# Problem Set 3

Problem 1.

For every PPT Algorithmthere exists a probabilistic polynomial-time algorithmsuch that for all efficiently sampleable distributions  and all polynomial time computable functions and there exists a negligible function that accepts a security parameter*.*





Where is chosen according to the distribution of  and the probability is computed over the message space and the key space , and any random coins used by *,* and *.*

*Problem 2.*

Let  be private-key encryption scheme that fails the eavesdropping indistinguishability experiment :

1. Suppose that adversary  is given input , and outputs a pair of messages  of different lengths.
2. The random key k is generated by running , and a random bit  is chosen. Ciphertext  is computed and given to.
3.  outputs a bit , and if it is equal to ** then  succeeded.

Since the adversary can choose the length of the output freely, the probability that will get encrypted is much more likely if the message space is of size *n.*

To be more precise, if the polynomial q is bigger than n:



Thus, if you were sending messages m1 and m2 suppose  message one was “Attack at Dawn” and the second message was “Attack at Dusk”, the adversary would become knowledgeable of the size of both message spaces, and even the amount of keys in use by discovering the length of message 0. This encryption method should never be used, unless you’re trying to lead your adversaries into a trap cipher. Not to mention they might get a little curious when the plain text message runs out of keys and uses the left over message space.

*Problem 3*

Let  where

Let  be a probabilistic polynomial-time adversary, and define  as



We use  to construct a distinguisher for the pseudorandom generator G, such that  succeeds with probability .

**Distinguisher **

 is given a string  (Assumption that *n* can be determined from .)

1. Run to obtain a pair of messages 
2. Choose a random bit  Set 
3. Give ** to and obtain output . Output 1 if  = , and output 0 otherwise.

