Data Privacy Course Exercise 3

PB18061352 郑龙韬

Question 1.

You (Eve) have intercepted two ciphertexts:

$c_1 = 111110010111100111001100001011110000110$

$c_2 = 111011000111110111001110000110101000010$

You know that both are OTP ciphertexts, encrypted with the same key. You know that either c_1 is an encryption of "alpha" and c_2 is an encryption of "three" or c_1 is an encryption of "delta" and c_2 is an encryption of "sigma" (all converted to binary from ascii in the standard way). Which of these two possibilities is correct, and why? What was the key k?

Answer:

Let m_1 and m_2 be the plain text of c_1 and c_2, respectively. Since they are encrypted with the same key k, i.e., $c_1 = m_1 \oplus k$ and $c_2 = m_2 \oplus k$, it is apparent that $c_1 \oplus c_2 = m_1 \oplus m_2$.

We have

$$\text{"alpha"} = 01100001\ 01101100\ 01110000\ 01101000\ 01100001$$

$$\text{"three"} = 01110100\ 01101000\ 01110010\ 01100101\ 01100101$$

$$\text{"alpha"} \oplus \text{"three"} = 0001010100000100000000100000110100000100 = c_1 \oplus c_2$$

and

$$\text{"delta"} = 01100100\ 01100101\ 01101100\ 01110100\ 01100001$$

$$\text{"sigma"} = 01110011\ 01101001\ 01100111\ 01101101\ 01100001$$

$$\text{"delta"} \oplus \text{"sigma"} = 0001011100001100000010110001100100000000 \neq c_1 \oplus c_2$$

Therefore, c_1 is an encryption of "alpha" and c_2 is an encryption of "three", i.e., the first possibility is correct.

The key $k = c_2 \oplus m_2 = 100110000001010110111100011111111100111$, where m_2 represents "three".

Question 2.

Show that the following libraries are **not** interchangeable. Describe an explicit distinguishing calling program, and compute its output probabilities when linked to both libraries:

$\mathcal{L}_{\text{left}}$
$\underline{\text{EAVESDROP}(m_L, m_R \in \{0,1\}^\lambda):}$
$\quad k \leftarrow \{0,1\}^\lambda$
$\quad c := k \oplus m_L$
$\quad \text{return } (k, c)$

$\mathcal{L}_{\text{right}}$
$\underline{\text{EAVESDROP}(m_L, m_R \in \{0,1\}^\lambda):}$
$\quad k \leftarrow \{0,1\}^\lambda$
$\quad c := k \oplus m_R$
$\quad \text{return } (k, c)$

Answer:

Here are $\mathcal{L}_{left}$ and $\mathcal{L}_{right}$ calling program $\mathcal{A}$.

Algorithm 1 $\mathcal{A}$

1: $(k, c) \leftarrow \text{EAVESDROP}(0^\lambda, 1^\lambda)$
2: **return** $c \overset{?}{=} k$

We argue that

$$\Pr[\mathcal{A} \diamond \mathcal{L}_{left} \Rightarrow 1] = 1$$

$$\Pr[\mathcal{A} \diamond \mathcal{L}_{right} \Rightarrow 1] = 0$$

Thus, **not** for all programs $\mathcal{A}$ that output a single bit, there is $\Pr[\mathcal{A} \diamond \mathcal{L}_{left} \Rightarrow 1] = \Pr[\mathcal{A} \diamond \mathcal{L}_{right} \Rightarrow 1]$, i.e., $\mathcal{L}_{left} \not\equiv \mathcal{L}_{right}$. These two libraries are **not** interchangeable.

Question 3.

Which of the following are negligible functions in λ? Justify your answers.

$$\frac{1}{2^\lambda}, \frac{1}{2^{\log(\lambda^2)}}, \frac{1}{\lambda^{\log \lambda}}, \frac{1}{\lambda^2}, \frac{1}{2^{(\log \lambda)^2}}, \frac{1}{(\log \lambda)^2}, \frac{1}{\lambda^{1/\lambda}}, \frac{1}{\sqrt{\lambda}}, \frac{1}{2^{\sqrt{\lambda}}}$$

Answer:

A function $f(\lambda)$ negligible if, for every polynomial function p, we have $\lim_{\lambda \to \infty} p(\lambda) f(\lambda) = 0$.

Let $p(\lambda) = \lambda^a$, where a is a large number. Consider all given functions:

$$\lim_{\lambda \to \infty} p(\lambda) \frac{1}{2^\lambda} = 0$$

$$\lim_{\lambda \to \infty} p(\lambda) \frac{1}{2^{\log(\lambda^2)}} = \lim_{\lambda \to \infty} p(\lambda) \frac{1}{2^{2\log \lambda}} = \lim_{\lambda \to \infty} \lambda^{a - \log 4} = \infty$$

$$\lim_{\lambda \to \infty} p(\lambda) \frac{1}{\lambda^{\log \lambda}} = \lim_{\lambda \to \infty} \lambda^{a - \log \lambda} = 0$$

$$\lim_{\lambda \to \infty} p(\lambda) \frac{1}{\lambda^2} = \infty$$

$$\lim_{\lambda \to \infty} p(\lambda) \frac{1}{2^{(\log \lambda)^2}} = 0$$

$$\lim_{\lambda \to \infty} p(\lambda) \frac{1}{(\log \lambda)^2} = \infty$$

$$\lim_{\lambda \to \infty} p(\lambda) \frac{1}{\lambda^{1/\lambda}} = \lim_{\lambda \to \infty} \lambda^{a - 1/\lambda} = \infty$$

$$\lim_{\lambda \to \infty} p(\lambda) \frac{1}{\sqrt{\lambda}} = \lim_{\lambda \to \infty} \lambda^{a - \frac{1}{2}} = \infty$$

$$\lim_{\lambda \to \infty} p(\lambda) \frac{1}{2^{\sqrt{\lambda}}} = 0$$

Thus, $\frac{1}{2^\lambda}, \frac{1}{\lambda^{\log \lambda}}, \frac{1}{2^{(\log \lambda)^2}}, \frac{1}{2^{\sqrt{\lambda}}}$ are negligible, while other functions are not.

Question 4.

$\mathcal{A}$
$x := \text{QUERY}()$ for all $s' \in \{0, 1\}^\lambda$:   if $G(s') = x$ then return 1   return 0

$\mathcal{L}^G_{\text{prg-real}}$	$\mathcal{L}^G_{\text{prg-rand}}$
$\underline{\text{QUERY}()}:$ $s \leftarrow \{0, 1\}^\lambda$ return $G(s)$	$\underline{\text{QUERY}()}:$ $r \leftarrow \{0, 1\}^{\lambda+\ell}$ return r

Let $G : \{0, 1\}^\lambda \to \{0, 1\}^{\lambda+\ell}$ be an injective (i.e., 1-to-1) PRG. Consider the following distinguisher:

(a) What is the advantage of $\mathcal{A}$ in distinguishing $\mathcal{L}^G_{\text{prg-real}}$ and $\mathcal{L}^G_{\text{prg-rand}}$? Is it negligible?

Answer:

$$\Pr[\mathcal{A} \diamond \mathcal{L}^G_{\text{prg-real}} \Rightarrow 1] = 1$$

$$\Pr[\mathcal{A} \diamond \mathcal{L}_{\text{prg-rand}}^{G} \Rightarrow 1] = \frac{2^{\lambda}}{2^{\lambda+\ell}} = \frac{1}{2^{\ell}}$$

Thus, the advantage of this adversary is $1 - \frac{1}{2^{\ell}}$, which is non-negligible.

(b) Does this contradict the fact that G is a PRG? Why or why not?

Answer:

No. Traversing all outputs of G in adversary $\mathcal{A}$ takes exponential time $O(2^{\lambda})$. The encryption is actually secure against polynomial time adversaries. Thus, $G(k)$ is "close enough" to uniform, i.e., there is no polynomial time algorithm can distinguish the distribution of $G(k)$ values from the uniform distribution.

Question 5.

Assume that Bob uses RSA and selects two "large" prime numbers $p = 101$ and $q = 103$.

(a) How many possible public keys from which Bob can choose?

Answer:

$\Phi(n) = (p-1)(q-1) = 100 \cdot 102 = 10200$.

$\forall x \in (1, 10200)$, there are 2559 numbers that satisfy $x \perp 10200$ (co-prime).

Therefore, there are 2559 public keys to choose.

(b) Assume also that Bob uses a public encryption key $e = 71$. Alice sends Bob a message $M = 2021$. What will be the ciphertext received by Bob?

Answer:

$n = pq = 10403$

$PU = \{e, n\} = \{71, 10403\}$

$C = M^e \mod n = 2021^{71} \mod 10403 = 10000$

Thus, the ciphertext is 10000.

(c) Show the detailed procedure that Bob decrypts the received ciphertext.

Answer:

The procedure is as follows:

Since $e = 71$, we select $d = 431$ on the grounds that $10200 + 1 = 71 \cdot 431$.

$M = C^d \mod n = 10000^{431} \mod 10403 = 2021$.

Question 6.

Let $N = pq$ be a product of two distinct primes. Show that if $\Phi(N)$ and N are known, then it is possible to compute p and q in polynomial time. (Hint: Derive a quadratic equation (over the integers) in the unknown p.)

Answer:

Since p and q are two distinct primes, we have

$$\begin{aligned}
\Phi(n) &= (p-1)(q-1) \\
&= pq - p - q + 1 \\
&= n - p - q + 1 \\
&= n - p - \frac{n}{p} + 1
\end{aligned}$$

Derive a quadratic equation (over the integers) in the unknown p

$$p^2 + (\Phi(n) - n - 1)p + n = 0$$

By this equation, p can be solved in polynomial time, and $q = \frac{n}{p}$.

Thus, if $\Phi(N)$ and N are known, then it is possible to compute p and q in polynomial time.