resulting from $D(k'_{1i}, c_i)$. Since we tried 2^{56} tuples, we expect to have $2^{56} \cdot 2^{-8} = 2^{48}$ false positive pairs and one positive.

Finally, we will verify each candidate in our list by encrypting a new random message and checking if the result is correct. This takes time $2^{48} + 1 = O(|\mathcal{K}|)$ in total.

The chance a false positive passes this test is $1/2^{64}$, so the expected number of false positives passing this test is $2^{48}/2^{64} = 2^{-16}$. If more than one candidate passes the test, we can repeat the test with a new random message until only one candidate remains. Notice that repeating test just a second time $2^{48}/2^{128} = 2^{-80}$.

4.9 (Strongly secure block ciphers). In Section 4.1.3 we sketched out the notion of a strongly secure block cipher.

(a) Write out the complete definition of a strongly secure block cipher as a game between a challenger and an adversary.

(b) Consider the following cipher $\mathcal{E}' = (E', D')$ built from a block cipher (E, D) defined over $(\mathcal{K}, \{0,1\}^n)$:

$E'(k, m) := D(k, t \oplus E(k, m))$ and $D'(k, c) := D(k, t \oplus E(k, c))$
where $t \in \{0,1\}^n$ is a fixed constant. For what values of t is this cipher $\mathcal{E}'$ semantically secure? Prove semantic security assuming the underlying block cipher is strongly secure

Answer: (a)

This is exactly same as the definition of secure block cipher, *i.e.* **Definition 4.1**, but now the adversary has access to both encryption and decryption oracles. *I.e.* the adversary can query the challenger with both (m_i) and (c_j), receiving back $E(k, m_i)$ and $D(k, c_j)$ respectively, for any $m_i, c_j \in \{0,1\}^n$ of its choice. Also notice that D is the inverse of E in both Experiments.

Answer: (b)

It is **not** semantically secure for any value of t.

Let's use the attach game 4.1.

The adversary just need to send one input block m theb it gets back $c = f(k, m)$. Now, it sends c. If it gets back m outputs $\hat{b} = 0$, otherwise $\hat{b} = 1$. Notice that in Experiment 0, $k \xleftarrow{r} \mathcal{K}$ and $f \leftarrow E'(k, \cdot)$. While in Experiment 1, $f \xleftarrow{r} Perms[\mathcal{X}]$

In Experiment 0, $\Pr[\hat{b} = 1] = 1$, in Experiment 1, $\Pr[\hat{b} = 1] = 1/2^n$.

Let's prove the Experiment 0 case:

$E'_k(c) = E'_k \circ E'_k(m) = E'_k \circ D_k(t \oplus E_k(m)) = D_k(t \oplus E_k \circ D_k(t \oplus E_k(m)))$
$= D_k(t \oplus t \oplus E_k(m)) = D_k(E_k(m)) = m$