

۱. سوال ۱۲.۳

۱.۱.

$$c_i = z_i \oplus \{x_1 x_2 \cdots x_n || H_1(x) H_2(x) \cdots H_m(x)\} \quad i = 1, 2, \dots, n + m$$

با فرض این که x تعداد n بیت دارد، اسکار ابتدا مقدار زیر را محاسبه می کند:

$$z_i = x_i \oplus c_i \quad i = 1, 2, \dots, n$$

با توجه به این که اسکار مقدار x را می داند، $H(x)$ را محاسبه می کند؛ با فرض این که $H(x)$ دارای m بیت خروجی است، اسکار مقدار زیر را محاسبه می کند:

$$z_{j+n} = H_j(x) \oplus c_{j+n} \quad j = 1, 2, \dots, m$$

سپس اسکار مقدار $H(x')$ را بدست می آورد و در انتها مقادیر زیر را محاسبه می کند:

$$c'_i = z_i \oplus x'_i \quad i = 1, 2, \dots, n$$

$$c'_{j+n} = z_{j+n} \oplus H_j(x') \quad j = 1, 2, \dots, m$$

با توجه به این که مهاجم کلید را با استفاده از plaintext و ciphertext بدست می آورد، این حمله در صورت استفاده از OTP نیز قابل اجرا است. البته با توجه به این که کلید هر سری متفاوت است؛ مهاجم برای هر پیام رد و بدل شده باید همه ی مراحل بالا را انجام دهد.

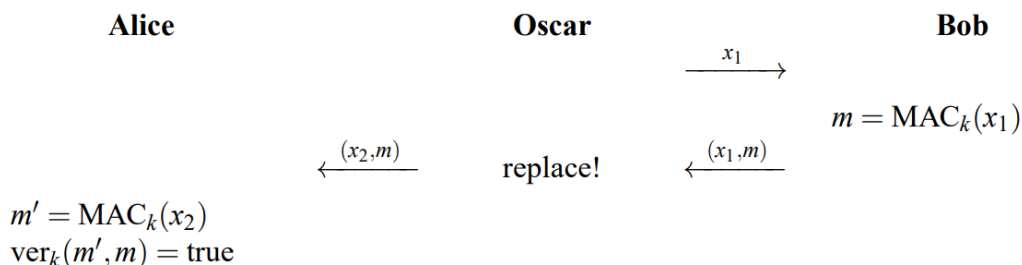
۲.۱.

خیر، با توجه به این که اسکار مقادیر z_1, z_2, \dots, z_n را می تواند بازیابی کند ولی قادر به بازیابی رشته بیت $z_{n+1}, z_{n+2}, \dots, z_{n+m}$ که برای رمز کردن $MAC_{k_2}(x)$ استفاده می شود، نیست. حتی اگر کل رشته بیت را بداند؛ باز هم به دلیل این که مقدار k_2 را نمی داند، قادر به محاسبه $MAC_{k_2}(x')$ نیست.

۲. سوال ۱۲.۵

۱.۲.

در این جا فرض می شود که اسکار می تواند باب را فریب دهد تا پیام x_1 را امضا کند.



۲.۲

برای ایجاد کالیزن، اسکار باید $\sqrt{2^n}$ مک را محاسبه کند. با توجه به این که اسکار مقدار کلید مخفی را نمی‌داند، باید به نحوی آلیس و یا باب را فریب دهد تا مقدار MAC را برای تعداد زیادی پیام محاسبه کند، که در عمل امکان‌پذیر نیست. از طرف دیگر، کالیزن در تابع هش توسط اسکار و بدون کمک آلیس و باب می‌تواند ایجاد شود، زیرا محاسبات بدون استفاده از کلید انجام می‌گیرد. بنابراین یک تابع مک ۸۰ بیتی، امنیت 2^{80} را ایجاد می‌کند، زیرا حملات کالیزن قابل اجرا نیستند و یک تابع هش با اندازه خروجی یکسان، تنها امنیت در حدود 2^{40} را به وجود می‌آورد.

۳. سوال ۱۳.۹

۱.۳

$$t = 10^6 \text{ bit/sec}$$

$$\text{storage} = t \cdot r = 2h \cdot 10^6 = 2 \cdot 3600 \cdot 10^6 = 7.2 \text{ Gbit} = 0.9 \text{ GByte}$$

بنابراین ذخیره‌سازی کمتر از ۱ گیگابایت را می‌توان با هزینه‌های کم انجام داد.

۲.۳

تعداد کلیدهایی که یک مهاجم در طی ۳۰ روز می‌تواند به آن‌ها دست پیدا کند، برابر است با:

$$\text{keys} = \frac{30 \text{ days}}{10 \text{ min}} = \frac{30 \cdot 24 \cdot 60}{10} = 4320$$

مدت زمان استخراج هر کلید برابر است با:

$$T_{\text{key derive}} = \frac{2h}{4320} = \frac{2 \cdot 3600}{4320} = 1.67 \text{ sec}$$

با توجه به این که توابع هش سریع هستند، بنابراین عملیات استخراج کلید می‌تواند به راحتی و با این سرعت انجام گیرد.

۴. سوال ۱۳.۱۱

Alice

$$A = 2^{228} \bmod 467 = 394$$

$$K_{AO} = O^a \bmod p = 156^{228} \bmod 467 = 243$$

Bob

$$B = 2^{57} \bmod 467 = 313$$

$$K_{BO} = O^b \bmod p = 156^{57} \bmod 467 = 438$$

Oscar

$$O = 2^{16} \bmod 467 = 156$$

$$K_{AO} = A^O \bmod p = 394^{16} \bmod 467 = 243$$

$$K_{BO} = B^O \bmod p = 313^{16} \bmod 467 = 438$$

۵. سوال ۱۳.۱۵

امضای CA صرفاً کلید عمومی یک کاربر را پوشش می‌دهد. حتی اگر تمامی پارامترهای الگوریتم امضا اعلام شود، کلید خصوصی به دلیل مسئله لگاریتم گسسته همچنان غیرقابل محاسبه است. پس اسکار نمی‌تواند کلیدهای جلسه‌ای که قبل از تشخیص کلید و الگوریتم امضای CA استفاده شده‌است را محاسبه کند. ولی اسکار اکنون می‌تواند با ارائه گواهی‌های جعلی، خودش را به عنوان هر کاربر دلخواه معرفی کند.