Assume, the adversary has access to the encryption server and therefore, before even looking at the ciphertext the adversary can get the encryptions of the messages of the adversary's choice. Such a scheme is not CPA secure. A cryptosystem is CPA secure if for all probabilistic polynomial time adversaries A,

$$P[b_{guess} = b] <= 0.5 + negl(n)$$

But no deterministic encryption algorithm can be CPA secure. If the algorithm is deterministic, that means, if it fixes the message and the key, then automatically the ciphertext gets fixed. Since the adversary has access to the encryption server, it can determine the message by normal string matching.

We can achieve CPA security through probabilistic encryption. Given the message and key, the ciphertext should look as if it is chosen randomly each time we encrypt the message. We will achieve this in the following way - each time we want to encrypt the message m, we will choose a random nonce r and encryption of r. Perform xor operation of Enc(r) with message m and send as ciphertext the pair (r, Enc(r) $\oplus$ m). Since r is randomly chosen, it is random and xor of the message with encryption of r is also random. Therefore each time we send the ciphertext, it looks random.

We can implement this in any of the modes of operation. For our implementation we have chosen randomised counter mode. It is a counter based block cipher method. Every time a counter initiated value is encrypted and given as input to xor with a message which outputs ciphertext.