به نام خدا

دانشکده‌ی علوم ریاضی

مقدمه‌ای بر رمزنگاری	دانشجو: علیرضا توفیقی محمدی
تمرین : سری ۳	
مدرّس: دکتر شهرام خزائی	شماره‌ی دانشجویی: ۹۶۱۰۰۳۶۳

مسأله‌ی ۱

آ)

PRFنیست چراکه حمله کننده‌ای را در نظر بگیرید که $F'_k(0^n)$ را از اراکل محاسبه کرده و اگر 0^n بود یک و در غیر اینصورت صفر می‌دهد. مزیت این حمله کننده برابر با:

$$Adv = \left| Pr[k \leftarrow \{0,1\}^n, A^{F'_k(\cdot)}(1^n) = 1] - Pr[f \leftarrow (\{0,1\}^n)^{(\{0,1\}^n)}, A^{f(\cdot)}(1^n) = 1] \right|$$

$$= |1 - 2^{-n}| = 1 - 2^{-n}$$

که غیر ناچیز است.

ب)

PRFنیست چراکه حمله کننده‌ای را در نظر بگیرید که $F(F'_k(0^n), 1^n)$ و $F'_k(1^n)$ را از اراکل محاسبه کرده و اگر برابر بودند یک و در غیر اینصورت صفر می‌دهد. مزیت این حمله کننده برابر با:

$$Adv = \left| Pr[k \leftarrow \{0,1\}^n, A^{F'_k(\cdot)}(1^n) = 1] - Pr[f \leftarrow (\{0,1\}^n)^{(\{0,1\}^n)}, A^{f(\cdot)}(1^n) = 1] \right|$$

$$= |1 - 2^{-n}| = 1 - 2^{-n}$$

که غیر ناچیز است.

ج)

امن است، فرض کنید چنین نباشد و حمله کننده‌ی A با مزیت غیرناچیز برای حمله به F' داشته باشیم، با کمک A حمله کننده‌ای برای F می‌سازیم، به این صورت که با گرفتن اوراکل $f(\cdot)$ از چالشگر، k تصادفی از $\{0,1\}^n$ ساخته و سپس $F_k \oplus f$ را به حمله کننده‌ی A می‌دهیم. اگر f یک تابع کاملا تصادفی باشد، XOR آن را هر تابعی نیز یک تابع کاملا تصادفی است و در نتیجه $F_k \oplus f$ نیز کاملا تصادفی است، اگر f برابر با $F_{k'}$ برای یک k' تصادفی باشد، $F_k \oplus f$ برابر با $F'_{k||k'}$ خواهد بود که در نتیجه مزیت حمله کننده‌ای که ساختیم برابر با مزیت حمله کننده‌ی A است، که غیر ناچیز بود، پس اگر F' PRFنباشد F نیز PRFنیست که تناقض است، پس فرض خلف باطل و F' PRFاست.

د)

هست چراکه طبق لم هیبرید داریم و PRFبودن F داریم:

$$F_{U_n} \equiv_C U_n^{U_n} \implies \text{reverse}(F_{U_n}) \equiv_C \text{reverse}(U_n^{U_n}) \equiv U_n^{U_n} \implies \text{reverse}(F_{U_n}) \equiv U_n^{U_n}$$

(منظورم از $U_n^{U_n}$ تابعی تصادفی است.)

ه)

هست چراکه طبق لم هیبرید داریم و PRFبودن F داریم:

$$F_{U_n} \equiv_C U_n^{U_n} \implies 1^n \oplus F_{U_n} \equiv_C 1^n \oplus \text{reverse}(U_n^{U_n}) \equiv U_n^{U_n} \implies 1^n \oplus F_{U_n} \equiv U_n^{U_n}$$

(منظورم از $U_n^{U_n}$ تابعی تصادفی است.)

و)

نیست چراکه حمله کننده‌ای را در نظر بگیرید که با کمک اوراکل $F'_k(1^n) \oplus F'_k(0^n)$ را محاسبه می‌کند، اگر 1^n بود یک و در غیر اینصورت صفر می‌دهد. مزیت این حمله کننده برابر است با:

$$Adv = \left| Pr[k \leftarrow \{0,1\}^n, A^{F'_k(\cdot)}(1^n) = 1] - Pr[f \leftarrow (\{0,1\}^n)^{(\{0,1\}^n)}, A^{f(\cdot)}(1^n) = 1] \right|$$

$$= |1 - 2^{-n}| = 1 - 2^{-n}$$

که غیر ناچیز است.

مسأله‌ی ۲

آ)

آزمایش $\mathrm{MacForge}_{A_\Pi}$ را به شکل زیر تعریف می‌کنیم:

۱. چالشگر یک کلید k تولید می‌کند:
$$k \leftarrow Gen(1^n)$$

۲. به مهاجم دسترسی اورکلی به $\mathrm{Mac}_k(\cdot)$ و $\mathrm{Vrfy}_k(\cdot)$ داده می‌شود و در نهایت مهاجم پس‌از پرسمان های لازم× یک زوج $\langle m, t\rangle$ تولید می‌کند:

$$\langle m, t\rangle A^{\mathrm{Mac}_k(\cdot),\mathrm{Vrfy}_k(\cdot)}(1^n)$$

مجموعه پرسمان‌هایی که مهاجم به $\mathrm{Mac}_k(\cdot)$ کرده است را Q می‌نامیم. خروجی آزمایش را به شکل زیر تعریف می‌کنیم:
$$\mathrm{MaxForge}_{A,\Pi}(n) = \begin{cases} 1, \text{if } m \notin Q \wedge \mathrm{Vrfy}_k(\langle m, t\rangle) = 1 \\ 0, \text{otherwise} \end{cases}$$
سیستم رمز Π دارای امنیتی که در صورت سوال گفته‌شده است اگر و تنها اگر برای هر مهاجم A که در آزمایش بالا شرکت کرده باشد تابع ناچیز $\epsilon(\cdot)$ وجود داشته‌باشد:

$$\forall n : Pr[\mathrm{MaxForge}_{A,\Pi}(n) = 1] \leq \epsilon(n)$$

ب)

با توجه به اینکه طبق این تعریف داریم:

$$\mathrm{Vrfy}_k(m, t) = \Big\{ 1, \text{if } \mathrm{Mac}_k(m) = t 0, \text{otherwise}$$

پس اگر اوراکل $\mathrm{Mac}_k(\cdot)$ را داشته باشیم، اوراکل Vrfy_k نیز از آن به دست می‌آید. پس اگر مهاجمی طبق تعریف جزوه امن باشد، آنگاه امنیت طبق تعریف بخش اول را نیز دارد زیرا داشتن اوراکل وریفای چیزی به آن اضافه نمی‌کند.

مسأله‌ی ۳

آ)

ساده‌است چرا که کافی است پارامتر دوم را بر پارامتر اول از طریق modular inverse تقسیم کنیم.

ب)

سخت است، چراکه محاسبه‌ی $\sqrt{x}$ از روی x مسئله‌ی سخت نیست، پس اگر بتوانیم $f(g^x, g^y)$ را محاسبه کنیم، قادر به محاسبه‌ی $\sqrt{f(g^x, g^y)} = g^{xy}$ نیز هستیم.

ج)

سخت است، چراکه محاسبه‌ی x^2 از روی x مسئله‌ی سخت نیست، پس اگر بتوانیم $f(g^x, g^y)$ را محاسبه کنیم، قادر به محاسبه‌ی $f(g^x, g^y)^2 = g^{xy}$ نیز هستیم.

د)

ساده‌است، جراکه کافی است دو مولفه‌ی تابع را در هم ضرب کنیم.

مسأله‌ی ۴

آ)

فرض کنید امن نباشد، یعنی حمله‌کننده‌ای مثل A برای Mac' وجود دارد که با احتمال غیرناچیز جعل می‌کند، کافی‌است حمله کننده‌ی A' را بسازیم که دقیقا مثل حمله کننده‌ی A عمل می‌کند با این تفاوت که همه‌ی پیام‌هایی که به اراکل می‌دهد و پیام خروجی‌اش را دوبار پشت سر هم پیامی که A می‌داد می‌گذارد. در این صورت A' با همان احتمال Mac را جعل می‌کند.

ب)

امن نیست چراکه حمله کننده‌ی A می‌تواند بدون استفاده از اوراکل $(0^n, 0^128)$ را بیرون دهد که به احتمال یک پیروز می‌شود.

ج)

امین است، فرض کنید امن نباشد، یعنی حمله‌کننده‌ای مثل A برای Mac' وجود دارد که با احتمال غیرناچیز جعل می‌کند، کافی‌است حمله کننده‌ی A' را بسازیم که دقیقا مثل حمله کننده‌ی A عمل می‌کند با این تفاوت که همه‌ی

جواب‌هایی که از اوراکل می‌گیرید و t خروجی‌اش را به شکل زوج مرتب $\langle t, t\rangle$ در می‌آورد. در این صورت A' با همان احتمال Mac را جعل می‌کند و به احتمال یک جعل کند.

د)

امن نیست، چراکه حمله کننده‌ی A را می‌سازیم که پیام 1^n را به Mac'_k بدهد و خروجی $\langle a, b\rangle$ را دریافت کند، حال چون $b = \mathrm{Mac}_k(0^n)$ است، پس $\mathrm{Mac}'_k(0^n) = \langle b, b\rangle$. پس کافی است 0^n و $\langle b, b\rangle$ را خروجی دهد و به احتمال یک جعل کند.

ه)

امن نیست، چراکه کافی است حمله کننده A پیام 0^n1^n را بپرسد و خروجی $\langle a, b\rangle$ را دریافت کند و سپس پیام 0^n0^n با مک $\langle a, a\rangle$ را خروجی دهد و به احتمال یک جعل کند.

و)

امن نیست، چراکه کافی است حمله کننده‌ی A پیام 0^n را بپرسد و خروجی a را دریافت کند، سپس پیام 1^n با مک a را خروجی دهد و به احتمال یک جعل کند.

مسأله‌ی ۵

آ)

امن نیست، چون Π دربرابر CCA امن است، پس احتمال اینکه $\mathrm{Enc}_p k(m)$ را دوبار اجرا کنیم و یک جواب بدهد ناچیز است. حال حمله کننده‌ی A دو پیام تصادفی m_0, m_1 را تولید کرده و به چالشگر می‌دهد، چالشگر بیت تصادفی b از صفر و یک را انتخاب کرده و $\mathrm{Enc}_p k'(m_b) = (a, b)$ را به A می‌دهد. طبق جمله‌ی اول احتمال اینکه $a = b$ باشد ناچیز است، حال حمله کننده رمز (a, a) را به اوراکل رمزگشایی می‌دهد و m_b را پیدا می‌کند و به احتمال یک منهای ناچیز درست حدس می‌زند.

ب)

من نیست، چون Π دربرابر CCA امن است، پس احتمال اینکه $\mathrm{Enc}_p k(m)$ را دوبار اجرا کنیم و یک جواب بدهد ناچیز است. حال حمله کننده‌ی A دو پیام $m_0 = 0^n, m_1 = 1^n$ را به چالشگر می‌دهد و چالشگر $\mathrm{Enc}_p k'(m_b) = (a, b)$ را حمله کننده می‌دهد، طبق جمله‌ی اول احتمال اینکه $a = b$ باشد از یک تابع ناچیز کمتر است، حال حمله کننده کافی است (b, b) را به اوراکل رمزگشایی بدهد، اگر بات داد یعنی پیام 1^n است و در غیر اینصورت پیام 0^n است و به احتمال یک منهای ناچیز که غیرناچیز است تمایز می‌دهد.

ج)

امن است، فرض کنید امن نباشد و یک تمایزگر با احتمال غیرناچیز برای Π' داشته باشیم، این حمله کننده را A بنامید، حال حمله کننده‌ی A' را می‌سازیم که از حمله کننده‌ی A استفاده می‌کند، هر درخواستی که A از رمزنگاری کرد، A' از Enc_pk می‌کند و اگر c را گرفت، (c, c) را به A می‌دهد و هر درخواستی مثل (c, c) که A از اوراکل رمزگشایی کرد، A' نیز c را از Dec_sk درخواست می‌کند. سپس دوپیامی که A برای تمایز داد را می‌دهد و دقیقا جواب A به آن را برمی‌گرداند.

د)

امن است، فرض کنید امن نباشد و یک تمایزگر با احتمال غیرناچیز برای Π' داشته باشیم، این حمله کننده را A بنامید، حال حمله کننده‌ی A' را می‌سازیم که از حمله کننده‌ی A استفاده می‌کند، هر درخواستی مثل m که A از رمزنگاری کرد ، A' $\mathrm{Enc}_pk(m \oplus 1^n)$ را به A می‌دهد و هر درخواستی مثل c که A از اوراکل رمزگشایی کرد، A' نیز c را از Dec_sk درخواست می‌کند و اگر m را گرفت، $m \oplus 1^n$ را به A می‌دهد. سپس دوپیامی که A برای تمایز داد را با 1^n ایکس‌اور می‌کند و به چالشگر می‌دهد و دقیقا جواب A به آن را برمی‌گرداند.

مسأله‌ی ۶

باید برای هر حمله کننده مثل A که PPT است، تابع ناچیز $\epsilon(\cdot)$ وجود داشته‌باشد که:

$$\forall n : Pr[(m, t) \leftarrow A^{\mathsf{Enc}_k(\cdot)}(1^n); m \notin Q \wedge \mathsf{Dec}_k(t) = m] \leq \epsilon(n)$$

که Q مجموعه‌ی همه‌ی درخواست‌هایی است که حمله کننده از Enc_k کرده‌است.

مسأله‌ی ۷

خیر چراکه کافی است حمله کننده مقدار $F_k(0^{n-1}1) \oplus F_k(0^{n-1}0)$ و $F_k(0^{n-2}11) \oplus F_k(0^{n-2}10)$ را محاسبه کند، اگر این دو برابر بودند حمله کننده یک و در غیر اینصورت ۰ می‌دهد، اگر $F_k(\cdot)$ یک تابع تصادفی باشد احتمال یک‌دادن حمله کننده برابر با 2^{-n} است و اگر از روش گفته‌شده در سوال ساخته شده‌باشد احتمال یک دادن حمله کننده یک است، پس مزیت حمله کننده $1 - 2^{-n}$ است که غیرناچیز است. پس PRF نیست.

مسأله‌ی ۸

کافی است آلیس مقدار g^a/b $(g^{1/b})^a =$ را محاسبه کرده و باب نیز مقدار g^a/b $(g^a)^{1/b} =$ را محاسبه کند، رمز مشترکشان $g^{a/b}$ می‌شود.

مسأله‌ی ۹

کافی است حمله کننده دو پیام تصادفی m_0, m_1 را انتخاب کرده و به چالشگر بدهد، حال چالشگر بیت تصادفی b را انتخاب می‌کند و مقدار $\mathsf{Enc}_{pk}(m_b) = c$ را به حمله‌کننده می‌دهد. حال کافی است حمله کننده مقدار $m = \mathsf{Dec}_{sk}(2.c)$ را محاسبه کند، اگر این مقدار برابر با $2m_0$ بود بیت ۰ و در غیراینصورت ۱ را خروجی دهد و با مزیت یک تمایز دهد.

مسأله‌ی ۱۰

خیر، چراکه برای امنیت داشتن برای حمله‌ی CPA سیستم‌های رمز public-key لازم است که آن‌ها تصادفی باشند، پس از پیام باید حداقل دو رمز داشته‌باشد، اگر طول فضای پیام و فضای رمز برابر باشد، این امر باعث می‌دهد که دو پیام یک رمز را بدهند که این باعث نقض شرط صحت می‌شود.

مسأله‌ی ۱۱

کافی است حمله کننده‌ی A، پیام 0^n را به Mac_k و (r, t) را دریافت کند، سپس یک پیام m پیدا کند که $CRC32(m) \neq 0$ و سپس مقدار $(m, (r, t \oplus CRC32(m)))$ را خروجی دهد. چون CRC32 عمل ایکس‌اور را پخش می‌دهد داریم:

$$F_k(r) \oplus CRC32(0^n) \oplus CRC32(m) = F_k(r) \oplus CRC32(0^n \oplus m) = F_k(r) \oplus CRC32(m)$$

پس حمله کننده با مزیت یک جعل می‌کند.

مسأله‌ی ۱۲

فرض کنید G یک تابع قابل پیشبینی باشد و A یک حمله کننده با مزیت غیرناچیز μ برای پیشبینی G باشد، حال حمله کننده‌ی A' را مزیت غیرناچیز برای تمایز G از یک تابع تصادفی می‌سازیم.
برای اینکار A' با اوراکل F، مقدار $A(1^n) = i$ را محاسبه می‌کند، سپس یک ورودی تصادفی مثل s می‌سازد و آن‌را به تابع ورودی‌اش F می‌دهد، حال i عضو اول $F(s)$ را به A می‌دهد و بیت g را می‌گیرد. اگر $g = F(s)[i]$ بود یک و در غیراینصورت صفر برمی‌گرداند.
حال اگر F برابر با G باشد، A به احتمال $\mu(n) + \frac{1}{2}$ درست پیشبینی کرده و A' به همین احتمال یک می‌دهد. و اگر F کاملا تصادفی باشد، A' به احتمال یک‌دوم یک برمی‌گرداند پس مزیت A برابر با:

$$|\mu(n) + \frac{1}{2}| - \frac{1}{2}| = \mu(n)$$

که غیرناچیز است، پس G PRF نیست.