به نام کیمیاگر عالم

دانشگاه صنعتی امیرکبیر
(پلی تکنیک تهران)

امنیت شبکه

عنوان

تمرین سوم – رمزنگاری

مدرس

دکتر سیاوش خرسندی

دانشجو

امیرحسین بابائیان

۴۰۱۱۳۱۰۰۲

ترم پاییز ۰۲-۰۱

گروه معماری کامپیوتر و شبکه های کامپیوتری

دانشکده مهندسی کامپیوتر، دانشگاه صنعتی امیرکبیر (پلی تکنیک تهران)

فهرست

سوال ۱

بخش a

هر شماره نماینده ۱ تا ۸ به صورت ترتیبی چیده شده است.

1	2	3	4	5	6	7	8
i	t	h	o	u	g	h	t
t	o	s	e	e	t	h	e
f	a	i	r	i	e	s	i
n	t	h	e	f	i	e	l
d	s	b	u	t	i	s	a
w	o	n	l	y	t	h	e
e	v	i	l	e	l	e	p
h	a	n	t	s	w	i	t
h	t	h	e	i	r	b	l
	.	.	.	.	.	.	
	.					.	
	.					.	
	.					.	
e	a	t	h	w	a	i	t
o	n	t	h	y	e	n	d

رمز :

He sitteth between the cherubims The isles may be glad thereof As the rivers in the south

بخش b

تا حد بسیار زیادی امن است، زیرا در هر ۸ حرف صرفا یک حرف انتخاب میشود که در صورت متمایز بودن تمام حروف در یک سطر ۸ حالت مختلف به ازای هر سطر داریم که از این جهت حالات کل برابر با ۸ به توان تعداد سطر ها می شود.

بخش C

امن محسوب نمی شود، با داشتن کلید میتوان حدس زدن که عدد ۸ ماکس یک چیزی است و اعداد بین یک تا ۸ می چرخند پس ممکن است انتخاب شماره حروف سطر باشد که الگوریتم لو رفته است.

سوال ۲

بخش a

```
str_in = '53bba305))6*;4826)4b.)4b);806*;48a8¶60))85;;]8*;:b*8a83(88)5*a;46(;88*9
6*?;8)*b(;485);5*a2:*b(;4956*2(5*-
4)88*;4069285);)6a8)4bb;1(b9;48081;8:8b1;48a85;4)485a528806*81(b9;48;(88;4(b?34;4
8)4b;161;:188;b?;'

frequency_counter = {}

for ch in str_in:
    if frequency_counter.get(ch, True):
        frequency_counter[ch]= str_in.count(ch)

    continue

print(frequency_counter)
```

قطعه کد فوق برای شمارش تعداد تکرار هر کاراکتر است، بجای کاراکتر های غیرقابل درج از a و b استفاده شده است.

خروجی کد:

```
{'5': 12, '3': 4, 'b': 15, 'a': 8, '0': 6, ')': 16, '6': 11, '*': 14, ';': 27, '4': 19, '8': 34, '2': 5, '.': 1,
 '¶': 1, ']': 1, ':': 4, '(': 9, '9': 5, '?': 3, '-': 1, '1': 7}
```

به ترتیب سه کاراکتری که بیشترین تکرار را دارد عبارتند از : ۸ و ; و (که به ترتیب تعداد فراوانی های ۳۴ و ۲۷ و ۱۶ دارند.

انجام جایگذاری E بجای 8:

```
string after replace e and 8:
53bba305))6*;4E26)4b.)4b);E06*;4EaE¶60))E5;;]E*;:b*EaE3(EE)5*a;46
(;EE*96*?;E)*b(;4E5);5*a2:*b(;4956*2(5*-4)EE*;40692E5);)6aE)4bb;1
(b9;4E0E1;E:Eb1;4EaE5;4)4E5a52EE06*E1(b9;4E;(EE;4(b?34;4E)4b;161;
:1EE;b?;
```

بخش b

با توجه به تکرار عبارت 4E; پیش بینی می شود که باید کلمه ی پرتکرار the باشد پس جایگذاری میکنیم:

```
 string after replace ;4E and THE:
53bba305))6*THE26)4b.)4b);E06*THEaE¶60))E5;;]E*;:b*EaE3(EE)5*a;46
(;EE*96*?;E)*b(THE5);5*a2:*b(;4956*2(5*-4)EE*;40692E5);)6aE)4bb;1
(b9THE0E1;E:Eb1THEaE5;4)4E5a52EE06*E1(b9THE;(EE;4(b?34THE)4b;161;
:1EE;b?;
```

بخش C

با استفاده از جدول بدست امده و فرکانس ها:

```
agoodglassinTHEbishopshostelinTHEdevilsseattwentyonedegreesandthi
rteenminutesnorTHEastandbynorthmainbranchseventhlimbeastsideshoot
fromTHElefteyeofTHEdeathsheadabeelinefromTHEtreethroughTHEshotfif
tyfeetout
```

جمله رمز نگاری شده:

```
a good glass in THE bishops hostel in THE devils seat twenty one degrees and
thirteen minutes nor THE a standby north main branch seven thlimbeast side
shoot from THE left eye of THE deaths head abee line from THE tree through THE
shot fifty feet out
```

کلماتی که آبی رنگ شده اند غیر قابل فهم بودند.

سوال ۳

بخش a

نیک میدانیم که که مکمل بر روی xor پخش می شود و اگر به صورت بیتی بررسی کنیم به صورت ذیل جدول درستی را خواهیم داشت:

a	b	Com(a XOR b)	Com(a) XOR Com(b)
0	0	1	1
0	1	0	0
1	0	0	0
1	1	1	1

برابری با جدول درستی اثبات شد.

چون XOR به صورت بلوک بلوک و نظیر به نظیر بیت ها اتفاق می افتد پس اگر بیاییم و عبارت را مکمل کنیم و کلید را نیز مکمل کنیم، حاصل در نهایت برابر می شود با مکمل XOR کلید و عبارت.

بخش b

به دقت پاسخ را نمیدانم اما حس میکنم با رابطه ی فوق نیازی به تست شدن کل کلید های ۲ به توان ۵۶ تایی نمی باشد و با تست هر کلید و مکمل کردن همان کلید می توان تست انجام شود، از این رو کلید ها نصف می شود.

سوال ۴

در این سوال برای رد هر کدام لازم است یک مثال نقض آورده شود.

فرض میشود توابع درهم ساز خروجی صفر ندارند.

بخش ۱

$$h(x) = f(x) \circ g(x)$$

در این بخش فرقی ندارد فرض کنیم کدام یک Collision Resistant است چرا که حاصلضرب هر دو در یک دیگر است پس فرض میکنیم که f، CR است و احتمال رخ داد Collision در g نرخ بسیار بالایی ندارد.

با توجه به فرض فوق با هر ورودی متفاوتی که به f بدهیم یک خروجی متفاوت به ما می دهد از این رو ضرب آن در عبارتی که g میدهد حتی اگر Collision رخ دهد هم باز حاصل متفاوتی به ما میدهد پس خروجی

بخش ۲

$$h(x) = f(g(x))$$

با فرض بخش ۱ اگر پیش برویم، g ممکن است گاهی مقدار های یکسانی برای x های متفاوت بدهد و با توجه به اینکه f یک ورودی ثابت میگیرد که f(g(x)) است پس خروجی یکسانی میدهد، از این رو با ۲ مقدار متفاوت به یک مقدار ثابت می رسیم که اینجا Collision Resistant نیست.

فرض میکنیم g(2)=g(3)=20 و f(20)=100 است، حال مقدار h(2)=f(g(2))=100 و h(3)=f(g(3))=100.

حال اگر فرض را جابجا کنیم و f و g به ترتیب غیرمقاوم در برابر تصادم و مقاوم باشند انگاه حالت ذیل اتفاق می افتد که باز هم Collision Resistant نیست.

به عنوان مثال نقض فرض میکنیم که g(2)=10، g(3)=20، f(20)=f(10)=99 است انگاه مقدار h(2)=f(g(2))=f(10)=99 و h(3)=f(g(3))=f(20)=99 پس با دو مقدار متفاوت یک مقدار یکسان حاصل شد و اینجاست که CR نیست.

بخش ۳

$$h(x) = f(g(x)) \circ g(f(x))$$

بخش سوم ترکیب بین دو حالت قبلی است که بر اساس آن دیدیم که یکی از عبارات طرفین ضرب CR است و دیگری نیست اما طبق بخش اول فهمیدیم که حاصل ضرب دو تابع با فرض گفته شده CR است.

سوال ۵

الف

جدول پر شده ی M3 به صورت ذیل است:

۵	۲	۱	۴	۵
۱	۴	۳	۲	۲
۳	۱	۲	۵	۳
۴	۳	۴	۱	۴
۲	۵	۵	۳	۱

ب

بین دو کلاینت اگر قرار است پیامی جابجا شود به صورت ذیل انجام می شود:

طرف اول یک عدد رندوم به عنوان کلید خصوصی بدست می آورد و آن را به M1 مپ میکند تا x را بدست آورد

طرف اول x را به عنوان کلید عمومی به طرف دوم میدهد.

طرف دوم با استفاده از x پیام متنی را با M2 رمز میکند تا z بدست بیاید، سپس پیام را به طرف اول ارسال میکند.

طرف اول با دریافت پیام و با استفاده از k رمزگشایی میکند z را به کمک M3 و پیام اصلی را بدست می آورد.

ج

بستگی دارد به بزرگی اعداد انتخابی و تصادفی بودن M1 و M2.

در این حالت به راحتی نمیتوان به متن پیام بدون کلید خصوصی دست پیدا کرد.

سوال ۶

بخش a

راه حل موجود در اینترنت :

Let –X be the additive inverse of X. That is –X $\boxed{+}$ X = 0. Then:

$$P = (C \boxed{+} -K_1) \oplus K_0$$

بخش b

First, calculate –C'. Then –C' = (P' $\oplus$ K_0) $\boxed{+}$ (–K_1). We then have:

C $\boxed{+}$ –C' = (P $\oplus$ K_0) $\boxed{+}$ (P' $\oplus$ K_0)

However, the operations $\boxed{+}$ and $\oplus$ are not associative or distributive with one another, so it is not possible to solve this equation for K_0.

سوال ۷

سوال را نفهمیدم، فهم سوال نیمی از پاسخ است، از این رو نتوانستم پاسخی برای سوال بنویسم.

آپدیت پیش از ارسال : پاسخ را با جستجو یافتم اما فایده ای نداشت.

For $1 \le i \le 128$, take $c_i \in \{0, 1\}^{128}$ to be the string containing a 1 in position i and then zeros elsewhere. Obtain the decryption of these 128 ciphertexts. Let m_1, $m_2, \ldots, m_{128}$ be the corresponding plaintexts. Now, given any ciphertext c which does not consist of all zeros, there is a unique nonempty subset of the c_i's which we can XOR together to obtain c. Let $I(c) \subseteq \{1, 2, \ldots, 128\}$ denote this subset. Observe

$$c = \bigoplus_{i \in I(c)} c_i = \bigoplus_{i \in I(c)} E(m_i) = E\left(\bigoplus_{i \in I(c)} m_i \right)$$

Thus, we obtain the plaintext of c by computing $\bigoplus_{i \in I(c)} m_i$. Let **0** be the all-zero string. Note that $\mathbf{0} = \mathbf{0} \oplus \mathbf{0}$. From this we obtain $E(\mathbf{0}) = E(\mathbf{0} \oplus \mathbf{0}) = E(\mathbf{0}) \oplus E(\mathbf{0}) = \mathbf{0}$. Thus, the plaintext of $c = \mathbf{0}$ is $m = \mathbf{0}$. Hence we can decrypt every $c \in \{0, 1\}^{128}$.

سوال ۸

پاسخ سوال را بلد نبودم.

آپدیت پیش از ارسال : پاسخ را یافتم اما اثری نداشت.

The opponent has the two-block message B1, B2 and its hash RSAH(B1, B2). The following attack will work. Choose an arbitrary C1 and choose C2 such that:

$$C2 = RSA(C1) \oplus RSA(B1) \oplus B2$$

then

$$RSA(C1) \oplus C2 = RSA(C1) \oplus RSA(C1) \oplus RSA(B1) \oplus B2$$

$$= RSA(B1) \oplus B2$$

so

$$RSAH(C1, C2) = RSA[RSA(C1) \oplus C2)] = RSA[RSA(B1) \oplus B2]$$
$$= RSAH(B1, B2)$$

سوال ۹

مفروضات سوال عبارتند از : q=11 و alpha=2

بخش a

اگر Ya برابر با ۹ باشد آنگاه مقدار Xa ؟

Ya = alpha ^ Xa mod q

9 = 2 ^ Xa mod 11

با جایگذاری می توانیم به جواب Xa برابر است با ۶ برسیم.

بخش b

اگر Yb برابر با ۳ باشد آنگاه مقدار k ؟

K = Yb ^ Xa mod q

K = 3^6 mod 11

K = 3

سوال ۱۰

Usage 1	Usage 2	Usage 3
(1) A → B: N_a (2) B → A: E(K, N_a)	(1) A → B: E(K, N_a) (2) B → A: N_a	(1) A → B: E(K, N_a) (2) B → A: E(K, f(N_a))

مشکل روش اول و دوم در reply حمله است که دومی باز هم به نسبت اولی قابل اتکا تر است، توضیحات:

در حالت اول N به صورت عادی مبادله می شود که حمله کننده می تواند N های مختلفی را ارسال کند و حمله ی reply را بعدا رقم بزند، در حالت دوم هم که با کلید مشترک بین این N رمز می شود هم حمله کننده می توانند شروع به حدس N نماید چرا که مقدار N ثابت می ماند و تغییر نمیکند، اما در روش سوم چون ابتدا عملیات رمز کردن اتفاق می افتد و سپس مقدار N افزایش می یابد آنگاه دیگر امکان حمله reply تا حد زیادی گرفته می شود.

سوال ۱۱

این مشکل یک راه حل ساده دارد، یعنی وارد کردن نام B در اطلاعات امضا شده برای پیام سوم، به طوری که اکنون پیام سوم می خواند:

$$A ===> B: \quad A\{r_B, B\}$$

سوال ۱۲

بخش a

A، آی دی خود را برای B می فرستد، B مقدار R1 را با کلید عمومی A رمز میکند و برای A می فرستد، پس صرفا A می تواند R1 را با کلید خصوصی خودش باز کند، حال محتوای ارسال شده از A صرفا با استفاده کلید عمومی اش قابل باز شدن است، به نوعی امضای خود را با کلید خصوصی اعمال می کند و برای B می فرستد.

بخش b

یک نفر سوم می تواند در وسط ماجرا حاضر شود و از A بخواهد تا پیامی را امضا کند، حال می تواند این امضا را همراه با پیامی برای B بفرستد، این حس را نشان میدهد که A برای B پیام فرستاده است