**CS-NC-616: Foundations of Cryptography** **(2015-16: Term II)**

# LECTURE 13: TUTORIAL II - PART I

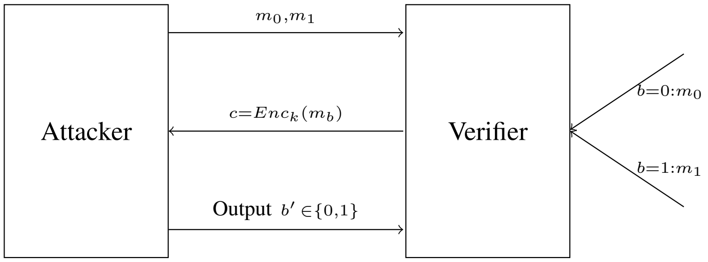
15/02/2016

*Lecturer: Ashish Choudhury* *Scribe: Devarsh KU, Shiva Ashish T*

# Recap

- **Perfectly-Secure Scheme:** A given scheme $\prod$ is perfectly secure if:

  For every non-PPT attacker i.e. an attacker with unbounded computing capability, the scenario to prove security can be modelled in the form of an indistinguishability game, played between an attacker and a verifier. It is represented pictorially as follows:

  Outcome of the experiment is 1 if $b' = b$.

  $\prod$ is perfectly secure if $Pr\left[PrivK^{eav}_{A,\prod} = 1\right] \leq \frac{1}{2}$ ( i.e.Probability that an attacker wins the game is less than or equal to 1/2). But, if there exists an attacker who can win the game with probability greater than $1/2$, then $\prod$ is not perfectly secure.

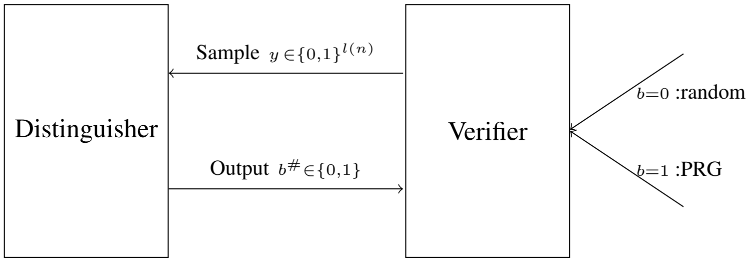
  $Enc$ algorithm requires length of key and message to be the same.

  Disadvantages:

  - Key cannot be reused.
  - Key is as large as the message.

- **PRG(Pseudorandom generators):** PRG takes a key of small size and expands it in a seemingly random way to the size of the message for the encryption scheme.

  Verification of whether a relation is a PRG or not can be modelled in the form of an indistinguishability game, played between an distinguisher and a verifier. It is represented pictorially as follows:

$G(k)$ is a PRG iff

$$|Pr\left[D(r)=1\right]-Pr\left[D\left(G(s)\right)=1\right]|\leq negl(n)$$

where, 1 as output signifies the use of a PRG,

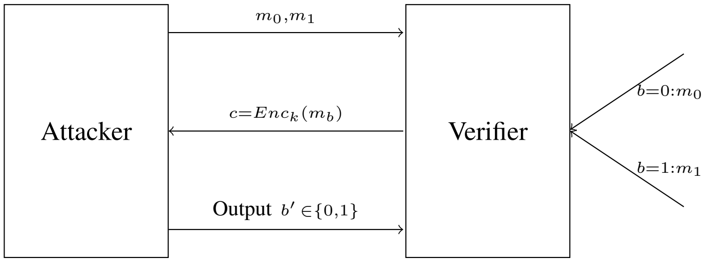
$Pr\left[D(r)=1\right]$signifies that the attacker outputs 1 while interacting with a TRG and

$Pr\left[D\left(G(s)\right)=1\right]$ signifies the attacker outputs 1 while interacting with a PRG.

This implies a PRG behaves like a TRG (Truly Random Generator) with very high probability.

- **COA (Cipher-text Only Attack) Secure Scheme:** A given scheme $\prod$ is COA single-secure if:

  For every PPT attacker i.e. an attacker with bounded computing capability, the scenario to prove security can be modelled in the form of an indistinguishability game, played between an attacker and a verifier. It is represented pictorially as follows:

  Outcome of the experiment is 1 if $b'=b$.

  $\prod$ is COA-single secure if $Pr\left[PrivK^{COA}_{A,\prod}=1\right]\leq\frac{1}{2}+negl(n)$ ( i.e.Probability that an attacker wins the game is less than or equal to 1/2). But, if there exists an attacker who can win the game with probability greater than $1/2$, then $\prod$ is not secure.

  Use of a PRG ensures that the key size is much smaller than the message.

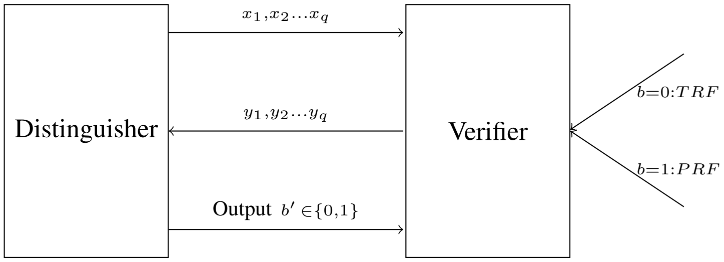
  COA single-message security does not imply COA multi-message security because the later would require a randomised $Enc$ scheme but single-message security is possible with a deterministic algorithm.

  Disadvantages:

  - Key cannot be reused.

- **PRF (Pseudo Random Function):** A PRF is a keyed function that takes in an argument along with a key of small size and its output is used in the $Enc$ scheme. The argument apart from the key contributes to the additional randomness that ensures the same message does not get encrypted to the same cipher-text with very high probability.

  Verification of whether a function is a PRF or not can be modelled in the form of an indistinguishability game, played between an distinguisher and a verifier. It is represented pictorially as follows:

$F_k(.)$ is a PRF iff

$$\left| Pr\left[D^{F_k(.)}\left(1^n\right)=1\right] - Pr\left[D^{f()}\left(1^n\right)=1\right]\right| \leq negl(n)$$

where, 1 as output signifies the use of a PRF,

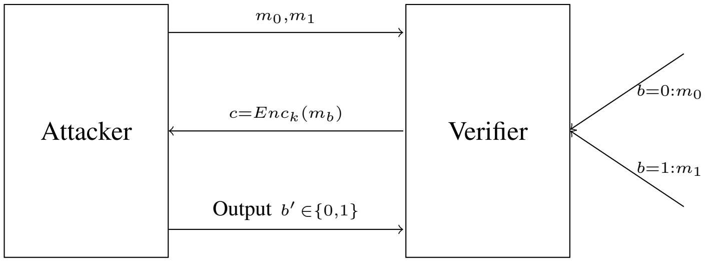
$Pr\left[D^{F_k(.)}\left(1^n\right)=1\right]$signifies that the attacker outputs 1 while interacting with a PRF and

$Pr\left[D^{f()}\left(1^n\right)=1\right]$ signifies the attacker outputs 1 while interacting with a TRF.

This implies a PRF behaves like a TRF (Truly Random Generator) with very high probability.

- **CPA(Chosen Plain Text) Secure Scheme:** A given scheme $\prod$ is CPA single-secure if:

  For every PPT attacker i.e. an attacker with bounded computing capability, the following indistinguishability game holds:the scenario to prove security can be modelled in the form of an indistinguishability game, played between an attacker and a verifier. It is represented pictorially as follows:



  Outcome of the experiment is 1 if $b' = b$.

  $\prod$ is CPA secure if $Pr\left[PrivK^{CPA}_{A,\prod} = 1\right] \leq \frac{1}{2} + negl(n)$ ( i.e.Probability that an attacker wins the game is less than or equal to 1/2). But, if there exists an attacker who can win the game with probability greater than $1/2$, then $\prod$ is not secure.

  Use of a PRF ensures that the key size is much smaller than the message and ensures key reusability. This involves another element of randomness in the use of PRF.

  CPA single-message security does imply CPA multi-message security because of the use of a randomised $Enc$ scheme which encrypts the same message to different cipher-texts with very high probability.

# Reductions

A method in which one instance of a problem is reduced to an instance of another problem. It is a common proof strategy used in cryptography. It is mostly used to prove statements of the form:

- If $\prod$ is secure, then $\prod'$ is secure: reduction involves conversion of problem of $\prod'$ to an instance of $\prod$. It can then be argued that if $\prod'$ is not secure, then $\prod$ is not secure either.
- If some condition $A$ holds, then $\prod$ is secure: this is used to prove security in encryption schemes involving the use of a PRG or a PRF. It can then be argued that if $\prod$ is not secure, then $A$ does not hold.
- If condition $A$ holds then $A2$ holds: this is used to show new schemes based on old schemes satisfying certain properties, as seen in the case of a PRG used to define a new PRG. It can then be argued that if $A2$ does not hold, then neither does $A$.

Detailed examples with explanations give better clarity on usage of reductions. Reductions are actively seen in this problem set.

# Question 1

$G : \{0,1\}^n \to \{0,1\}^{\ell(n)}$ be a PRG. Are the following constructions also a PRG:

a. $G'(k) \overset{def}{=}$ reverse($G(k)$)

b. $G'(k)$ = $G'(k) \parallel 0$

# Answer

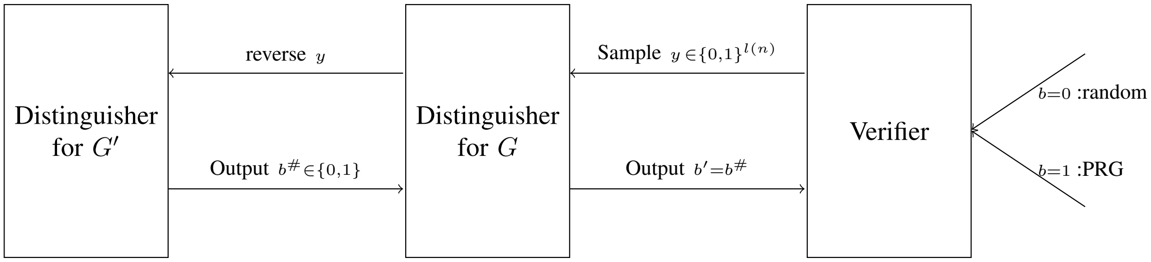
$G(k)$ is given to be a PRG. In order to argue that a new scheme is indeed a PRG, we have to provide a reduction based proof of security. However, to show that a new scheme is not a PRG, it is sufficient to provide a distinguishing strategy that can break this new PRG with non-negligible probability.

a. **Intuition**
Since $G(k)$ is a PRG, it produces a bitstring that is indistinguishable from a random bitstring. Intuitively, it seems that the reverse of such a bitstring is also indistinguishable from a random bitstring. Therefore, we will prove that the scheme $G'(k) \overset{def}{=}$ reverse($G(k)$) is indeed a PRG.

**Proof by Contrapositive**
To show: If $G'(k)$ is not a PRG, then neither is $G(k)$
Let $G'(k)$ not be a PRG. Therefore, there is a distinguisher $D_{G'}$ such that

$$|Pr[D_{G'}(1^n,\, r) \;=\; 1] - Pr[D_{G'}(1^n,\, G'(k)) \;=\; 1]| \geq f(n) \tag{1}$$

where $f(n)$ is a non-negligible function.
Design a new distinguisher $D_G$ for the scheme $G(k)$ which reduces an instance of a $G(k)$ problem to an instance of a $G'(k)$ problem as follows:



It is seen here that $D_G$ attains a bitstring of length $l(n)$ from a verifier. He then reverses this sample bitstring and sends it as input to $D_{G'}$. $D_{G'}$ views this as an instance of the problem $G'(k)$, and breaks it with non-negligible probability, passing $b^{\#} \in \{0,1\}$ as his output. $D_G$ relies on this property of $D_{G'}$, and blindly copies the output by passing $b' = b^{\#}$ as output to the verifier.
The probability that $D_G$ breaks $G(k)$ is

$$|Pr[D_G(1^n,\, r) \;=\; 1] - Pr[D_G(1^n,\, G(k)) \;=\; 1]|$$

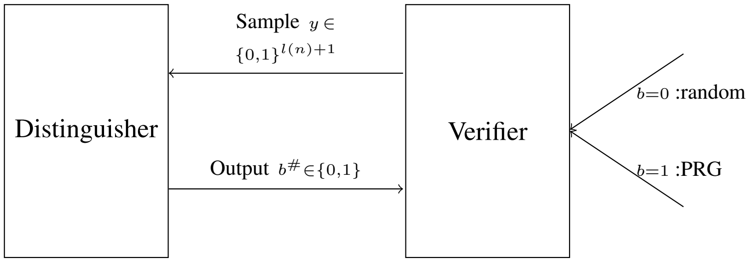
However, this is the same as the LHS of equation 1. Therefore,

$$|Pr[D_G(1^n,\, r) \;=\; 1] - Pr[D_G(1^n,\, G(k)) \;=\; 1]| \geq f(n)$$

where $f(n)$ is a non-negligible function.
However, this is a contradiction to the given statement that $G(k)$ is a PRG. Therefore, $G'(k)$ is a PRG.

b. **Intuition**
The new scheme appends a constant character to every pseudorandom number generated. A distinguisher could potentially look at this constant last character to guess if the number game from a PRG or a TRG.

**Distinguisher strategy**
A distinguisher can be designed for this new scheme $G'(k)$. If the last bit of the sample bitstring is 0, return $b' = 1$ i.e y is from $G'(k)$. If it is not, then return $b' = 0$ i.e sample bitstring is from TRG.



In this case,

$$Pr[D_{G'}(1^n, G'(k)) = 1] = 1$$

$$Pr[D_{G'}(1^n, r) = 1] = \frac{1}{2}$$

because if the last bit of the sample bitstring is 0, distinguisher guesses correctly every time and if the last bit is 1, he guesses correctly half the time (the sample can be from either TRG or PRG with equal probability). Therefore, the difference is

$$|Pr[D_{G'}(1^n, r) = 1] - Pr[D_{G'}(1^n, G'(k)) = 1]| = 1 - \frac{1}{2} = \frac{1}{2}$$

which is non-negligible. Therefore, $G'(k)$ is not a PRG.

# Question 2

Let $\prod$ = (Gen, Enc, Dec) be a COA-secure scheme. Determine whether the following modifications of Enc retain COA-security:

a. Enc'$_k$(m) $\overset{def}{=}$ Enc$_k$(m)||k.

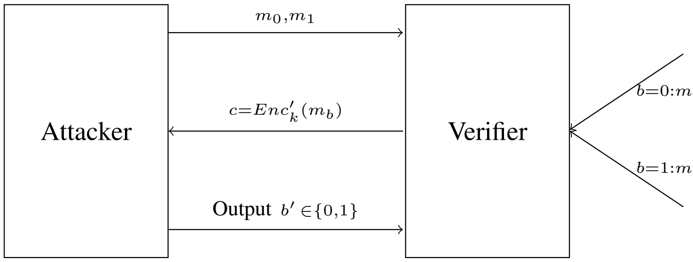
b. Enc'$_k$(m)$\overset{def}{=}$ 0||Enc$_k$(m).

# Answer

The scheme $\prod$ is given to be COA-secure. In order to argue that a new scheme is COA-secure as well, we have to provide a reduction based proof of security. However, to show that a new scheme is not COA-secure, it is sufficient to provide an attacker that can break this new scheme with non-negligible probability.

a. **Intuition**
It is seen that the key itself is part of the ciphertext in this case. An attacker who has access to the ciphertext can simply extract the key and decrypt the ciphertext.

**Attacker strategy**
The attack is modelled using the indistinguishability game. The attacker's strategy follows the intuition exactly. He sends two messages $m_0$ and $m_1$.



On receiving the ciphertext, he extracts the key from it and encrypts both the messages. On comparing the ciphertext obtained (with the key removed) and the new encryptions, the attacker can identify with certainty which message was encrypted.

$$Pr[PrivK^{COA}_{A,\prod'}(1^n) = 1] = 1$$

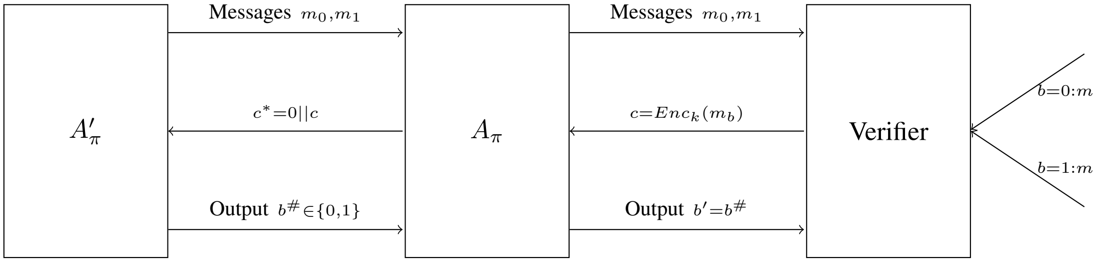
$\therefore$ The new scheme is not COA-secure.

b. **Intuition**
A $'0'$ appended to the ciphertext of a COA secure scheme will not alter the nature of security, as the attacker gains no additional information from the presence of that extra character. He must still break the remaining ciphertext, which is not possible.

Let $\prod'$ : $Enc'_k(m) \overset{def}{=} 0 \,||\, Enc_k(m)$
**Proof by Contrapositive**
To show: If $\prod'$ is not a COA-secure, then neither is $\prod$
Let $\prod'$ not be COA-secure. Therefore, there is an attacker $A_{\prod'}$ such that

$$Pr[PrivK^{COA}_{A,\prod'}(1^n) = 1] \geq \frac{1}{2} + f(n) \tag{2}$$

where $f(n)$ is a non-negligible function.
Design a new attacker $A_{\prod}$ for the scheme $\prod$ which reduces an instance of a $\prod$ problem to an instance of a $\prod'$ problem as follows:



It is seen here that $A_{\prod'}$ passes two messages $m_0$and $m_1$to $A_{\prod}$. $A_{\prod}$ passes on the messages as is to the verifier, for which it receives a ciphertext. $A_{\prod}$ then appends a 0 to the start of the ciphertext and passes it on to

$A_{\prod'}$. $A_{\prod'}$ views this as an instance of the problem $\prod'$, and breaks it with non-negligible probability, passing $b^{\#} \in \{0,1\}$ as his output. $A_{\prod}$ relies on this property of $A_{\prod'}$, and blindly copies the output by passing $b' = b^{\#}$ as output to the verifier.
Probability that $A_{\prod}$ wins the game is

$$Pr[PrivK^{COA}_{A,\prod}(1^n) = 1]$$

This is the same as the LHS of equation 2.

$$Pr[PrivK^{COA}_{A,\prod'}(1^n) = 1] = Pr[PrivK^{COA}_{A,\prod}(1^n) = 1]$$

$$\geq \frac{1}{2} + f(n)$$

This is a contradiction, as $\prod$ is known to be secure. $\therefore$ $\prod'$ is secure.

# Question 3

Consider a PRG G :$\{0,1\}^* \rightarrow \{0,1\}^*$ . Then prove whether the following construction is a PRG or not:

$$G_1(s,\, t) \overset{def}{=} G(s) \| t$$

where $|s| = |t|$ and s,t are random

# Answer

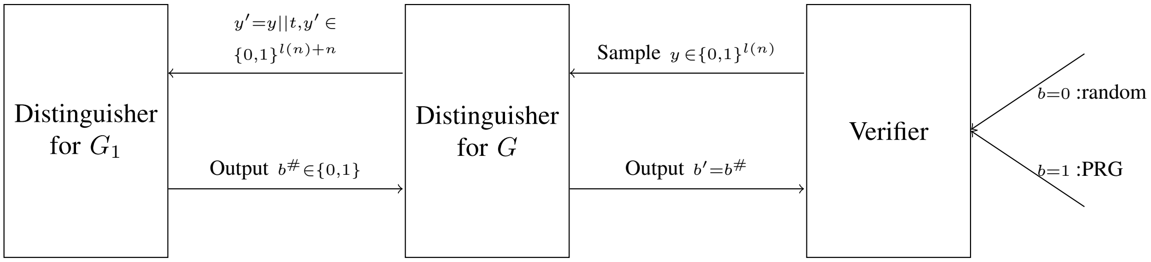
$G(s)$ is given to be a PRG. In order to argue that a new scheme is indeed a PRG, we have to provide a reduction based proof of security. However, to show that a new scheme is not a PRG, it is sufficient to provide a distinguishing strategy that can break this new PRG with non-negligible probability.

**Intuition**
Since $G(s)$ is a PRG, it produces a bitstring that is indistinguishable from a random bitstring. Intuitively, it seems that this birstring appended with a truly random bitstring is also indistinguishable from a random bitstring. Therefore, we will prove that the scheme $G_1(s,\, t) \overset{def}{=} G(s) \| t$ is indeed a PRG.

**Proof by Contrapositive**
To show: If $G_1(s,\, t)$ is not a PRG, then neither is $G(s)$
Let $G_1(s,\, t)$ not be a PRG. Therefore, there is a distinguisher $D_{G'}$ such that

$$|Pr[D_{G_1}(1^n,\, r) \,=\, 1] - Pr[D_{G_1}(1^n,\, G_1(s,\, t)) \,=\, 1]| \geq f(n) \tag{3}$$

where $f(n)$ is a non-negligible function.
Design a new distinguisher $D_G$ for the scheme $G(s)$ which reduces an instance of a $G(s)$ problem to an instance of a $G_1(s,\, t)$ problem as follows:



It is seen here that $D_G$ attains a bitstring of length $n$ from a verifier. He then appends a random bitstring of the same length and passes the new bitstring of length $l(n) + n$ to $D_{G_1}$. $D_{G_1}$ views this as an instance of the problem $G_1(s,\, t)$, and breaks it with non-negligible probability, passing $b^{\#} \in \{0, 1\}$ as his output. $D_G$ relies on this property of $D_{G_1}$, and blindly copies the output by passing $b' = b^{\#}$ as output to the verifier.
The probability that $D_G$ breaks $G(s)$ is

$$|Pr[D_G(1^n,\, r) \,=\, 1] - Pr[D_G(1^n,\, G(k)) \,=\, 1]|$$

However, this is the same as the LHS of equation 3. Therefore,

$$|Pr[D_G(1^n,\, r) \,=\, 1] - Pr[D_G(1^n,\, G(k)) \,=\, 1]| \geq f(n)$$

where $f(n)$ is a non-negligible function.
However, this is a contradiction to the given statement that $G(s)$ is a PRG. Therefore, $G_1(s,\, t)$ is a PRG.

# Question 4

Consider a PRG $G$ and define a function $G'(s)$ to be the output of $G$ truncated to $n$ bits (where $|s| \,=\, n$). Prove whether the following function is a PRF or not:

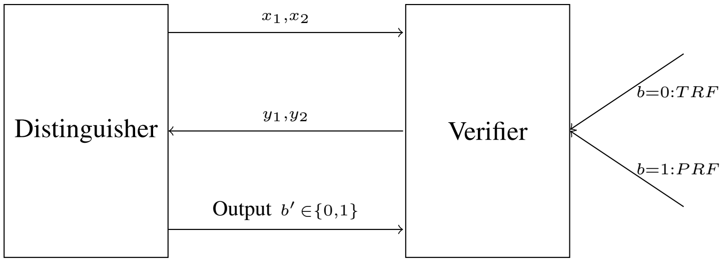
$$F_k(x) = G'(k) \oplus x.$$

# Answer

$G(s)$ is given to be a PRG. Therefore, $G'(s)$ is also pseudorandom (as it is just a truncated version of $G(s)$). In order to argue that the new function is indeed a PRF, we have to provide a proof of security where an attacker cannot differentiate a string to be from TRF or a PRF with a probablilty greater than $\frac{1}{2} + negl(n)$. However, to show that a new scheme is not a PRF, it is sufficient to provide a distinguishing strategy that can break this new PRF with non-negligible probability.

**Intuition**
The new scheme simply performs the XOR of the message with the same pseudorandom bitstring give my $G'(k)$. This property can be exploited in a way that an attacker can perform the XOR of two outputs and of $F_k(x)$ and check if it matches the XOR of the two inputs to PRF.

**Distinguisher strategy**
A distinguisher can be designed for this new scheme $F_k(x)$. An attacker sends two inputs $x_1$and $x_2$ to a verifier and they are encrypted using a PRF or a TRF. The attacker simply performs a an XOR of two outputs $y_1$and $y_2$ of $F_k(x)$ and checks if it matches the XOR of the two inputs to PRF. It it does match, output is $b' = 1$ i.e. PRF, if not output is $b' = 0$ i.e. TRF. This is because effect of same $G'(k)$ is void as it is used in XOR multiple times. The indistinguishability game is as follows:



For the scheme to be a PRF,

$$\left|Pr[D^{F_k(.)}(1^n)\ =\ 1] - Pr[D^{f(.)}(1^n,)\ =\ 1]\right| \leq negl(n)$$

For the above attacker,

$$Pr[D^{F_k(.)}(1^n)\ =\ 1] = 1$$

because $y_1 \oplus y_2 = x_1 \oplus x_2$ always holds.

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

because $y_1 = x_1 \oplus x_2 \oplus y_2$ holds only for one assignment of $y$. So,

$$\left|Pr[D^{F_k(.)}(1^n)\ =\ 1] - Pr[D^{f(.)}(1^n,)\ =\ 1]\right| = 1 - \frac{1}{2^n} \nleq negl(n)$$

$\therefore$ $F_k(x)$ is not a PRF.

# Question 5

Let F be a length-preserving PRP; define the following fixed-length encryption scheme for encrypting messages of $n/2$ bits: on input m $\in \{0,1\}^{n/2}$and k $\in \{0,1\}^n$, the encryption algorithm Enc selects a random string r $\in \{0,1\}^{n/2}$and outputs $c \leftarrow F_k(r\|m)$.

a. Write down the corresponding decryption algorithm.

b. Prove whether this scheme is CPA-secure or not.

# Answer

a. The ciphertext $c$ is of the form $F_k(r\|m)$ where r is the random pad used in the encryption process. The decryption algorithm would be to apply the inverse of the PRP on $c$ (which is known to exist because of the properties of PRP) and remove the first $n/2$ bits of the resulting inverse.

   Decryption algorithm is as follows:

   Let $c$ be the recieved cipher-text. Then

$$c^* = F_k^{-1}(c)$$

   $c*$is of the form $r||m$. So, the original message sent is the last $n/2$ bits of $c*$.

b. The function $F$ is given to be a PRP. In order to argue that the new scheme defined using F is CPA-secure, we have to provide a reduction based proof of security. However, to show that a new scheme is not CPA-secure, it is sufficient to provide an attacker that can break this new scheme with non-negligible probability.

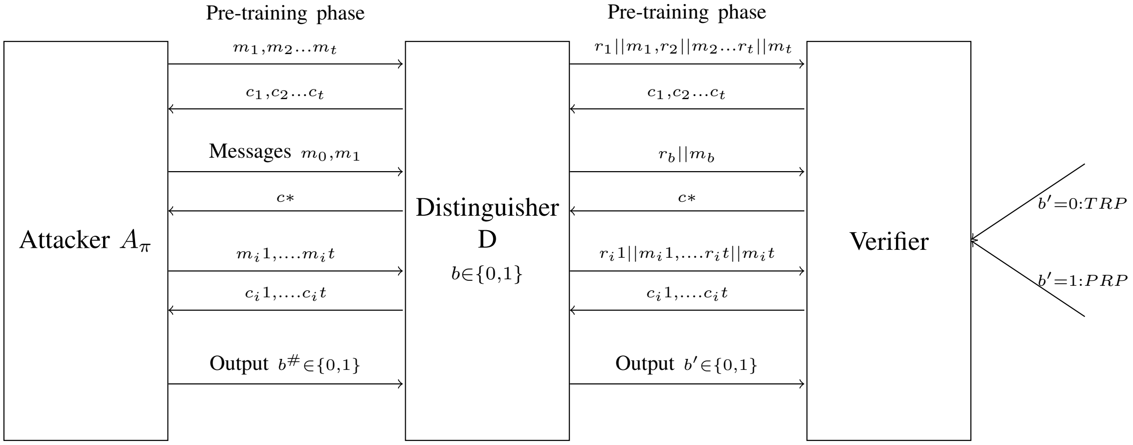
**Intuition**
The function $F$ is given to be a PRP. The new scheme $\prod$ appends a random r bits to the start of the message. It then invokes the function F on this concatenated string. Since F is a PRF, the bitstring generated by F is indistinguishable from a truly random bitstring. Therefore, the ciphertext is likely to be CPA-secure.

Let the given scheme be denoted by $\prod$.

**Proof by Contrapositive**
To show: If $\prod$ is not a CPA-secure, then $F$ is not a PRP
Let $\prod$ not be CPA-secure. Therefore, there is an attacker $A_{\prod}$ such that

$$Pr[PrivK^{CPA}_{A,\prod}(1^n) = 1] \geq \frac{1}{2} + f(n) \tag{4}$$

where $f(n)$ is a non-negligible function.
Design a new distinguisher $D$ for the PRP $F$ which uses the attacker to the scheme $\prod$ to break the PRF $F$ as follows:



The distinguisher $D$ acts like a verifier to the attacker $A_{\prod}$. $D$ takes messages $m_1$, $m_2$... $m_t$ as input from $A_{\prod}$ as part of the pre-training phase. The distinguisher simply appends the messages to the random bitstrings $r_1$, $r_2$... $r_t$ and passes it to the verifier of the PRP. The verifier randomly selects to encrypt them using either the PRP ot a TRP, and passes back ciphertexts $c_1$, $c_2$... $c_t$ to $D$, who passes them as is to the attacker $A_{\prod}$. This completes the pre-training phase.
$A_{\prod}$ now chooses two messages $m_0$and $m_1$ and sends them to $D$. $D$ chooses one of these messages at random by picking $b \in \{0, 1\}$. It also chooses a random bitstring $r_b$ and passes the message $r_b \| m_b$ to the verifier. The verifier passes back a ciphertext $c^*$, which is passed to $A_{\prod}$ as is by $D$.
After this stage, a post-training phase also takes place exactly similar to the pre-training phase.

**Distinguisher's strategy:** $A_{\prod}$ views this as an instance of the problem $\prod$, and breaks it with non-negligible probability, passing $b^{\#} \in \{0, 1\}$ as his output. $D$ sees if $b^{\#}$matched the $b$ chosen by him during the game. If it does, then $D$ passes $b' = 1$ i.e PRP to the verifier. If it does not match, then $D$ passes $b' = 0$ i.e TRP to the verifier.

- **Case 1:** If $D$ queried a PRP,

  $$Pr\left[D^{F_k(\cdot)}\left(1^n\right) = 1\right] = Pr[PrivK^{CPA}_{A,\prod}(1^n) = 1] = \frac{1}{2} + f(n)$$

  This is because , the view of $A_{\prod}$ will be exactly the same as when the attacker playes the actual CPA-indistinguishability game against scheme $\prod$.
- **Case 2:** If D queried a TRP,
  This case can be modelled as another indistinguishability game where:

  $$\begin{gathered}\overline{\pi} = (Gen,\, Enc,\, Dec) \text{ using a TRP } f\\ Enc:\ random\, r \in \{0,1\}^{n/2}\\ c \leftarrow \left(f\left(r||m\right), r\right)\\ \overline{A}:\ \text{Attacker for } \overline{\pi}\end{gathered}$$

  In this game the 2 possibilities arise for an attacker:

  - **Possibility 1**: If $(m_b,\, r_b)$ was encrypted in any of the training phase.

  In this case on comparing the obtained cipher-text $c*$with those from the training phase, $m_b$can be obtained. Here $\bar{A}_{\overline{\pi}}$can ensure $b^{\#} = b$ with probability

  $$\frac{q\left(n\right)}{2^n} = negl(n)$$

  where $q(n)$ is the number of queries that have different $r_i$ from the pre and post training phases. And the total number of possibilities is $2^n$because the function $f$ takes a string of length $n$ which includes the message and the random choice $r_i$ .

  - **Possibility** 2: If $(m_b,\, r_b)$ was never encrypted in any of the training phase cases.
    In this case, $c*$ is an evaluation of $f$ on a truly random input. So, in this case $\bar{A}_{\overline{\pi}}$can ensure $b^{\#} = b$ with probability exactly $\frac{1}{2}$.
    So, when D queried a TRP,

    $$Pr\left[D^{f(\cdot)}\left(1^n\right) = 1\right] = Pr[PrivK^{CPA}_{\overline{A},\prod}(1^n) = 1] = \frac{1}{2} + \frac{q(n)}{2^n}$$

So, for a PRP,

$$\begin{aligned}\left|Pr\left[D^{F_k(\cdot)}\left(1^n\right) = 1\right] - Pr\left[D^{f(\cdot)}\left(1^n\right) = 1\right]\right| = &\left|\frac{1}{2} + f(n) - \frac{1}{2} - \frac{q(n)}{2^n}\right|\\ =&f(n) - \frac{q(n)}{2^n}\\ =&f(n) - negl(n)\\ \nleq& negl(n)\end{aligned}$$

This implies $F_k(.)$ is not a PRP. This completes the proof of contrapositive.
$\therefore \prod$ is CPA secure.