Nirbhay Sharma (B19CSE114)

# Assignment- 3 - Cryptography

---

**Ans -1.**

Since the above scheme is single message CPA secure then we can say that

$$|Pr[D^{E(k,.)}(m_1^0) = 1] - Pr[D^{E(k,.)}(m_1^1) = 1]| < \epsilon$$

now for multimessage scheme we have encryption as follows

$$E(k, (m_1^0, m_2^0, \ldots, m_q^0)) = (E(k, m_1^0), E(k, m_2^0), \ldots, E(k, m_q^0)) = (c_1^0, c_2^0, \ldots, c_q^0)$$

similar encryption for the message: $(m_1^1, m_2^1, \ldots, m_q^1)$

Proof- since the given scheme is single message cpa secure then a property holds true for each of the message

$$|Pr[D^{E(k,.)}(m_1^0) = 1] - Pr[D^{E(k,.)}(m_1^1) = 1]| < \epsilon$$

$$|Pr[D^{E(k,.)}(m_2^0) = 1] - Pr[D^{E(k,.)}(m_2^1) = 1]| < \epsilon$$

$$\ldots$$

$$|Pr[D^{E(k,.)}(m_q^0) = 1] - Pr[D^{E(k,.)}(m_q^1) = 1]| < \epsilon$$

summing up all above equations we got

$$\sum_{i=1}^{q} \left| Pr[D^{E(k,.)}(m_i^0) = 1] - Pr[D^{E(k,.)}(m_i^1) = 1] \right| < q\epsilon$$

from triangular inequality we also know that

$$|\sum_{i=1}^{q} a_i| \leq \sum_{i=1}^{q} |a_i|$$

$$|\sum_{i=1}^{q} (Pr[D^{E(k,.)}(m_i^0) = 1] - Pr[D^{E(k,.)}(m_i^1) = 1])| \leq \sum_{i=1}^{q} |(Pr[D^{E(k,.)}(m_i^0) = 1] - Pr[D^{E(k,.)}(m_i^1) = 1])| \leq q\epsilon$$

Now since these cipher texts are calculated independently and then concatenated so we can use the property of independence i.e

$$Pr(A_1 \cup A_2 \cup A_2 \ldots \cup A_n) = Pr(A_1) + Pr(A_2) + \ldots + Pr(A_n)$$

so we can finally write

$$\sum_{i=1}^{q} Pr[D^{E(k,.)}(m_i^0) = 1] = Pr[D^{E(k,.)}((m_1^0, m_2^0, \ldots, m_q^0))]$$

$$\sum_{i=1}^{q} Pr[D^{E(k,.)}(m_i^1) = 1] = Pr[D^{E(k,.)}((m_1^1, m_2^1, \ldots, m_q^1))]$$

and hence

$$|\sum_{i=1}^{q} (Pr[D^{E(k,.)}(m_i^0) = 1] - Pr[D^{E(k,.)}(m_i^1) = 1])| = |Pr[D^{E(k,.)}((m_1^0, m_2^0, \ldots, m_q^0))] - Pr[D^{E(k,.)}((m_1^1, m_2^1, \ldots, m_q^1))]| \leq q\epsilon$$

and since q is given here polynomial so $q\epsilon$ will also be a small number and hence the multimessage scheme is also CPA secure.

**Ans -2.**

$G_k(x_1, x_2) = (F_k(x_1) \oplus x_2 || F_k(x_2))$

if it is PRF then we need to prove that it is indistinguishable from a random function which also gives $2n$ bit random output

so we need to find

$$\left| Pr[D^{G_k(.)}(x_1, x_2) = 1] - Pr[D^{f(.)}(x_1, x_2) = 1] \right|$$

if the above expression is $\leq \epsilon$ then is Prf otherwise not

so construct one distinguisher D such that

- Each time it gives $(x_1, x_2)$ (where $x_1 = x_2$) to the function and it will take first n bits of output xor it with x2 and match it with last n bit
- if it matches, it outputs 1 and if it does not match it outputs as 0

so for $G_k(x_1, x_2)$ D outputs 1 for sure and for random D can output 1 with a probability of $\frac{1}{2^n}$

$P[D^{G_k(.)}(x_1, x_2) = 1] = 1$

$P[D^{f(.)}(x_1, x_2) = 1] = \frac{1}{2^n}$

so $\left| Pr[D^{G_k(.)}(x_1, x_2) = 1] - Pr[D^{f(.)}(x_1, x_2) = 1] \right| = |1 - \frac{1}{2^n}| > \epsilon$

and hence it is not a Prf

Now simiar other Discriminatory can be easily constructed which can give two three inputs based on some pattern, which breaks its scheme like one input can be- querying two inputs $(x_1, x_2) \& (x_1', x_2)$ notice that here right half is always same and hence it can match with the last n bit of the output for the two inputs, in a similar manner other discriminatories are constructed.

### Ans -3.

a) consider a single round of feistel structure

$(L_i, R_i) = (R_{i-1}, L_{i-1} \oplus f_k(R_{i-1}))$

also consider the two identities

1. $\bar{x} \oplus y = \overline{x \oplus y}$
2. $\bar{x} \oplus \bar{y} = x \oplus y$

**To prove:**

we need to show that if we use $(\overline{L_{i-1}}, \overline{R_{i-1}})$ with key as $\bar{k}$ as the input to the feistel one round then we should get $(\overline{L_i}, \overline{R_i})$ i.e.

$(\overline{L_i}, \overline{R_i}) = (\overline{R_{i-1}}, \overline{L_{i-1}} \oplus f_{\bar{k}}(\overline{R_{i-1}}))$

**proof:** first trivial case is $\overline{R_{i-1}} = \overline{L_i}$ since $R_{i-1} = L_i$

now for the second expression we have

$\overline{L_{i-1}} \oplus f_{\bar{k}}(\overline{R_{i-1}})$

to prove this to be equal to $\overline{R_i}$ consider $f_k(R)$ so what $f_k(R)$ does is it first expand $R$ and then xor it with $K$ and output is then send to sboxes where it reduces the length of the output giving us 32 bit of output so effectively it does $R \oplus k$ and hence if we even pass $(\overline{R_i}, \bar{k})$ into it, then from second property ($\bar{x} \oplus \bar{y} = x \oplus y$) its output does not change so we can write as

$f_{\bar{k}}(\overline{R_{i-1}}) = f_k(R_{i-1})$

so

$\rightarrow \overline{L_{i-1}} \oplus f_{\bar{k}}(\overline{R_{i-1}})$

$\rightarrow \overline{L_{i-1}} \oplus f_k(R_{i-1}) = \overline{L_{i-1} \oplus f_k(R_{i-1})} \quad (\bar{x} \oplus y = \overline{x \oplus y})$

$\rightarrow \overline{L_{i-1} \oplus f_k(R_{i-1})} = \overline{R_i}$

so we can see that if we use $(\overline{L_{i-1}}, \overline{R_{i-1}})$ with key $\bar{k}$ we get output as $(\overline{L_i}, \overline{R_i})$ for one fiestel round

now for DES we have 16 that rounds and since the above result is generalized for any $i^{th}$ round of feistel then we can conclude from here that -

$DES_{\bar{k}}(\bar{m}) = \overline{DES_k(m)}$

b. from above theorem we can observe that the serching set for keys is reduced to half because once we computed result for $(k, m)$ we can easily get the result for $(\bar{k}, \bar{m})$ and hence we need to bruteforce only half the key set and hence my iterations are $\frac{2^{56}}{2} = 2^{55}$

**Ans -4.**

consider a single round of feistel structure

$(L_i, R_i) = (R_{i-1}, L_{i-1} \oplus f_k(R_{i-1}))$

so if we want to take it to two rounds then the output is as follows

$(L_1, R_1) = (R_0, L_0 \oplus f_{k_1}(R_0))$

$(L_2, R_2) = (R_1, L_1 \oplus f_{k_2}(R_1)) = (L_0 \oplus f_{k_1}(R_0),\ R_0 \oplus f_{k_2}(L_0 \oplus f_{k_1}(R_0)))$

output of above equation is not leaking any information regarding input $(L_0, R_0)$ like in the case of $1$ round of filestal but considering that adversary has polynomial chances to query the encryption scheme so consider one more input as follows

input: $(L_0', R_0)$ notice that this time only left half of input is changed but right half remain as it is.

consider output from the above input

$(L_2', R_2') = (L_0' \oplus f_{k_1}(R_0),\ R_0 \oplus f_{k_2}(L_0' \oplus f_{k_1}(R_0)))$

now we can observe from the above two outputs that

$L_2' \oplus L_2 = (L_0' \oplus f_{k_1}(R_0)) \oplus (L_0 \oplus f_{k_1}(R_0)) = L_0' \oplus L_0$

now if any one of $L_0$ and $L_0'$ is known we can easily get another.

once the above result is proved we can construct a discriminatory as follows

- it will give two inputs $(L_0, R_0)$ and $(L_0', R_0)$ and get output of those and xor the first n bits of both the outputs i.e. $L_2' \oplus L_2$, if it equals to $(L_0' \oplus L_0)$ then it outputs 1 else it outputs 0

so

Notations:

- feistel is represented as $F$
- discriminator has access to oracle $F$, $D^{F()}$

$Pr[D^{F()} = 1] = 1$ (distinguisher can detect whether it is interacting with $F$ with prob 1 due to its construction and defintion of 2 round feistel)

$Pr[D^{R()} = 1] = \frac{1}{2^n}$ (since in case of random for which the two xor's need to be equal, it is true with a probability of $\frac{1}{2^n}$ in case of random)

so $|Pr[D^{F()} = 1] - Pr[D^{R()} = 1]| = |1 - \frac{1}{2^n}| > \epsilon$

so in this way 2 round feistel structure is not a secure PRP

**Ans -5.**

key generation algorithm:

consider a binary tree where each node represents a disk, so the algorithm for key generations is inspired from GGM construction

we fix some key $k$ and make it as root node and then to generate key at each level, like in GGM take 0 in left part and 1 in right part, so suppose dvd is at path LRLRRLRR (L-left,R-right) so we can decode it as 01011011 and similarly unique numbers of same length will be generated at each level and then we can try to expand the length to N bits, if length is small, and try to reduce the length if length is larger and pass it to PRG which generate a random key which is the final key at each node which will also be unique.

the encryption scheme is like this:

suppose we need to go from one root node to some other node, we can go to it by following a unique path and we also want to decrypt the dvd so we can simply use DES here, so we can use DES rounds according to the height of the binary tree, precisely the rounds in DES at a particular level is the height of that level in binary tree and again the decryption is simple using the same hardware by providing the input in reverse manner and reversing the keys, in this way we can encrypt the dvd at any level and also decrypt it from that level itself by reversing the keys.

the advantage of DES here is that it is secure and also it is very easy to decrypt using it at a particular node. and it will work for many dvd's since at each level we only doing H rounds of DES where H is the height of the tree at that level.