
امنیت شبکه

علی فانیان

a.fanian@iut.ac.ir

فهرست مطالب

- مبانی رمزنگاری کلید عمومی
- کاربردهای رمزنگاری کلید عمومی
 - توزیع کلید
 - امضای دیجیتال
- توابع یک طرفه
- پروتکل توافق کلید Diffie-Helman
- الگوریتم رمزنگار نامتقارن RSA

مبانی رمزنگاری کلید عمومی

- رمزنگاری کلید عمومی اساساً با انگیزه رسیدن به دو هدف طراحی شد:
 - حل مساله توزیع کلید
 - امضای رقمی (دیجیتال)

نمادها و قراردادهای

□ **کلید عمومی** : کلید رمز گذاری

➤ این کلید را برای شخص A با PU_a نشان می دهیم

□ **کلید خصوصی** : کلید رمز گشایی

➤ این کلید را برای شخص A با PR_a نشان می دهیم

رمز نگاری کلید عمومی

- کلید های رمز گذاری و رمز گشایی متفاوت اما مرتبط هستند.
- رسیدن به کلید رمز گشایی از کلید رمز گذاری از لحاظ محاسباتی ناممکن می باشد.
- رمز گذاری امری همگانی میباشد و اساساً نیازی به اشتراک گذاشتن اطلاعات محرمانه ندارد.
- رمز گشایی از طرف دیگر امری اختصاصی بوده و محرمانگی پیامها محفوظ میماند.

نیازمندیهای رمزنگاری کلید عمومی

□ از نظر محاسباتی برای طرف **B**، تولید یک زوج کلید آسان باشد

□ برای فرستنده، تولید متن رمز آسان باشد:

$$C = E_{PU_b}(M)$$

□ برای گیرنده، رمزگشایی متن با استفاده از کلید خصوصی آسان باشد

$$M = D_{PR_b}(C) = D_{PR_b}(E_{PU_b}(M))$$

نیازمندیهای رمزنگاری کلید عمومی

❑ از نظر محاسباتی تولید کلید خصوصی با دانستن کلید عمومی غیر ممکن باشد

❑ بازیابی پیام M ، با دانستن PU_b و C غیر ممکن باشد

$$C = E_{PU_b}(M)$$

❑ ویژگی تقارنی: از هر یک از کلیدها می توان برای رمزکردن استفاده نمود. در این صورت از کلید دیگر برای رمزگشایی استفاده می شود.

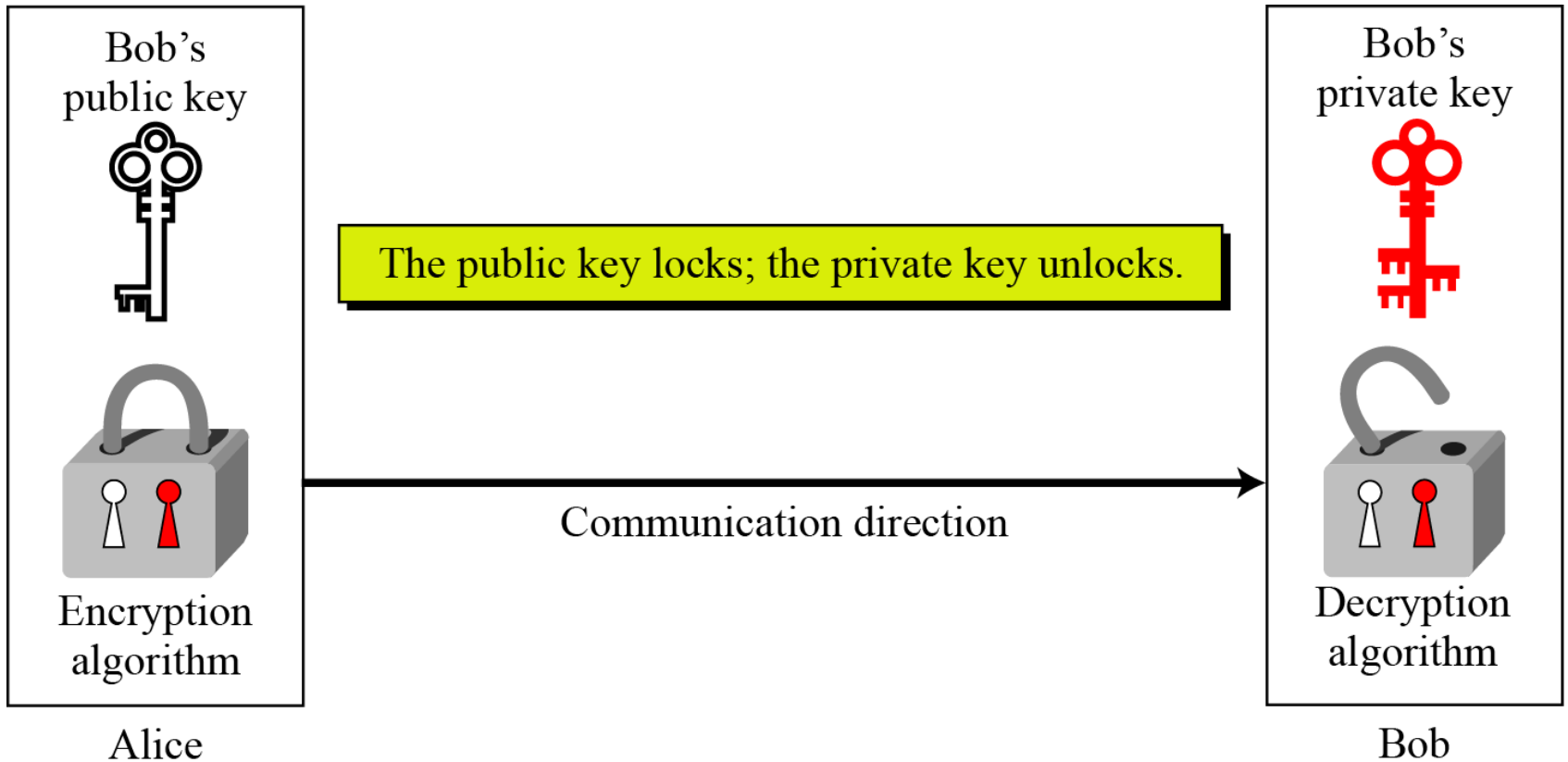
$$M = D_{PR_b}(E_{PU_b}(M)) = D_{PU_b}(E_{PR_b}(M))$$

رمزگذاری کلید عمومی

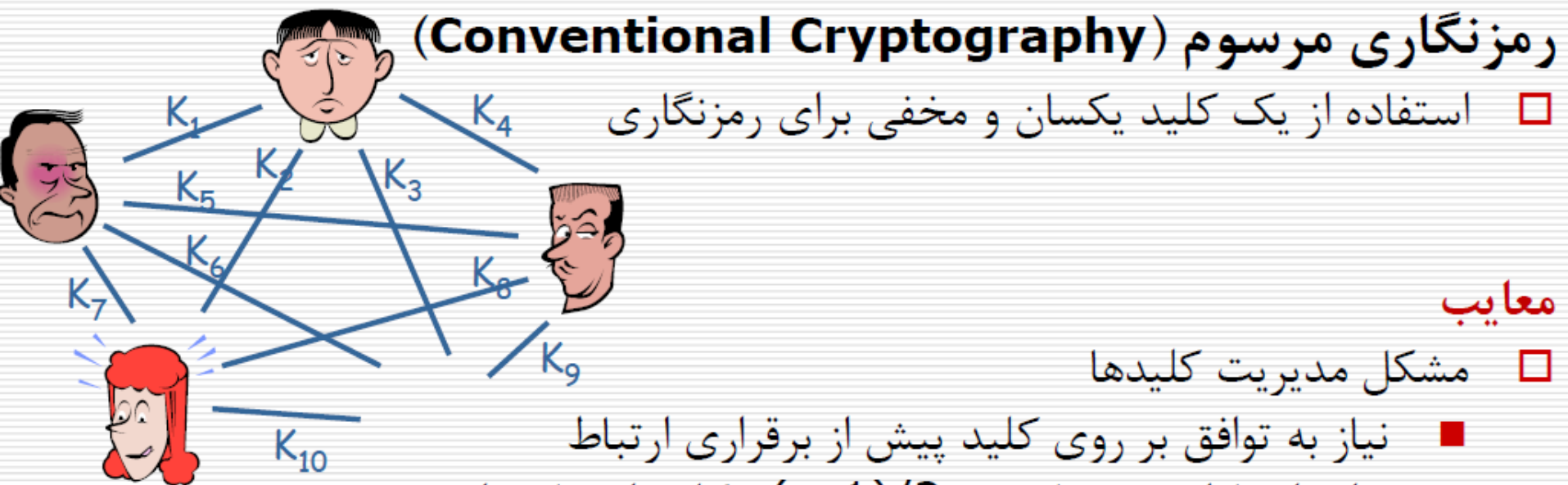
□ برای رمز نگاری کلید عمومی گامهای زیر را برمیداریم:

- هر کاربر یک زوج کلید رمزگذاری و رمز گشایی تولید میکند.
- کاربران کلید رمزگذاری خود را به صورت عمومی اعلان میکنند درحالی که کلید رمز گشایی مخفی میباشد.
- همگان قادر به ارسال پیام رمز شده برای هر کاربر دلخواه با استفاده از کلید رمزگذاری (عمومی) او میباشند.
- هر کاربر میتواند با کمک کلید رمزگشایی (خصوصی) پیامهایی که با کلید رمزگذاری (عمومی) او رمز شده رمزگشایی کند.

رمزگذاری با کلید عمومی



مقایسه رمزنگاری مرسوم و رمزنگاری کلید عمومی



معایب

- ❑ مشکل مدیریت کلیدها
- نیاز به توافق بر روی کلید پیش از برقراری ارتباط
- برای ارتباط n نفر باهم به $n(n-1)/2$ کلید احتیاج داریم.
- ❑ عدم پشتیبانی از امضاء الکترونیکی

مزایا

- ❑ با این وجود از الگوریتم‌های رمزنگاری با کلید عمومی سریع‌تر است.

جایگزینی یا تکمیل؟

از نظر کاربردی، رمزگذاری با کلید عمومی بیش از آنکه
جایگزینی برای رمزگذاری مرسوم باشد نقش **مکمل** آنرا برای
حل مشکلات توزیع کلید بازی می کند.

Misconceptions!



دو تصور اشتباه دیگر درباره کلید عمومی

– رمزنگاری با کلید عمومی امن تر است!

• در هر دو روش رمزنگاری امنیت به طول کلید وابسته است.

– مسئله توزیع کلید در رمزنگاری با کلید عمومی برطرف شده است

• چگونه مطمئن شویم کلید عمومی لزوما متعلق به شخص ادعاکننده است؟!

• توزیع کلید عمومی آسانتر است، ولی بدیهی نیست.

توابع یک طرفه

- تعریف: تابع $f(.)$ را یک طرفه گوئیم اگر:
 - $f(.)$ در زمان چند جمله ای قابل محاسبه **باشد**.
 - $f^{-1}(.)$ در زمان چند جمله ای قابل محاسبه **نباشد**.

توجه:

- تابع $f(.)$ لزوماً یک به یک نیست.

مثال: لگاریتم گسسته

q یک عدد اول و α یک مولد برای این عدد میباشد.

(یعنی هر عدد بین ۱ تا q را میتوان به صورت توانی از α نشان داد.)

$$\{1, \alpha, \alpha^2, \dots, \alpha^{q-1}\}$$

مثلا عدد ۵ یک مولد برای عدد اول ۲۳ است

اگر $a = \alpha^b \bmod q$ باشد آنگاه یافتن b از روی a را محاسبه لگاریتم گسسته گویند.

فرض: محاسبه لگاریتم گسسته با بزرگ شدن پارامترها از لحاظ محاسباتی ناممکن است در حالی که نمارسانی گسسته همچنان به سادگی میسر است.

مثال: لگاریتم گسسته

اگر $a = \alpha^b \bmod q$ باشد آنگاه یافتن b از روی a را محاسبه لگاریتم گسسته گویند.

فرض: محاسبه لگاریتم گسسته با بزرگ شدن پارامترها از لحاظ محاسباتی ناممکن است در حالی که نمارسانی گسسته همچنان به سادگی میسر است.

محاسبه نمای گسسته

• برای محاسبه $a^b \pmod{N}$ الگوریتمهای متفاوتی ابداع شده است...

– فرض کنید $b_k b_{k-1} \dots b_0$ نمایش مبنای ۲ عدد b باشد.

– بنابراین خواهیم داشت:

$$a^b = a^{\sum_{b_i \neq 0} 2^i} = \prod_{b_i \neq 0} a^{2^i}$$

$$a^b \pmod{n} = \left[\prod_{b_i \neq 0} a^{2^i} \right] \pmod{n} = \left[\prod_{b_i \neq 0} \left(a^{2^i} \pmod{n} \right) \right] \pmod{n}$$

الگوریتم تولید کلید مشترک Diffie - Hellman

توافق بر روی مقادیر α و q

Alice

Bob

عدد تصادفی X_A را انتخاب میکند

عدد تصادفی X_B را انتخاب میکند

$$Y_A = \alpha^{X_A} \bmod q$$

$$Y_B = \alpha^{X_B} \bmod q$$

$$K_{AB} = (Y_B)^{X_A} \bmod q$$

$$K_{AB} = (Y_A)^{X_B} \bmod q$$

کلید مشترک برابر است با $\alpha^{(X_A \times X_B)} \bmod q$

الگوریتم Diffie - Hellman

• مثال

– توافق روی $\alpha=3$ و $q=353$

– تولید کلیدهای مخفی

• انتخاب $x_A=97$ توسط A و $x_B=233$ توسط B

– محاسبه کلید عمومی

$$y_A = 3^{97} \bmod 353 = 40 \cdot$$

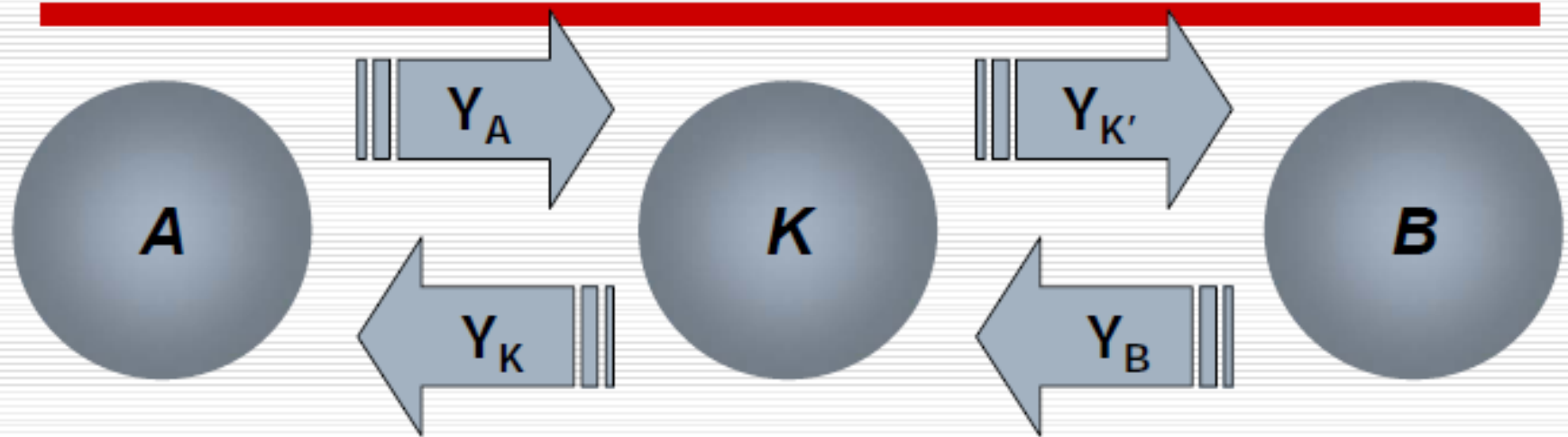
$$y_B = 3^{233} \bmod 353 = 248 \cdot$$

– محاسبه کلید جلسه مورد توافق

$$K_{AB} = y_B^{x_A} \bmod 353 = 248^{97} = 160 \cdot$$

$$K_{AB} = y_A^{x_B} \bmod 353 = 40^{233} = 160 \cdot$$

حمله مردی در میان



$$K_1 = \alpha^{(X_A \times X_K)} \bmod q$$

A گمان می کند
کلید K_1 را با B
به اشتراک
گذاشته است.

$$K_2 = \alpha^{(X_A \times X_{K'})} \bmod q$$

B گمان می کند
کلید K_2 را با A به
اشتراک گذاشته
است.

RSA

- کلیات

- توسط Rivest-Shamir -Adelman در سال ۱۹۷۷ در MIT ارائه شد
- مشهورترین و پرکاربردترین الگوریتم رمزگذاری کلید عمومی
- مبتنی بر توان رسانی پیمانه ای
- استفاده از اعداد طبیعی خیلی بزرگ
- امنیت آن ناشی از دشوار بودن تجزیه اعداد بزرگ، که حاصلضرب دو عامل اول بزرگ هستند، می باشد.
- مستندات مربوط به آن تحت عنوان PKCS استاندارد شده است.

نمادگذاری RSA

- انتخاب دو عدد اول بزرگ p, q و تولید $N = p * q$ به عنوان پیمانه محاسبات
- e : نمای رمزگذاری (کلید عمومی شامل e, N)
- d : نمای رمزگشایی (کلید خصوصی شامل d, N)
- M : پیام
- تابع رمزگذار RSA: $x \rightarrow x^e \bmod N$
- تابع رمزگشای RSA: $x \rightarrow x^d \bmod N$

الگوریتم Pohling-Hellman

• مفروضات:

• N عدد بزرگ می باشد.

$$d \times e \equiv 1 \pmod{\varphi(N)}$$
$$\varphi(N) = (p-1)(q-1)$$

• رمزگذاری:

$$C = M^e \pmod{N}$$

• رمزگشایی:

$$\begin{aligned} C^d \pmod{N} &= (M^e)^d \pmod{N} \\ &= M^{ed} \pmod{N} = M^{ed \pmod{\varphi(N)}} \pmod{N} \\ &= M \pmod{N} = M \end{aligned}$$



مبانی ریاضی RSA

- p و q دو عدد اول می باشند.
- $\varphi(N)$: تعداد اعداد (کوچکتر از N) که نسبت به N اول است.

$$N = p \times q$$

$$\varphi(N) = (p - 1) \times (q - 1)$$

$$\gcd(\varphi(N), e) = 1$$

$$d \times e \equiv 1 \bmod \varphi(N)$$

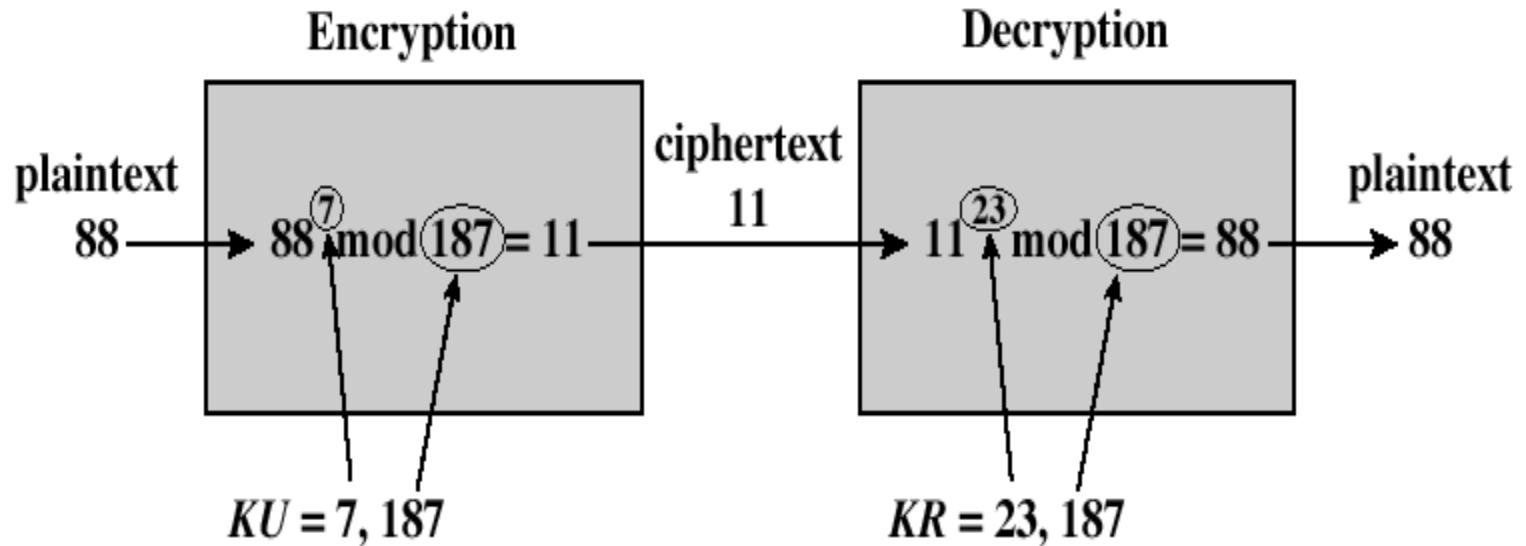
$$C = M^e \bmod N$$

$$M = C^d \bmod N = (M^e)^d \bmod N$$

RSA

- هم فرستنده و هم گیرنده مقدار N را می دانند
- فرستنده مقدار e را می داند
 - کلید عمومی : (N, e)
- تنها گیرنده مقدار d را می داند
 - کلید خصوصی : (N, d)
- نیازمندیها:
 - محاسبه M^e و C^d آسان باشد
 - محاسبه d با دانستن کلید عمومی غیرممکن باشد

RSA-مثال



$$p = 17, q = 11, n = p * q = 187$$

$$\Phi(n) = 16 * 10 = 160, \text{ pick } e=7, d.e=1 \bmod \Phi(n) \Rightarrow d = 23$$

کاربردهای رمزنگاری کلید عمومی

- دسته بندی کلی کاربردها

– رمزگذاری / رمز گشایی : برای حفظ محرمانگی

– امضاء رقمی : برای حفظ اصالت پیام و معین نمودن

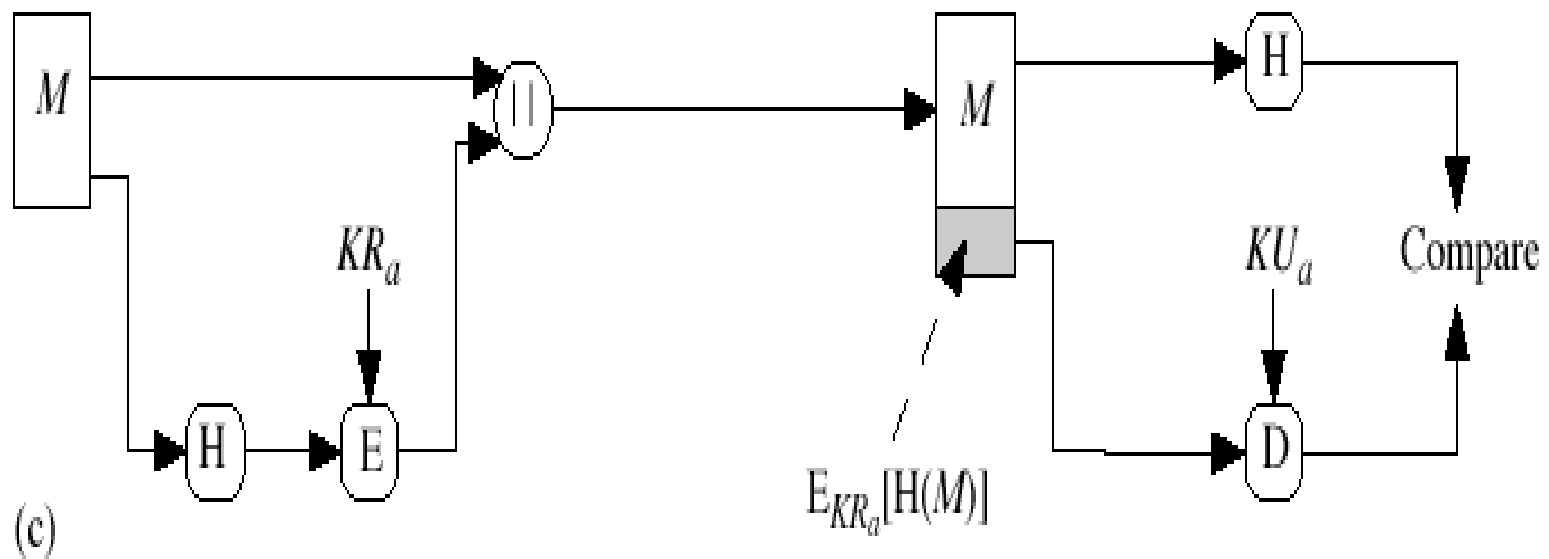
فرستنده پیام (پیوند دادن پیام با امضاء کننده)

– توزیع کلید : برای توافق طرفین روی کلید جلسه مخفی

جایگاه عملی رمزنگاری کلید عمومی

- کلیدهای این نوع از الگوریتمها بسیار طولانی تر از الگوریتمهای مرسوم (کلید پنهان) میباشند.
 - الگوریتم **RSA** با پیمانه ۱۰۲۴ بیتی امنیتی در حد الگوریتمهای متقارن با کلیدهای ۸۰ بیتی دارد.
- سرعت الگوریتمهای کلید عمومی از الگوریتمهای رمزگذاری مرسوم پایین تر است.
 - **RSA** تقریباً ۱۰۰۰ بار کند تر از رمزهای کلید پنهان (با امنیت یکسان) میباشد.

RSA



کاربردهای برخی الگوریتم های کلید عمومی

الگوریتم	رمزگذاری / رمز گشایی	امضاء رقومی	توزیع کلید
RSA	✓	✓	✓
Diffie-Hellman	×	×	✓
DSS	×	✓	×
Elliptic Curve	✓	✓	✓

لغت نامه

Public Key	کلید عمومی
Authentication	احراز اصالت
Message Integrity	اصالت پیام
Cipher Text	پیام رمز شده
Plain Text	پیام واضح
Binding	پیوند دادن
Modulus	پیمانه
One Way Function	تابع یک طرفه
Factorization	تجزیه
Modular Exponentiation	نما رسانی پیمانه ای

Probabilistic Polynomial Time	زمان چند جمله ای تصادفی
Trapdoor	دریچه
Decryption	رمز گشایی
Encryption	رمز گذاری
Factor	عامل
Session Key	کلید جلسه
Confidential	محرمانه
Man In The Middle	فردی در میان
Conventional	مرسوم
Infeasible	ناممکن
Exponent	نما
One to One	یک به یک

