Fundamentals of Cryptography

Homework 7

Sepehr Ebadi 9933243

Question 1

به صورت کلی به دلیل اینکه در MAC ما کلید داریم.

در مورد توابع هش، هدف این است که اطمینان حاصل شود احتمال وقوع تصادم (دو ورودی مختلف با خروجی یکسان) بسیار کم باشد. که در این مدل حمله احتمال تصادم به صورت نمایی افزایش پیدا میکند. در خروجی ۱۶۰ بیتی، احتمال وقوع تصادم بسیار کم است، اما در خروجی‌های کوتاه‌تر مثل ۸۰ بیت، این احتمال به شدت افزایش می‌یابد و امنیت تابع هش کاهش پیدا می‌کند.

اما در موردMAC ، خروجی ۸۰ بیتی کافی است، زیرا:

در سیستمMAC ، کلید مخفی k بین فرستنده و گیرنده به اشتراک گذاشته می‌شود. این یعنی، برای حمله به سیستم، اسکار باید کلید مخفی k را پیدا کند.

حمله اسکار شامل تلاش برای جعل پیام x و ایجاد یک MAC معتبر است، اما برای انجام این کار باید کلید مخفی k را بداند. این به طور معمول نیازمند بررسی۲ به توان ۸۰ ترکیب مختلف برای یافتن کلید یا تصادفی‌سازی پیام است، که عملاً برای طول خروجی ۸۰ بیتی کافی است.

بنابراین، برای MAC طول ۸۰ بیتی کفایت می‌کند، زیرا امنیت آن نه تنها به طول خروجی بلکه به وجود کلید مخفی وابسته است.

Question 2

1)

دلیلش این است که چون الگوریتم رمز دنباله ای است و یک پامی از ان را مثل x را میداند :

$$k \oplus x = c$$

حال با داشتن یک پیام x و متن رمز شده ان میتواند کلید را بدست بیاورد :

$$x \oplus c = k$$

حال اسکار میتونه پیام x’ و هش ان را محاسبه کند و با کلید رمزش کند و ارسالش کند :

$$k \oplus x' || h(x') = c'$$

خیر، این حمله در رمزنگاری با یک‌بار رمز (One-Time Pad) مؤثر نیست.

One-Time Pad از یک کلید کاملاً تصادفی استفاده می‌کند که فقط یک بار استفاده می‌شود. بنابراین:

اسکار نمی‌تواند کلید k را محاسبه کند حتی اگر پیام x و متن رمز c را داشته باشد، زیرا کلید کاملاً تصادفی و بی‌ارتباط با متن است.

2)

اگر به جای یک تابع هش ساده h(x)، از یک تابع هش مبتنی بر کلید (MAC) استفاده شود، فرمول پیام رمزنگاری‌شده به صورت زیر تغییر می‌کند:

$$e_k(x||MAC(x)) = c$$

در این حالت از کلید ۲ به توان k برای تابع مک استفاده میشود.

اگر پیام x را بداند همچنان باز میتواند کلید رمز دنباله ای را پیدا کند.

اما برای جایگزین کردن پیام x با پیام دلخواه x'، اسکار باید یک MAC(x') معتبر تولید کند. از آنجایی که محاسبه این نیاز به کلید ۲ به توان k دارد و اسکار به این کلید دسترسی ندارد، او نمی‌تواند c' معتبری بسازد.

Question 3

Diffie-Hellman : $p = 18, \alpha = 18$

Alice : $a = 11, ID(A) = 1, K_{EA} = 213$

Bob : $b = 22, ID(B) = 2, K_{EB} = 215$

Charley : $c = 33, ID(C) = 3, K_{EC} = 217$

CA : $P' = 467, d' = kpriCA = 127, \alpha' = 2, \beta = kpubCA = {\alpha'}^{d'} \ mod\ p' = \ 2^{127}\ mod\ 467 = 132$

$x_A = 4 * 11 + 1 = 45$
$x_B = 4 * 22 + 2 = 90$
$x_C = 4 * 33 + 3 = 135$

1)

$$Cert(Alice) = \left(A, ID(A), sig_{d'}\left(A, ID(A)\right)\right)$$
$$rA = {\alpha'}^{K_{EA}}\ mod\ p' = 2^{213}\ mod\ 467 = 29$$

$$sA = (x_A - d'.rA){K_{EA}}^{-1}\ mod\ p' - 1 = (45 - 127.29)213^{-1} mod\ 466$$
$$= (-3638)431\ mod\ 466 = 112$$
$$\boldsymbol{Cert(Alice)} = \left(\boldsymbol{K_{EA}}, \boldsymbol{ID(A)}, \boldsymbol{sig_{d'}}\left(\boldsymbol{K_{EA}}, \boldsymbol{ID(A)}\right)\right) = (\mathbf{213}, \mathbf{1}, (\mathbf{29}, \mathbf{112}))$$

$$rB = {\alpha'}^{K_{EB}}\ mod\ p' = 2^{215}\ mod\ 467 = 116$$
$$sB = (x_B - d'.rB){K_{EB}}^{-1}\ mod\ p' - 1 = (90 - 127.116)215^{-1} mod\ 466$$
$$= (-14642)453\ mod\ 466 = 218$$
$$\boldsymbol{Cert(Bob)} = \left(\boldsymbol{K_{EB}}, \boldsymbol{ID(B)}, \boldsymbol{sig_{d'}}\left(\boldsymbol{K_{EB}}, \boldsymbol{ID(B)}\right)\right) = (\mathbf{215}, \mathbf{2}, (\mathbf{116}, \mathbf{218}))$$

$$rC = {\alpha'}^{K_{EC}}\ mod\ p' = 2^{217}\ mod\ 467 = 464$$
$$sC = (x_C - d'.rC){K_{EC}}^{-1}\ mod\ p' - 1 = (135 - 127.464)217^{-1} mod\ 466$$
$$= (-58793)131\ mod\ 466 = 165$$
$$\boldsymbol{Cert(Charley)} = \left(\boldsymbol{K_{EC}}, \boldsymbol{ID(C)}, \boldsymbol{sig_{d'}}\left(\boldsymbol{K_{EC}}, \boldsymbol{ID(C)}\right)\right) = (\mathbf{217}, \mathbf{3}, (\mathbf{464}, \mathbf{9}))$$

2)

Alice:

$$verify\left(x_A, (rA, sA)\right) = \left(45, (29{,}112)\right)$$
$$t = \beta^r.r^s\ mod\ p' = 132^{29}.29^{112}\ mod\ 467 = 80$$
$${\alpha'}^{45}\ mod\ 467 = 2^{45}\ mod\ 467 = 80 \rightarrow Valid$$

Bob:

$$verify\left(x_B, (rB, sB)\right) = \left(90, (116{,}218)\right)$$
$$t = \beta^r.r^s\ mod\ p' = 132^{116}.116^{218}\ mod\ 467 = 329$$
$${\alpha'}^{90}\ mod\ 467 = 2^{90}\ mod\ 467 = 329 \rightarrow Valid$$

Charley:

$$verify\left(x_C, (rC, sC)\right) = \left(135, (464{,}165)\right)$$
$$t = \beta^r.r^s\ mod\ p' = 132^{464}.464^{165}\ mod\ 467 = 168$$
$${\alpha'}^{135}\ mod\ 467 = 2^{135}\ mod\ 467 = 168 \rightarrow Valid$$

3)

$$K_{AB} = (a^a)^b = (18^{11})^{22}\ mod\ 61 = 19$$
$$K_{AC} = (a^a)^c = (18^{11})^{33}\ mod\ 61 = 37$$
$$K_{BC} = (a^b)^c = (18^{22})^{33}\ mod\ 61 = 27$$

Question 4

1)

در اولی کلید جلسه به صورت خطی و معکوس کلید جلسه قبلی، در دومی از تابع هش استفاده شده و وابستگی غیر خطی و غیر معکوس کلید ها، و در سومی برای تولید کلید جلسه بعدی از کلید جلسه قبلی و کلید اصلی استفاده میشود.

2)

دومی و سومی (b,c) چون کلید جلسه ها را نمیتوان از کلید جلسه های اخر بدست اورد.

3)

در اولی (a) همه جلسه ها زیرا pfs وجود ندارد.

در دومی (b) همه جلسه هایی که که از کلید جلسه nام استفاده میکنند و همه جلسه های بعدی.

در سومی (c) فقط جلسه اخر.چون کلید اصلی که نامعلوم است برای تولید کلید های بعدی استفاده شده.

4)

همه کلیدها.

Question 5

1)

Oscarمی‌خواهد که Bob فکر کند که کلید عمومی CA که دریافت کرده است، درست است، در حالی که در واقع کلید عمومی CA توسط او جایگزین شده است. در اینجا مراحل دقیقی که Oscar باید انجام دهد آورده شده است:

Bobگواهی را با ارسال درخواست به CA دریافت می‌کند.

Oscarکه بر روی ارتباطات Bob کنترل دارد، می‌تواند کلید عمومی CA را که از Bob درخواست شده است، جایگزین کند.

به این صورت که Oscar می‌تواند کلید عمومی خود را به جای کلید عمومی CA در پیام‌های ارسالی به Bob قرار دهد.

هنگامی که Bob درخواست گواهی را به CA ارسال می‌کند، Oscarاین درخواست را رهگیری کرده و در آن کلید عمومی خودش را به جای کلید عمومی واقعی CA قرار می‌دهد.

سپس، Oscarاین پیام را به CA ارسال می‌کند .پاسخی که در آن گواهی را با کلید عمومی خود Oscar برای Bob می‌فرستد، ارسال می‌کند.

زمانی که Oscar پاسخ را دریافت می‌کند، گواهی را که حاوی کلید عمومی خود اوست، به Bob می‌فرستد.

از آنجا که Bob هیچ راهی برای تأیید کلید عمومی CA ندارد، تصور می‌کند که کلید عمومی که دریافت کرده صحیح است.

در این روش، Bobگواهی و کلید عمومی Oscar را به عنوان کلید عمومی CA قبول می‌کند و هیچ‌گاه متوجه نمی‌شود که کلید عمومی که دریافت کرده غلط است.

2)

در این بخش، Oscarمی‌خواهد که یک کلید جلسه (session key) با Bob برقرار کند و Bob فکر کند که پروتکل را با Aliceاجرا می‌کند، در حالی که در واقع این Oscar است که این پروتکل را با Bob اجرا می‌کند.

ابتدا، Bobو Alice به‌طور معمول پروتکل Diffie–Hellman را آغاز می‌کنند. در این پروتکل، هر طرف یک کلید خصوصی و یک کلید عمومی تولید می‌کند و از آن‌ها برای تبادل کلید مشترک استفاده می‌کنند.

Oscar، که به ارتباطات میان Bob و Alice دسترسی کامل دارد، کلید عمومی Alice را که به Bob ارسال می‌شود، تغییر می‌دهد و به جای آن، کلید عمومی خود را به Bob می‌فرستد.

این کار به این صورت است که وقتی Bob کلید عمومی Alice را برای شروع پروتکل دریافت می‌کند، در حقیقت این کلید عمومی Oscar است که به او فرستاده می‌شود.

مشابه با مرحله قبل، Oscarوقتی که Bob کلید عمومی خود را برای Alice ارسال می‌کند، کلید عمومی خود را جایگزین می‌کند.

این‌طور به نظر می‌آید که Bob و Alice در حال تبادل کلید هستند، اما در واقع، Oscarو Bob کلید مشترک را تعیین می‌کنند.

پس از تعویض کلیدهای عمومی، Oscarو Bob یک کلید مشترک (session key) محاسبه می‌کنند، به طوری که Bob فکر می‌کند این کلید مشترک باید همان چیزی باشد که با Alice به اشتراک گذاشته شده است.

در اینجا، چون Oscar توانسته است تا پروتکل Diffie–Hellman را با استفاده از کلید عمومی خودش جایگزین کند، Bobهیچ‌گاه متوجه نمی‌شود که در واقع Oscar است که با او ارتباط برقرار کرده است، نه.Alice

به همین ترتیب، در مرحله‌ای که Bob بخواهد با Alice ارتباط برقرار کند، کلید جلسه‌ای که او محاسبه کرده، در واقع توسط Oscar کنترل می‌شود و کلید معتبر نخواهد بود.