INFO-F-405 : Security
Encryption and ciphers

Some exercises were inspired by exercises from the Security course given by Dan Boneh.

Exercise 1

In order to save space or bandwidth we often use compression algorithms. Suppose you want to use data compression with encryption. How should these two operations be combined? Why?

Answer of exercise 1

Compress then encrypt. An encrypted message looks like random data and the compression algorithm will not be able to reduce the size.

Exercise 2

Imagine that you have to use a relatively unreliable network i.e., packets often arrive with errors (bit flips) to their destination. Suppose you want to use an error correction code with encryption. How should these two operations be combined? Why?

Answer of exercise 2

Encrypt then apply error correction code. If the ciphertext is corrupted, the decryption algorithm will not work and there is no way to retrieve any meaningful information.

Exercise 3

Suppose that the probability of wining the lottery is $\frac{1}{10^6}$. What is more likely guessing an AES-128 key on the first try or winning the lottery 5 times in a row? What about winning the lottery 6 times in a row?

Answer of exercise 3

The probability to guess an AES-128 key is

$$P_{AES128} = \frac{1}{2^{128}} = 2^{-128}$$

The probability to win five times in a row to the lottery is

$$P_{lottery_5} = (10^{-6})^5 = 10^{-30}$$

We know $2^{10} > 10^3$. Hence $2^{-10} < 10^{-3}$.

$$\begin{aligned} P_{AES128} &\overset{?}{\geq} P_{lottery_5} \\ \Leftrightarrow 2^{-8} \times (2^{-10})^{12} &\overset{?}{\geq} 10^{-30} \\ \Leftrightarrow 2^{-8} \times (10^{-3})^{12} &\overset{?}{\geq} 10^{-30} \\ \Leftrightarrow 2^{-8} \times 10^{-36} &< 10^{-30} \end{aligned}$$

It is more likely to win the lottery five times in a row than to guess an AES key.

The probability to win six times in a row to the lottery is

$$P_{lottery_6} = (10^{-6})^6 = 10^{-36}$$

And

$$2^{-8} \times 10^{-36} < 10^{-36}$$

So it is still more likely to win 6 times in a row than to guess an AES key on the first try.

Exercise 4

A bank defines an encryption algorithm to encrypt their account numbers. An account number is a 12-digit number, i.e., an element of $\mathbb{Z}_{10^{12}}$. Every time an account number is encrypted, a fresh key is chosen randomly and uniformly from the same set $\mathbb{Z}_{10^{12}}$. The encryption goes as follows:

$$E_k(m) = m + k \pmod{10^{12}} \quad D_k(c) = c - k \pmod{10^{12}}$$

Does this cipher have perfect secrecy?

Answer of exercise 4

Yes, as only one key maps a plaintext m to a ciphertext c, i.e., $\forall m, c$ there exists exactly one value k s.t. $E_k(m) = c$. So, the probability $\Pr[E_k(m) = c] = \Pr[k \text{ was drawn as key}] = 10^{-12}$ is independent of m.

Exercise 5

Suppose that (E, D) is an IND-CPA secure encryption scheme, i.e., there is no known way of winning the IND-CPA game other than with a probability negligibly close to $\frac{1}{2}$ or with over-astronomical resources. Let the key and plaintext spaces be $\{0,1\}^n$ with $n \geq 128$. Point out the IND-CPA secure schemes in the following list:

- $E'_k(m) = E_k(m)||E_k(m)$
  (+a side-question: will the first and last n bits of $E'_k(m)$ always be equal?)
- $E'_k(m) = E_k(m)||\text{first bit}(m)$
- $E'_k(m) = (\text{last bit}(m) \oplus \text{first bit}(m))||E_k(m)$

- $E'_k(m) = E_k(m)||1$
- $E'_k(m) = k||E_k(m)$
- $E'_k(m) = E_k(m)||(k \oplus 1^n)$
- $E'_{k1,k2}(m) = E_{k1}(m)||E_{k2}(m)$
- $E'_k(m) = E_{0^n}(m)$
- $E'_k(m) = E_k(m)||(m \oplus k)$

For each scheme that is *not* IND-CPA secure, please specify how an adversary can win the IND-CPA game, i.e., what the adversary chooses as plaintexts m_0 and m_1, what are the queries to be made before or after this choice, how the adversary distinguishes between the ciphertexts of m_0 and m_1.

Answer of exercise 5

+ $E'_k(m) = E_k(m)||E_k(m)$: The two ciphertexts do not help the adversary, so E' is IND-CPA secure. (The answer to the side-question is no: If (E, D) is IND-CPA secure, the adversary cannot even tell whether the same plaintext was encrypted twice. This means that the ciphertext space can be larger than the plaintext space, since one plaintext can be mapped to several ciphertexts.)

- $E'_k(m) = E_k(m)||\text{first bit}(m)$: The adversary chooses m_0 and m_1 such that they differ in the first bit. It can recognize which plaintext was encrypted thanks to $\text{first bit}(m)$ exposed in the ciphertext. No need for queries during the game.

- $E'_k(m) = (\text{last bit}(m) \oplus \text{first bit}(m))||E_k(m)$: Same reasoning, except that the adversary chooses m_0 and m_1 such that they differ in either the first or the last bit (but not both).

+ $E'_k(m) = E_k(m)||1$: The extra bit 1 present in all ciphertexts do not help the adversary, so E' is IND-CPA secure.

- $E'_k(m) = k||E_k(m)$: The key is revealed in the ciphertext, removing any security. To win the game, it suffices the adversary to use the revealed key to decrypt $E_k(m_0 \text{ or } m_1)$ and find out which one it was.

- $E'_k(m) = E_k(m)||(k \oplus 1^n)$: Same reasoning, the key is also revealed since the mask 1^n is known.

+ $E'_{k1,k2}(m) = E_{k1}(m)||E_{k2}(m)$: The notation here suggests that the new encryption scheme has a $2n$-bit key, but each n-bit half is used independently. The adversary has twice more chance of winning the IND-CPA on E' than on E, since distinguishing either $E_{k1}(m_0 \text{ or } m_1)$ or $E_{k2}(m_0 \text{ or } m_1)$ is sufficient. Twice a neglible probability is still a neglible probability, hence E' is IND-CPA secure, although 1-bit less secure than E.

- $E'_k(m) = E_{0^n}(m)$: The scheme E' does not make use of its secret key. Anything encrypted with E_{0^n} can therefore be decrypted by the adversary with D_{0^n}, and it can immediately distinguish $E_{0^n}(m_0 \text{ or } m_1)$ to win the game.

- $E'_k(m) = E_k(m)||(m \oplus k)$: As the plaintexts are chosen by the adversary, $m \oplus k$ reveals the key.

Exercise 6

The Mantin-Shamir attack on RC4 shows that the second byte of keystream has a probability of about $\frac{2}{256}$ of taking value 0 and a probability slightly less than $\frac{1}{256}$ of taking each other value. What is the probability of winning the IND-CPA game when exploiting this property? Give an upper bound on the security strength s of RC4.

Answer of exercise 6

Since the second byte of keystream is biased towards value 0, the second byte of ciphertext has slightly more chance of being equal to the second byte of the plaintext than to another value.

We therefore assume the following strategy for the adversary. The adversary chooses the plaintext m_0 (resp. m_1) such that the second byte has value a (resp. b), with $a \neq b$. When getting the second byte c of the encryption of m_0 or m_1, the adversary says that the plaintext is m_0 (resp. m_1) when $c = a$ (resp. $c = b$), and guesses randomly when $c \notin \{a, b\}$.

$$\begin{aligned}\Pr[\text{win}] = \Pr[\text{challenger encrypts } m_0](& \Pr[c = a \mid \text{challenger encrypts } m_0] \\ & + \frac{1}{2}\Pr[c \notin \{a,b\} \mid \text{challenger encrypts } m_0]) \\ + \Pr[\text{challenger encrypts } m_1](& \Pr[c = b \mid \text{challenger encrypts } m_1] \\ & + \frac{1}{2}\Pr[c \notin \{a,b\} \mid \text{challenger encrypts } m_1]).\end{aligned}$$

The challenger will choose m_0 or m_1 each with probability $\frac{1}{2}$. Therefore, by symmetry, we have

$$\begin{aligned}\Pr[\text{win}] =& \Pr[c = a \mid \text{challenger encrypts } m_0] + \frac{1}{2}\Pr[c \notin \{a,b\} \mid \text{challenger encrypts } m_0] \\ \approx& \frac{2}{256} + \frac{1}{2} \times \frac{253}{256} \\ =& \frac{257}{512} = \frac{1}{2} + \frac{1}{512}.\end{aligned}$$

The advantage is therefore $\epsilon = \frac{1}{512}$ and, considering the negligible amount of computations needed to mount this attack, we set $\log_2(t + d) = 0$ and deduce that RC4 has at most 9 bits of IND-CPA security.

Exercise 7

Suppose that Alice would like to send a secret message m that they create together to Charles and Bob. Alice wants to make sure that Bob and Charles decrypt this message together by collaborating, but each one of them should not be able to decrypt the message alone. Suggest several ways of sharing the secret information among people, for each of them list their pros and cons by answering following questions. You can suppose that Alice has a special trusted and secure channel that she can use in order to communicate a key to Bob and to Charles.

- What should Alice do? How should she share a secret key k with Charles and Bob? How would the encryption/decryption would look like?

- Suppose that Alice already shared k with Bob and Charles using a scheme that you've suggested. Now she wants to send other messages to Bob, Charles and David by using the same secret key k in such way that at least two of them should collaborate during the decryption. What kind of information should she give to David? How this scheme could be extended in order to include Eve?

Answer of exercise 7

Discuss and analyze 3 cases for the Alice's key K:

- $K = k_1||k_2$
- $K = (k_1, k_2)$ (encrypt twice)
- $K = k_1 \oplus k_2$

To share between 3 people: encrypt with K, sample 3 random numbers r_1, r_2 and r_3.
Bob gets $K \oplus r_1$ and $K \oplus r_2$. Charles gets r_1 and $K \oplus r_3$ David gets r_2 and r_3