0 CPA-Security

Describe the game for CPA-Security.

$Priv_{A,\Pi}^{CPA}(n)$

1. Generate a *key* as $k \xleftarrow{\$} \{0,1\}^n$
2. **Training Phase**
   (a) $\mathsf{Enc}(\mathrm{k}, m_i)$
   (b) **return** c_i
3. **Challenge Phase**
   (a) $\mathsf{Enc}(\mathrm{k}, m_0)$ and $\mathsf{Enc}(\mathrm{k}, m_1)$
   (b) **return** c_0, c_1

In the Game for CPA Security, First a random key is generated as k.

Training Phase: The adversary A sends messages (m_i) to the Oracle, (Drawn above), which encrypts these messages and sends back the ciphertext (c_i) to the adversary. The adversary now has a mapping for the messages to their cipher texts.

Challenge Phase: The adversary A now sends a pair of messages m_0, m_1 to the Oracle. The Oracle generates a bit $b \xleftarrow{\$} \{0,1\}$ and chooses randomly one of the received messages to encrypt as $c* \leftarrow \mathsf{Enc}(k, m_b)$. This $c*$ is then returned to the A as a challenge. A guesses a bit b' where $b' \in \{0,1\} \implies \{m_0, m_1\}$ that the A thinks is actually encrypted as $c*$.

The adversary A wins if $b' = b$

An encryption scheme Π is said to be CPA secure if the following holds true.

$$\Pr[A \text{ wins } Priv_{A,\Pi}^{CPA}(n)] = \frac{1}{2} + \varepsilon(n)$$

where $\varepsilon(n)$ is a negligible function.

1 PRP

Suppose F is a PRP where $K = M = \{0,1\}^\lambda$ and $C = (\{0,1\}^\lambda)^2)$.
For each

1. Describe what the corresponding Dec procedure looks like.
2. Give a proof of CPA-security of the encryption scheme, or show an attack.

Enc(k, m)	Dec(k, x, y)
$r \leftarrow_\$ \{0,1\}^\lambda$ $x := F(k, r)$ $y := r \oplus m$ **return** (x, y)	#decryption $r := F^{-1}(k, x)$ $m := y \oplus r$ **return** (m)

Theorem: If F is a secure PRP then the given scheme $\Pi(Gen, Enc, Dec)$ is a secure encryption scheme

Proof by contradiction: We will prove the following statement.
If Π is not a secure encryption scheme then, F is not a secure PRP.

Step 1: Real Scheme If Π is not a secure encryption scheme, it means that $\exists$ PPT algorithm A^{CPA} which wins the CPA game with probability $\frac{1}{2} + \varepsilon(n)$ where $\varepsilon(n)$ is a non-negligible function. For $Priv^{CPA}_{A,\Pi}$ we have the following steps

1. Pick $k \xleftarrow{\$} \{0,1\}^n$
2. Training Phase
   (a) Adversary A^{CPA} would send messages m_i
   (b) Pick a $r_i \xleftarrow{\$} \{0,1\}^\lambda$
   (c) Compute $x_i := F(k, r_i)$
   (d) Compute $y_i := r_i \oplus m_i$
   (e) Return (x_i, y_i) to A^{CPA}

3. Challenge Phase
   (a) Adversary A^{CPA} sends two messages m_0, m_1 for encryption
   (b) A bit $b \xleftarrow{\$} \{0,1\}$ is picked by the challenger
   (c) Pick $r* \xleftarrow{\$} \{0,1\}^\lambda$
   (d) Compute $x* := F(k, r*)$
   (e) Compute $y* := r * \oplus m_b$
   (f) Return $(x*, y*)$ to A^{CPA}

In this step, the Probability that A^{CPA} wins is given as follows

$$Pr[A\ wins\ Priv_{A,\Pi}^{CPA}] = \frac{1}{2} + \varepsilon(n) \text{ where } \varepsilon(n) \text{ is non-negligible.}$$

Step 2: Ideal Scheme For $Priv_{A,\Pi'}^{CPA}$ we have the following steps

1. Training Phase
   (a) Adversary A^{CPA} would send messages m_i
   (b) Pick a $r_i \xleftarrow{\$} \{0,1\}^\lambda$
   (c) Compute $x_i := TF(\cdot)$
   (d) Compute $y_i := r_i \oplus m_i$
   (e) Return (x_i, y_i) to A^{CPA}
2. Challenge Phase
   (a) Adversary A^{CPA} sends two messages m_0, m_1 for encryption
   (b) A bit $b \xleftarrow{\$} \{0,1\}$ is picked by the challenger
   (c) Pick $r* \xleftarrow{\$} \{0,1\}^\lambda$
   (d) Compute $x* := TF(\cdot)$
   (e) Compute $y* := r * \oplus m_b$
   (f) Return $(x*, y*)$ to A^{CPA}

In this step, the Probability that A^{CPA} wins is negligible since, the cipher text looks completely random since we're using a truly random function.

$$Pr[A\ wins\ Priv_{A,\Pi'}^{CPA}] = \frac{1}{2} + p(n) \text{ where } p(n) \text{ is negligible } \implies \frac{q}{2^\lambda}$$

where $\frac{q}{2^\lambda}$ is the collision probability in picking r over q queries.

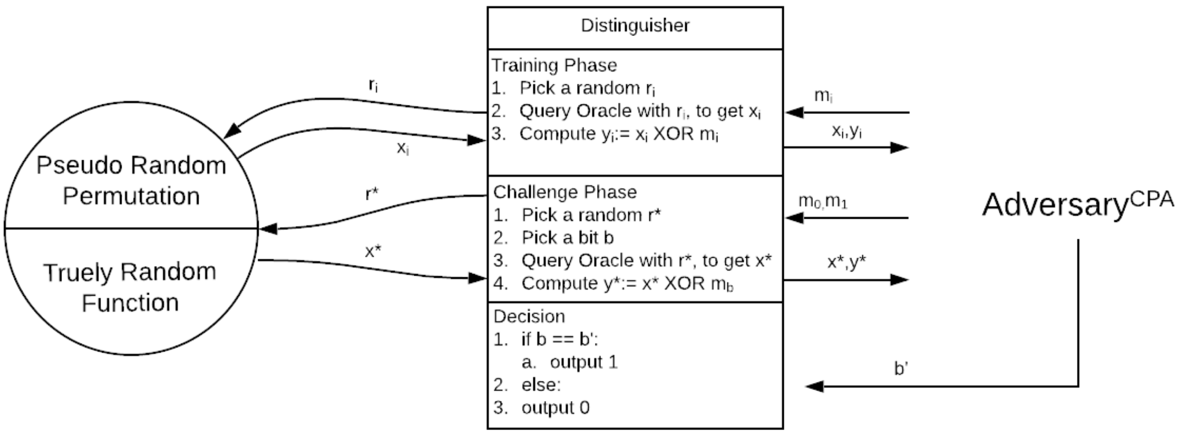

Figure 1: Reduction

Step 3: Reduction We now define a distinguisher D that would activate/simulate A^{CPA} tp break the PRP.

1. Training Phase
   (a) A^{CPA} sends messages m_i to the distinguisher D assuming it's playing the CPA game.
   (b) Distingusher D would pick a random bit $r_i \xleftarrow{\$} \{0,1\}^\lambda$
   (c) Distinguisher D would query the *Oracle* with r_i and would get back some x_i
   (d) D computes $y_i := x_i \oplus\ m_i$ and returns this (x_i, y_i) to A^{CPA}
2. Challenge Phase
   (a) A^{CPA} sends messages m_0, m_1 to the distinguisher D assuming it's playing the CPA game.
   (b) Distingusher D would pick a random bit $r* \xleftarrow{\$} \{0,1\}^\lambda$
   (c) Distinguisher D would pick a random bit $b \xleftarrow{\$} \{0,1\}$
   (d) Distinguisher D would query the *Oracle* with $r*$ and would get back some $x*$
   (e) D computes $y* := x* \oplus m_b$ and returns this $(x*, y*)$ to A^{CPA}
   (f) D outputs 1 when A^{CPA} wins the game

Step 4: Analysis of Success probability of reduction of A

1. $\mathcal{O} = F_k(\cdot) \implies$ Pseudo Random Permutation
   (a) The view of A^{CPA} is exactly the same as the view of A^{CPA} if it were playing the $Priv_{A,\Pi}^{CPA}(n)$ game
   (b) Since we know that D outputs 1, when A^{CPA} wins the game we have
$$Pr[D^{F_k(\cdot)} = 1] = Pr[A\ wins\ Priv_{\Pi}^{CPA}] = \frac{1}{2} + \varepsilon(n) \tag{1}$$
2. $\mathcal{O} = TF(\cdot) \implies$ Truly Random Function
   (a) The view of A^{CPA} is exactly the same as the view of A^{CPA} if it were playing the $Priv_{A,\Pi'}^{CPA}(n)$ game
   (b) Since we know that D outputs 1, when A^{CPA} wins the game we have
$$Pr[D^{F_k(\cdot)} = 1] = Pr[A\ wins\ Priv_{\Pi'}^{CPA}] = \frac{1}{2} + p(n) \tag{2}$$

We have the difference between *(1)* and *(2)* as follows,

$$\frac{1}{2} + \varepsilon(n) - \left(\frac{1}{2} + p(n)\right) = \varepsilon(n) - p(n) = \varepsilon'(n)$$

Where $\varepsilon'(n)$ is non-negligible. So that means that distinguisher is able to distingush between the PRP and the Truly Random Function which is contradiction. Hence the given $\Pi(Gen, Enc, Dec)$ is a secure encryption scheme.

Suppose F is a PRP where $K = M = \{0,1\}^\lambda$ and $C = (\{0,1\}^\lambda)^2)$.
For each

1. Describe what the corresponding Dec procedure looks like.
2. Give a proof of CPA-security of the encryption scheme, or show an attack.

$\mathsf{Enc}(k, m)$	$\mathsf{Dec}(k, x, r)$
$r \leftarrow_{\$} \{0,1\}^\lambda$ $x := F(k, m \oplus r) \oplus r$ **return** (r, x)	$x' = x \oplus r$ $m := F^{-1}(k, x') \oplus r$ **return** (m)

Theorem: If F is a secure PRP then the given scheme $\Pi(Gen, Enc, Dec)$ is a secure encryption scheme

Proof by contradiction: We will prove the following statement.
If Π *is not a secure encryption scheme then, F is not a secure PRP.*

Step 1: Real Scheme If Π is not a secure encryption scheme, it means that $\exists$ PPT algorithm A^{CPA} which wins the CPA game with probability $\frac{1}{2} + \varepsilon(n)$ where $\varepsilon(n)$ is a non-negligible function. For $Priv^{CPA}_{A,\Pi}$ we have the following steps

1. Pick $k \xleftarrow{\$} \{0,1\}^n$
2. Training Phase
   (a) Adversary A^{CPA} would send messages m_i
   (b) Pick a $r_i \xleftarrow{\$} \{0,1\}^\lambda$
   (c) Compute $x_i := F(k, m_i \oplus r_i) \oplus r_i$
   (d) Return (x_i, r_i) to A^{CPA}

3. Challenge Phase
   (a) Adversary A^{CPA} sends two messages m_0, m_1 for encryption
   (b) A bit $b \xleftarrow{\$} \{0,1\}$ is picked by the challenger
   (c) Pick $r* \xleftarrow{\$} \{0,1\}^\lambda$
   (d) Compute $x* := F(k, m_b \oplus \ r*) \oplus r*$
   (e) Return $(x*, r*)$ to A^{CPA}

In this step, the Probability that A^{CPA} wins is given as follows

$$Pr[A\ wins\ Priv^{CPA}_{A,\Pi}] = \frac{1}{2} + \varepsilon(n) \text{ where } \varepsilon(n) \text{ is non-negligible.}$$

Step 2: Ideal Scheme For $Priv^{CPA}_{A,\Pi'}$ we have the following steps

1. Training Phase
   (a) Adversary A^{CPA} would send messages m_i
   (b) Pick a $r_i \xleftarrow{\$} \{0,1\}^\lambda$
   (c) Compute $x_i := TF(\cdot)$
   (d) Return (x_i, r_i) to A^{CPA}
2. Challenge Phase
   (a) Adversary A^{CPA} sends two messages m_0, m_1 for encryption
   (b) A bit $b \xleftarrow{\$} \{0,1\}$ is picked by the challenger
   (c) Pick $r* \xleftarrow{\$} \{0,1\}^\lambda$
   (d) Compute $x* := TF(\cdot)$
   (e) Return $(x*, r*)$ to A^{CPA}

In this step, the Probability that A^{CPA} wins is negligible since, the cipher text looks completely random since we're using a truly random function.

$$Pr[A\ wins\ Priv^{CPA}_{A,\Pi'}] = \frac{1}{2} + p(n) \text{ where } p(n) \text{ is negligible. } \implies \frac{q}{2^\lambda}$$

where $\frac{q}{2^\lambda}$ is the collision probability in picking r over q queries.

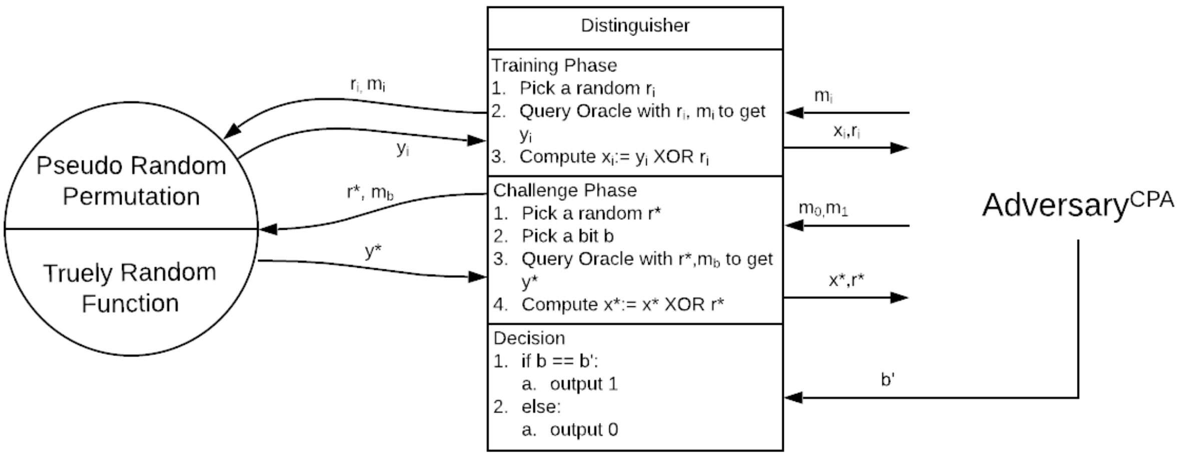

Figure 2: Reduction

Step 3: Reduction We now define a distinguisher D that would activate/simulate A^{CPA}

1. Training Phase
   (a) A^{CPA} sends messages m_i to the distinguisher D assuming it's playing the CPA game.
   (b) Distingusher D would pick a random bit $r_i \xleftarrow{\$} \{0,1\}^\lambda$
   (c) Distinguisher D would query the *Oracle* with $r_i,\ m_i$ and would get back some y_i
   (d) D computes $x_i := y_i \oplus\ r_i$ and returns this (x_i, r_i) to A^{CPA}
2. Challenge Phase
   (a) A^{CPA} sends messages m_0, m_1 to the distinguisher D assuming it's playing the CPA game.
   (b) Distingusher D would pick a random bit $r* \xleftarrow{\$} \{0,1\}^\lambda$
   (c) Distinguisher D would pick a random bit $b \xleftarrow{\$} \{0,1\}$
   (d) Distinguisher D would query the *Oracle* with $r*, m_b$ and would get back some $y*$
   (e) D computes $x_* := y * \oplus r*$ and returns this $(x*, r*)$ to A^{CPA}
   (f) D outputs 1 when A^{CPA} wins the game

Step 4: Analysis of Success probability of reduction of A

1. $\mathcal{O} = F_k(\cdot) \implies$ Pseudo Random Permutation
   (a) The view of A^{CPA} is exactly the same as the view of A^{CPA} if it were playing the $Priv_{A,\Pi}^{CPA}(n)$ game
   (b) Since we know that D outputs 1, when A^{CPA} wins the game we have

$$Pr[D^{F_k(\cdot)} = 1] = Pr[A\ wins\ Priv_{\Pi}^{CPA}] = \frac{1}{2} + \varepsilon(n) \tag{3}$$

2. $\mathcal{O} = TF(\cdot) \implies$ Truly Random Function
   (a) The view of A^{CPA} is exactly the same as the view of A^{CPA} if it were playing the $Priv_{A,\Pi'}^{CPA}(n)$ game
   (b) Since we know that D outputs 1, when A^{CPA} wins the game we have

$$Pr[D^{F_k(\cdot)} = 1] = Pr[A\ wins\ Priv_{\Pi'}^{CPA}] = \frac{1}{2} + p(n) \tag{4}$$

We have the difference between *(1)* and *(2)* as follows,

$$\frac{1}{2} + \varepsilon(n) - \left(\frac{1}{2} + p(n)\right) = \varepsilon(n) - p(n) = \varepsilon'(n)$$

Where $\varepsilon'(n)$ is non-negligible. So that means that distinguisher is able to distingush between the PRP and the Truly Random Function which is contradiction. Hence the given $\Pi(Gen, Enc, Dec)$ is a secure encryption scheme.

Suppose F is a PRP where $K = M = \{0,1\}^\lambda$ and $C = (\{0,1\}^\lambda)^2)$.
For each

1. Describe what the corresponding Dec procedure looks like.

2. Give a proof of CPA-security of the encryption scheme, or show an attack.

Enc(k, m)	Dec(k, x, y)
$r \leftarrow_\$ \{0,1\}^\lambda$ $x := F(k, r)$ $y := F(k, r) \oplus m$ **return** (x, y)	$m := x \oplus y$ **return** (m)

The given encryption scheme $\Pi(Gen, Enc, Dec)$ is *not CPA secure*. We show the attack as follows.

Attack: Since the encryption scheme $\Pi(Gen, Enc, Dec)$ is not CPA secure, it means that $\exists$ algorithm that would work as follows.

Training Phase:

1. A^{CPA} would play the CPA game by sending query message $m_i = 0^n$ to the challenger.

2. Gets back $x_i,\ y_i$ to A^{CPA}

Challenge Phase:

1. A^{CPA} will query a pair of messages $m_0 = 0^n, m_1 = 1^n$ to the challenger

2. Gets back $x*,\ y*$ to A^{CPA}

Decision:

1. A^{CPA} would output bit 1 if $y* \ \oplus x* = m_0$ else it would output 0

Analysis of A's success:

1. Case $b = 1$
   (a) $m_0 = 0^n$
   (b) $y*, x* = Enc_k(0^n) \implies y* := x * \oplus m_0 \implies y* := x * \oplus 0^n$
   (c) $0^n := y * \oplus x* \implies m_0 := y * \oplus x*$
   (d) A^{CPA} will output 1 with probability 1
2. Case $b = 0$
   (a) $m_1 = 1^n$
   (b) $y*, x* = Enc_k(1^n) \implies y* := x * \oplus m_1 \implies y* := x * \oplus 1^n$
   (c) $1^n := y * \oplus x* \implies m_1 := y * \oplus x*$
   (d) A^{CPA} will output 0 with probability 1

Conclusion: We see that the distinguisher wins with probability 1 when the bit chosen is 1 or 0, therefore the given encryption scheme is not CPA secure.

2 Block Ciphers

Consider the following block cipher modes for encryption, applied to a PRP F, where

$$F : \{0,1\}^\lambda \times \{0,1\}^\lambda \to \{0,1\}^\lambda.$$

For each

1. Describe what the corresponding Dec procedure looks like.
2. Show an attack (using CPA-security). Describe the distinguisher and compute its advantage.

$\mathsf{Enc}(k, m_1||\ldots||m_\ell)$

$r_0 \leftarrow_\$ \{0,1\}^\lambda$
$c_0 := r_0$
for $i = 1$ *to* ℓ **do**
  $r_i := F(k, m_i)$
  $c_i := r_i \oplus r_{i-1}$
end
return $c_0||\ldots||c_\ell$

$\mathsf{Dec}(k, c_0||\ldots||c_\ell)$

$r_0 := c_0$
for $i = 1$ *to* ℓ **do**
  $r_i := c_i \oplus r_{i-1}$
  $m_i := F^{-1}(k, r_i)$
end
return $m_1||\ldots||m_\ell$

Attack: The given algorithm is *not CPA secure*, this means that $\exists$ distinguisher D that would work as follows:

Training Phase:

1. Query Oracle $\mathcal{O}$ with messages $m_i := m_1||\ldots||m_\ell = 0^n$
2. Gets back $c^m := c_0^m||\ldots||c_\ell^m$

Challenge Phase:

1. Query Oracle $\mathcal{O}$ with a pair of messages $m_0 = 0^n, m_1 = 1^n$
2. Gets back $c^b := c_0^b||\ldots||c_\ell^b$

Decision:

1. Output bit 1 if $c_i^b \oplus c_{i-1}^b = c_i^m \oplus c_{i-1}^m$ else it would output 0

Analysis of D's success:

1. Case $b = 1$
   (a) $m_0 = 0^n$
   (b) $c^b := c_0^b || \ldots || c_\ell^b$
   (c) $c_i^b \oplus c_{i-1}^b = F_k(m_i) = c_i^m \oplus c_{i-1}^m$
   (d) D will output 1 with probability 1
2. Case $b = 0$
   (a) $m_1 = 1^n$
   (b) $c^b := c_0^b || \ldots || c_\ell^b$
   (c) $c_i^b \oplus c_{i-1}^b = F_k(m_i) = c_i^m \oplus c_{i-1}^m$
   (d) D will output 0 with probability 1

Conclusion: We see that the distinguisher wins with probability 1 when the bit chosen is 1 or 0, therefore the given encryption scheme is not CPA secure.

Consider the following block cipher modes for encryption, applied to a PRP F, where

$$F : \{0,1\}^\lambda \times \{0,1\}^\lambda \to \{0,1\}^\lambda.$$

For each

1. Describe what the corresponding Dec procedure looks like.
2. Show an attack (using CPA-security). Describe the distinguisher and compute its advantage.

$\mathsf{Enc}(k, m_1 \| \dots \| m_\ell)$	$\mathsf{Dec}(k, c_0 \| \dots \| c_\ell)$

$c_0 \leftarrow_{\$} \{0,1\}^\lambda$
for $i = 1$ *to* ℓ **do**
  $c_i := F(k, m_i) \oplus c_{i-1}$
end
return $c_0 \| \dots \| c_\ell$
	for $i = 1$ *to* ℓ **do**
	  $x_i := c_i \oplus c_{i-1}$
	  $m_i := F^{-1}(k, x_i)$
	end
	return $m_1 \| \dots \| m_\ell$

Attack: The given algorithm is *not CPA secure*, this means that $\exists$ distinguisher D that would work as follows:

Training Phase:

1. Query Oracle $\mathcal{O}$ with messages $m_i := m_1 \| \dots \| m_\ell = 0^n$
2. Gets back $c^m := c_0^m \| \dots \| c_\ell^m$

Challenge Phase:

1. Query Oracle $\mathcal{O}$ with a pair of messages $m_0 = 0^n, m_1 = 1^n$
2. Gets back $c^b := c_0^b \| \dots \| c_\ell^b$

Decision:

1. Output bit 1 if $c_i^b \oplus c_{i-1}^b = c_i^m \oplus c_{i-1}^m$ else it would output 0

Analysis of D's success:

1. Case $b = 1$
   (a) $m_0 = 0^n$
   (b) $c^b := c_0^b || \dots || c_\ell^b$
   (c) $c_i^b \oplus c_{i-1}^b = F_k(m_i) = c_i^m \oplus c_{i-1}^m$
   (d) D will output 1 with probability 1
2. Case $b = 0$
   (a) $m_1 = 1^n$
   (b) $c^b := c_0^b || \dots || c_\ell^b$
   (c) $c_i^b \oplus c_{i-1}^b = F_k(m_i) = c_i^m \oplus c_{i-1}^m$
   (d) D will output 0 with probability 1

Conclusion: We see that the distinguisher wins with probability 1 when the bit chosen is 1 or 0, therefore the given encryption scheme is not CPA secure.

3 CPA Security

Suppose Σ is an encryption scheme and $\mathcal{A}$ is a program which can recover the key from a chosen plaintext attack. In other words the game for $\mathcal{A}$ looks like:

For polynomially many i.

1. $\mathcal{A}$ queries the challenger on m_i.
2. challenger returns $c_i := \Sigma.\mathsf{Enc}(k, m_i)$.

Finally, $\mathcal{A}$ outputs k.

Prove that Σ does not have CPA security.

We assume that the Encryption scheme works like this

$\mathsf{Enc}(k, m)$

$r \leftarrow_{\$} \{0,1\}^\lambda$
$x := F(k, r)$
$c := r \oplus m$
return (x, c, k)

Attack: The given algorithm is *not CPA secure*, this means that $\exists$ distinguisher D that would work as follows:

Training Phase:

1. Query Oracle $\mathcal{O}$ with messages m_i
2. Gets back $c_i,\ x_i$
3. Get the key k

Challenge Phase:

1. Query Oracle $\mathcal{O}$ with a pair of messages $m_0 = 0^n, m_1 = 1^n$
2. Gets back $c_b,\ x_b$

Decision:

1. Output bit 1 if $F_k^{-1}(x_b) \oplus c_b = m_0$ else it would output 0

Analysis of D's success:

1. Case $b = 1$
   (a) $m_0 = 0^n$
   (b) $c_b := r \oplus m_b = F_k^{-1}(x_b) \oplus m$
   (c) D will output 1 with probability 1
2. Case $b = 0$
   (a) $m_0 = 1^n$
   (b) $c_b := r \oplus m_b = F_k^{-1}(x_b) \oplus m$
   (c) D will output 0 with probability 1

Conclusion: We see that the distinguisher wins with probability 1 when the bit chosen is 1 or 0, therefore the given encryption scheme is not CPA secure.

——x——