

Solutions to homework 1

Problem 2.2

If encryption is secure then by Definition 2.1 this condition implies that $Pr[M = m] = Pr[M = m']$ for all m, m' in the message space, which is obviously not true if M is sampled from *any* distribution over the message space.

Problem 2.3

After throwing out the key $k = 0^l$ the OTP encryption is no longer perfectly secure because then the key-space has one fewer element than the message space. A good example is that for any $c \in \{0, 1\}^l$ the message $m = c$ is impossible, because $0^l \notin \mathcal{K}$. But any other message is possible, which violates for example the condition in lemma 2.3, which says that encryption is perfectly secure only if for all c, m_0, m_1 the probability that m_0 encrypts to c (over keys) is the same as the probability that m_1 encrypts to c (over keys).

Problem 2.4(b)

Since the substitution cipher has key-space K of size $26!$, it can provide perfect secrecy only if the message space M has at most $26!$ elements too. Let M be the set of 26-letter strings with no repetitions. Then $|M| = 26!$, so it *can* be perfectly secure, but *is* it secure? (Note that $|K| \geq |M|$ is a sufficient, but not necessary condition for perfect secrecy. For example if different keys act like one and the same key then the fact that there's lots of different keys might not be that useful...) As one student pointed out, the substitution cipher on this message space can be thought of as a one-time pad encryption: The message space M is a set of permutations on $\{1, \dots, 26\}$, and so is the key space K . To encrypt $Enc_k(m)$ takes permutations m and k and outputs c which is their composition, i.e. $c = k \circ m$. It's like OTP because it satisfies Shannon's theorem, i.e. (1) every key in K has the same probability, and (2) for every permutation m and c on the set $\{1, \dots, 26\}$, there is a unique k s.t. $c = k \circ m$, namely $k = c \circ m^{-1}$. Since m is a permutation, it has a unique inverse.

Two Lemmas on Negligible Functions

To show that $e_3(n) = e_1(n) + e_2(n)$ is negligible if e_1, e_2 are negligible, take any polynomial $p(n)$. Since $2p(n)$ is a polynomial, by negligibility of e_1, e_2 there exists n_0 s.t. for all $n \geq n_0$ both $e_1(n) < 1/(2p(n))$ and $e_2(n) < 1/(2p(n))$. Therefore for all $n \geq n_0$ we have $e_3(n) < 1/(2p(n)) + 1/(2p(n)) = 1/p(n)$, as needed.

To show that $e'(n) = p'(n)e(n)$ is negligible for any polynomial $p(n)$ if e is negligible, take any polynomial $p(n)$. Since $p'(n) * p(n)$ is a polynomial, by negligibility of e there exists n_0 s.t. for all $n \geq n_0$ we have $e(n) < 1/(p'(n) * p(n))$. Therefore for all $n \geq n_0$ we have $e'(n) < 1/p(n)$, as needed.

Problem 2.12

This is an interesting twist on encryption. Here are two schemes which are perfectly secure, $|\mathcal{K}| = 2^{-t}|\mathcal{M}|$, but decryption is correct only with probability 2^{-t} :

In both schemes $\mathcal{M} = \{0,1\}^n$ and $\mathcal{C} = \mathcal{K} = \{0,1\}^{n-t}$. In both schemes encryption $Enc_k(m)$ outputs $c = k \oplus [m]_1^{n-t}$, i.e. k xor-ed with the substring of m which is m with the last t bits chopped off. The first scheme decrypts c as $c \oplus k$ concatenated with O^{n-t} , and the second scheme decrypts c as $c \oplus k$ concatenated with a *random* string of length $n - t$. Both schemes have that for all m , $Pr[Dec_k(Enc_k(m)) = m] = 2^{-t}$, and both schemes are perfectly secure.

We can also show that one cannot go beyond this bound, i.e. that if $|\mathcal{K}| < 2^{-t}|\mathcal{M}|$ then even this “ 2^{-t} -relaxed” encryption scheme cannot meet perfect security: The argument is a generalization of Theorem 2.7.

Problem 2.13

One can show that $|\mathcal{K}| \geq \frac{1}{\epsilon|\mathcal{M}|+1}|\mathcal{M}|$. Otherwise one gets that the distance between $Pr[M = m|C = c]$ and $Pr[M = m]$ is at least ϵ .

But can you show a cipher that meets this bound, perhaps at least for ϵ of some convenient form? Try any cipher that uses an $(n - t)$ -bit key on n -bit messages, and reveals t bit of the message, but keeps the other $n - t$ bits perfectly secure, e.g. via OTP. The distance in question becomes 2^{-n} for m which *cannot* be encrypted to a given c and $2^{n-t} - 2^{-n}$ for m which *can* be encrypted to a given c . Value ϵ is therefore $\max(2^{-n}, 2^{n-t} - 2^{-n})$. [...]