Module Title: Introduction to Modern Cryptography
Exam Diet: May 2024

1. **(Shift cipher)** Since the Shift cipher encrypts each letter of the plaintext with the same key, the distance between the letters of the plaintext is also preserved between the letters of the ciphertext. Eve observes that in STOP the distance (shift) between the letters is $1, 21, 1, 3$, while for DEMO the distances are $1, 8, 2, 15$. As a result, by looking at the ciphertext and inspecting the distances between the letters (aside from the first two letters), she can safely decide if the selected message was DEMO or STOP.

2. **PRF and PRG**

   (a) (PRF)

   (i). F' is not a pseudorandom function. To see this, suppose $H^\star$ is a hash function for $\{0,1\}^n \to \{0,1\}^{\frac{n}{2}}$, we construct the following hash function H: H output $H^\star([x]^{n}_{1})||0^{\frac{n}{2}}$. In this case, H is still a hash function. However, with the function H we defined, we formally define the following attacker $\mathcal{A}$ given 1^n and access to some function g:
   $\underline{\mathcal{A}^g(1^n):}$
   - Query x and receive $y \leftarrow F'(k, x)$.
   - Output 1 if and only if $[y]^{n}_{\frac{n}{2}+1} = 0^{\frac{n}{2}}$.

   As shown above, we have $\Pr_{k \xleftarrow{\$} \{0,1\}^n}[\mathcal{A}^{F'(k,\cdot)}(1^n) = 1] = 1$. But when g is a random function then y is independent, uniform string of length n, and so the probability that they are the same is exactly $2^{-\frac{n}{2}}$. Thus, $\Pr_{f \xleftarrow{\$} \mathsf{Func}}[\mathcal{A}^{f(\cdot)}(1^n) = 1] = 2^{-\frac{n}{2}}$, and the difference

   $$\left| \Pr_{k \xleftarrow{\$} \{0,1\}^n}[\mathcal{A}^{F'(k,\cdot)}(1^n) = 1] - \Pr_{f \xleftarrow{\$} \mathsf{Func}}[\mathcal{A}^{f(\cdot)}(1^n) = 1] \right| = 1 - 2^{-\frac{n}{2}}$$

   is not negligible.

   (ii). F'' is a pseudorandom function. We prove this by showing that if the adversary can distinguish F'', then he can also distinguish F. Suppose adversary $\mathcal{A}$ can distinguish F'' with non-negligible probability, i.e.,

   $$\left| \Pr_{k \xleftarrow{\$} \{0,1\}^n}[\mathcal{A}^{F''(k,\cdot)}(1^n) = 1] - \Pr_{f \xleftarrow{\$} \mathsf{Func}}[\mathcal{A}^{f(\cdot)}(1^n) = 1] \right| > \mathsf{negl}(n),$$

   We let $\mathcal{D}$ simulate the experiment of F'' for $\mathcal{A}$:
   - $\mathcal{A}(1^n)$ outputs x with length n.
   - Upon receiving x from $\mathcal{A}$, $\mathcal{D}$ computes $H(x||0^n)$, and hands it to the challenger.
   - $\mathcal{D}$ receive y from the challenger (y is either sampled from the evaluation of F, or from U_n), and sends it to $\mathcal{A}$; $\mathcal{D}$ outputs 1 iff. $\mathcal{A}$ outputs 1.

   Let $\mathcal{D}^F$ denote the output of $\mathcal{D}$ when the values $\mathcal{D}$ receives are computed using a PRF F. We have $\Pr[\mathcal{D}^F = 1] = \Pr_{k \xleftarrow{\$} \{0,1\}^n}[\mathcal{A}^{F''(k,\cdot)}(1^n) = 1]$. Let

$\mathcal{D}^{U_n}$ be the output of the distinguisher $\mathcal{D}$ when the values it receives are sampled from U_n, we have $\Pr[\mathcal{D}^{U_n} = 1] = \Pr_{f \xleftarrow{\$} \mathsf{Func}}[\mathcal{A}^{f(\cdot)}(1^n) = 1]$. Because F is a PRF, for any PPT distinguisher $\mathcal{D}$, we have $|\Pr[\mathcal{D}^F = 1] - \Pr[\mathcal{D}^{U_n} = 1]| \leq \mathsf{negl}(n)$, and therefore

$$\left| \Pr_{k \xleftarrow{\$} \{0,1\}^n} [\mathcal{A}^{F''(k,\cdot)}(1^n) = 1] - \Pr_{f \xleftarrow{\$} \mathsf{Func}} [\mathcal{A}^{f(\cdot)}(1^n) = 1] \right| \leq \mathsf{negl}(n),$$

This contradicts our assumption, which means the $\mathcal{A}$ does not exist, hence F'' is a pseudorandom function.

(b) (PRG)

(i). G' is not a pseudorandom generator. To see this, we formally define the following attacker $\mathcal{A}$ given 1^n and access to some function g:

$\underline{\mathcal{A}^g(1^n):}$

- Receive y from the challenger.
- Output 1 if and only if the last bit of y is 1.

Now we have $\Pr_{x \xleftarrow{\$} \{0,1\}^n}[\mathcal{A} \text{ wins } | g \leftarrow G'(x)] = 1 - 2^{-n}$ when y is from G'. But when y is from uniform distribution U_{2n+1}, then y is independent, uniform string of length $2n+1$, and so the probability that the last bit is 1 is exactly $\frac{1}{2}$. Thus, $\Pr[\mathcal{A} \text{ wins } | g \leftarrow U_{2n+1}] = \frac{1}{2}$, and the probability

$$\begin{aligned} \Pr[\mathcal{A} \text{ wins }] &= \left| \frac{1}{2} \Pr_{x \xleftarrow{\$} \{0,1\}^n} [\mathcal{A} \text{ wins } | g \leftarrow G'(x)] + \frac{1}{2} \Pr[\mathcal{A} \text{ wins } | g \leftarrow U_{2n+1}] \right| \\ &= \frac{3}{4} - 2^{-(n+1)} \end{aligned}$$

is not negligible.

(ii). G'' is not a pseudorandom generator. To see this, we formally define the following attacker $\mathcal{A}$ given 1^n and access to some function g:

$\underline{\mathcal{A}^g(1^n):}$

- Receive y from the challenger.
- Output 1 if and only if $y = 0^{2n}$.

Now we have $\Pr_{x \xleftarrow{\$} \{0,1\}^n}[\mathcal{A} \text{ wins } | g \leftarrow G''(x)] = 1$ when y is from G''. But when y is from uniform distribution U_{2n}, then y is independent, uniform string of length $2n$, and so the probability that they are the same is exactly 2^{-2n}. Thus, $\Pr[\mathcal{A} \text{ wins } | g \leftarrow U_{2n}] = 2^{-2n}$, and the probability

$$\begin{aligned} \Pr[\mathcal{A} \text{ wins }] &= \left| \frac{1}{2} \Pr_{x \xleftarrow{\$} \{0,1\}^n} [\mathcal{A} \text{ wins } | g \leftarrow G''(x)] + \frac{1}{2} \Pr[\mathcal{A} \text{ wins } | g \leftarrow U_{2n}] \right| \\ &= \frac{1}{2} - 2^{-(2n+1)} \end{aligned}$$

is not negligible.

(iii). G''' is not a pseudorandom generator. To see this, suppose $G^\star$ is a PRG that $\{0,1\}^n \to \{0,1\}^{2n-1}$, we construct the following PRG G: G output $G^\star(x)||[x]^n_n$. In this case, because $G^\star$ is a PRG, and for $x \xleftarrow{\$} \{0,1\}^n$, $\Pr[[x]^n_n = 0] = \Pr[[x]^n_n = 1] = \frac{1}{2}$, G is still a PRG. we formally define the following attacker $\mathcal{A}$ given 1^n and access to some function g:

$\underline{\mathcal{A}^g(1^n):}$

- Receive y from the challenger.
- Output 1 if and only if the last bit of y is 1.

Now we have $\Pr_{x \xleftarrow{\$} \{0,1\}^n}[\mathcal{A} \text{ wins } | g \leftarrow G'''(x)] = 1 - 2^{-(n-1)}$ when y is from G'''. But when y is from uniform distribution U_{2n}, then y is independent, uniform string of length n, and so the probability that they are the same is exactly $\frac{1}{2}$. Thus, $\Pr[\mathcal{A} \text{ wins } | g \leftarrow U_{2n}] = \frac{1}{2}$, and the probability

$$\begin{aligned}\Pr[\mathcal{A} \text{ wins }] &= \left| \frac{1}{2} \Pr_{x \xleftarrow{\$} \{0,1\}^n}[\mathcal{A} \text{ wins } | g \leftarrow G'''(x)] + \frac{1}{2} \Pr[\mathcal{A} \text{ wins } | g \leftarrow U_{2n}] \right| \\ &= \frac{3}{4} - 2^{-n}\end{aligned}$$

is not negligible.

3. **(Hash Functions)**

   (a) H' is not collision resistant. Assume $F^\star$ is a PRF, we construct the following PRF F: If the key $k = 0^n$, F outputs 0^n, otherwise it behaves the same as $F^\star$. In this case, because $\Pr[[x]_1^n = 0^n] = \frac{1}{2^n}$, F is still a PRF. Consider x, x' such that $[x]_1^n = [x']_1^n = 0^n$ and $[x]_{n+1}^{2n} \neq [x']_{n+1}^{2n}$. We have $H'(x) = H'(x') = 0^n$.

   (b) H'' is not collision resistant. Consider x, x' such that they are different only in the last bit. I.e., $[x]_1^{2n-1} = [x']_1^{2n-1}$ and $[x]_{2n}^{2n} \neq [x']_{2n}^{2n}$. Then we have $H''(x) = H([x]_1^{2n-1}||0) = H([x']_1^{2n-1}||0) = H''(x')$.

4. **(Public key and secret key encryptions)**

   (a) **(ElGamal)** Correctness is trivial. The security follows via a simple reduction to the security of the ElGamal scheme we have seen at the lecture. The reduction works as follows. Let $\mathcal{A}$ denote the adversary attacking Π, we construct a new adversary $\mathcal{A}'$ that attacks the ElGamal scheme. This new adversary works as follows.

   Algorithm $\mathcal{A}'$:

   - Execute $\mathcal{A}$ on input the public key PK received from the challenger of the ElGamal encryption scheme.
   - When $\mathcal{A}$ sends (m_0, m_1), sample a random value a and compute $\tilde{m}_0 \leftarrow a - m_0$, $\tilde{m}_1 \leftarrow a - m_1$, and send $\tilde{m}_0, \tilde{m}_1$ to the challenger of ElGamal.
   - Upon receiving the ciphertext c from the challenger, send (c, a) to $\mathcal{A}$.
   - Return the choice bit of $\mathcal{A}$.

   The proof ends with the observation that when c corresponds to an encryption of $\tilde{m}_b$ then the output of $\mathcal{A}'$ corresponds to the output of $\mathcal{A}$ in input a ciphertext of m_0. Hence, if $\mathcal{A}$ breaks the security of Π with a non-negligible advantage, so does $\mathcal{A}'$ in breaking the security of ElGamal. But this is a contradiction.

   (b) We design a scheme $\Pi = (\mathsf{Gen'}, \mathsf{Enc'}, \mathsf{Dec'})$ that is CPA secure but not CCA secure. Below we describe how the algorithms of Π' work.

   - $\mathsf{Gen'}(1^n)$:Return $\mathsf{Gen}(1^n)$
   - $\mathsf{Enc'}(k, m)$:
     - $c \leftarrow \mathsf{Enc}(k, m)$

    - $r \leftarrow \{0,1\}^n$
    - Return c, r.
  - $\mathsf{Dec'}(k, c')$
    - parse c' as (c, r)
    - return $\mathsf{Dec}(k, c)$

  The new scheme is not CCA secure as the adversary can generate a new valid ciphertext starting from the challenge ciphertext $c' = (c, r)$, by simply changing the r component, and ask the decryption oracle to decrypt the new ciphertext.

  The scheme remains CPA secure, and this can be proven via a simple reduction to the CPA security of Π (recall that Π is assumed to be CCA secure, hence it must also be CPA secure). In more detail, assume towards contradiction that there exists an adversary $\mathcal{A}'$ attacking Π'. Any time the adversary submits an encryption query, the reduction (our adversary $\mathcal{A}$ attacking Π) forwards the encryption query to the encryption oracle of Π, and upon receiving the ciphertext c, samples a random value r, and sends c, r to $\mathcal{A}'$. When $\mathcal{A}'$ sends the challenge messages (m_0, m_1), $\mathcal{A}$ sends these messages to the challenger, and upon receiving the ciphertext c, it samples a new random value r, and sends (c, r) to $\mathcal{A}'$. At the end $\mathcal{A}$ returns the choice bit of $\mathcal{A}'$. The proof ends with the observation that if c is an encryption of m_b, so is the ciphertext (c, r). Hence, any advantage that $\mathcal{A}$ has in breaking Π', is translated in an advantage in breaking Π.

5. **(Digital signatures)**

   Sig' is not a secure digital signature scheme. We can construct the following adversary $\mathcal{A}$:

   $\underline{\mathcal{A}:}$

   - Receive pk from challenger $\mathcal{C}$.
   - Uniformly randomly sample message m, compute $m_0 \leftarrow [m]_1^{n-1}||0$, query the signature oracle m_0 and receive the signature θ.
   - Output $(m_1 \leftarrow [m]_1^{n-1}||1, \theta)$.

   In this case, (m_1, θ) is a valid signature because $[m_0]_1^{n-1} = [m_1]_1^{n-1}$, $\mathsf{Verify}(pk, m_1, \theta)$ will output 1. Also, $m_1 \neq m_0$, which means that m_1 is not in the set of queries the signature oracle receives.