یا ذاالامن والامان

امنیت داده و شبکه

# رمزنگاری نامتقارن (کلید عمومی)

مرتضی امینی - سیدمهدی خرازی

نیم‌سال اول ۱۴۰۴-۱۴۰۳

# فهرست مطالب

- مبانی رمزنگاری کلید عمومی
- مقایسه با رمزنگاری سنتی و متقارن
- کاربردهای رمزنگاری کلید عمومی
- الگوریتم رمز RSA
- الگوریتم تبادل کلید دیفی-هلمن
- الگوریتم رمز الجمل

# فهرست مطالب

- مبانی رمزنگاری کلید عمومی
- مقایسه با رمزنگاری سنتی و متقارن
- کاربردهای رمزنگاری کلید عمومی
- الگوریتم رمز RSA
- الگوریتم تبادل کلید دیفی-هلمن
- الگوریتم رمز الجمل

# رمزنگاری کلید عمومی

- هر فرد دارای یک **زوج کلید عمومی و خصوصی** است.
- کلیدهای عمومی و خصوصی متفاوت اما مرتبط هستند.
- رسیدن به کلید خصوصی از کلید عمومی از لحاظ محاسباتی ناممکن است.

# نمادها و قراردادها

- **کلید عمومی:**
  - این کلید را برای شخص A با $PU_a$ نشان می‌دهیم.
  - کلید رمزگذاری (در حفظ محرمانگی)
  - کلید وارسی (در کنترل صحت با امضای دیجیتال)

- **کلید خصوصی:**
  - این کلید را برای شخص A با $PR_a$ نشان می‌دهیم.
  - کلید رمزگشایی (در حفظ محرمانگی)
  - کلید تولید امضاء (در کنترل صحت با امضای دیجیتال)

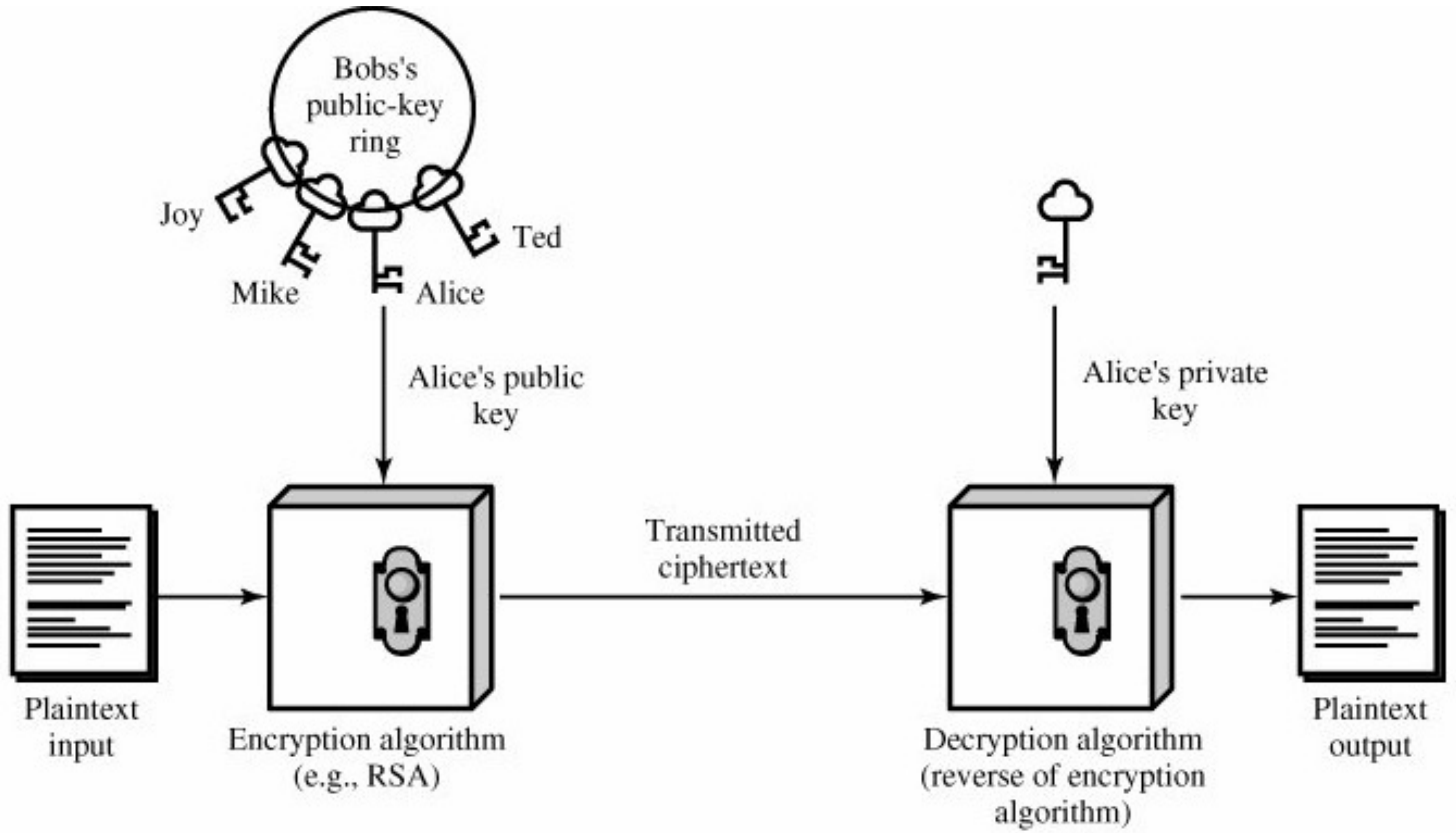
# نیازمندیهای رمزنگاری کلید عمومی

- از نظر محاسباتی، تولید کلید خصوصی ($PR_b$) با دانستن کلید عمومی ($PU_b$) غیرممکن باشد.

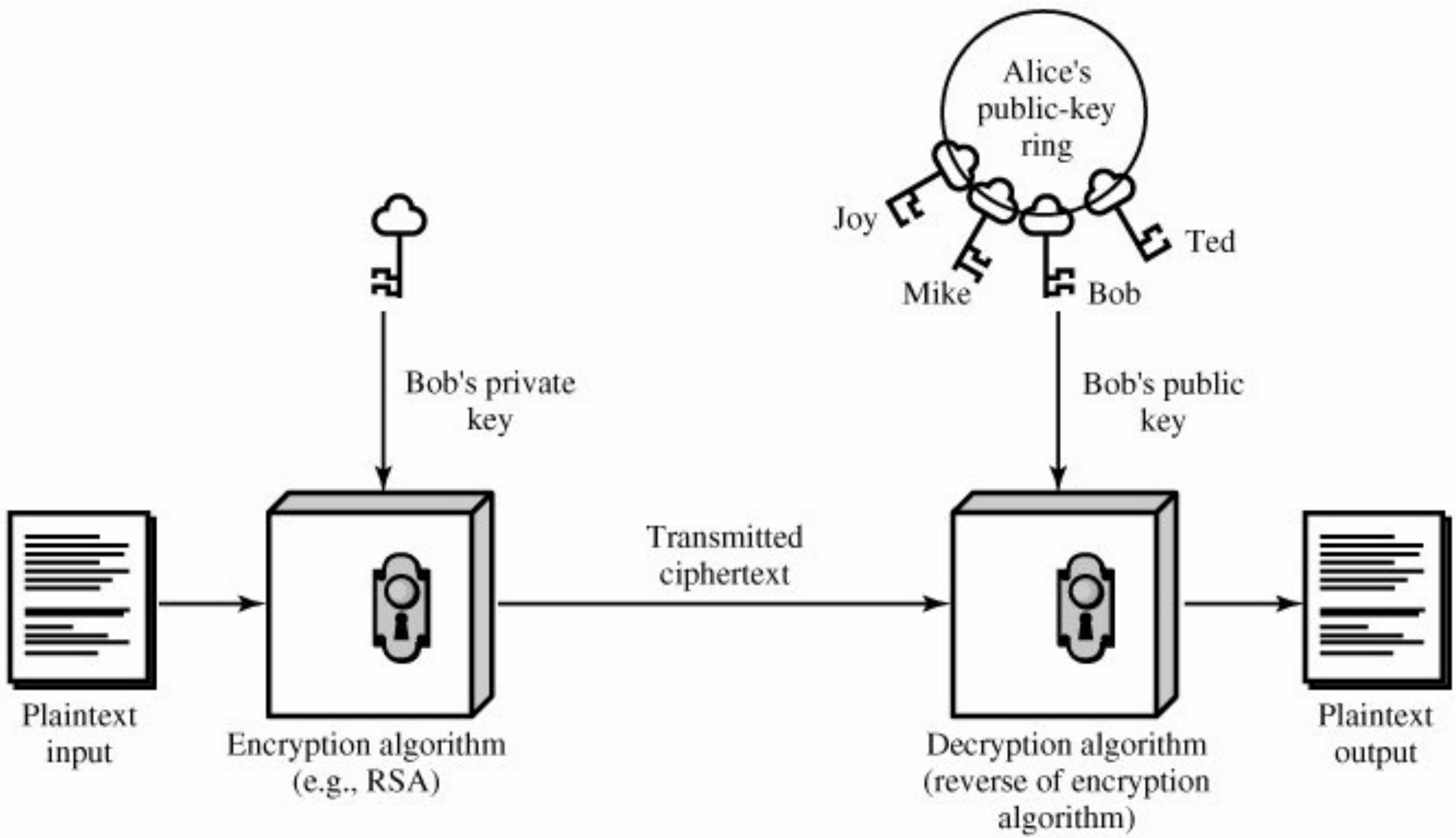
- ویژگی تقارنی: از هر یک از کلیدها می‌توان برای رمزکردن استفاده کرد. در این صورت از کلید دیگر برای رمزگشایی استفاده می‌شود.

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

# رمزگذاری با کلید عمومی (در حفظ محرمانگی)



# رمزگذاری با کلید خصوصی (در کنترل صحت و اصالت)



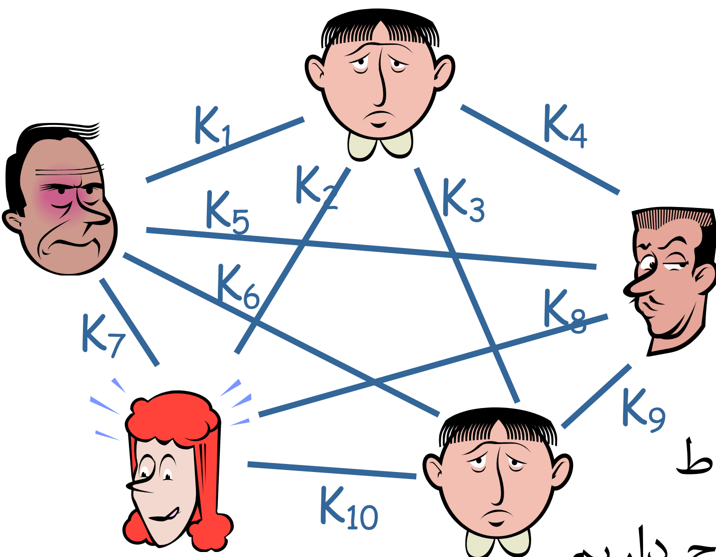
# فهرست مطالب

- مبانی رمزنگاری کلید عمومی
- **مقایسه با رمزنگاری سنتی و متقارن**
- کاربردهای رمزنگاری کلید عمومی
- الگوریتم رمز RSA
- الگوریتم تبادل کلید دیفی-هلمن
- الگوریتم رمز الجمل

# مقایسه رمزنگاری متقارن و رمزنگاری کلید عمومی

## رمزنگاری متقارن

- استفاده از یک کلید یکسان و مخفی برای رمزنگاری




### 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 رقمی (دیجیتال) و عدم ارایه سرویس عدم انکار

### مزایا

- با این وجود، رمز متقارن از رمز نامتقارن (رمز کلید عمومی) سریع‌تر است.

# جایگزینی یا تکمیل؟

از نظر کاربردی، رمزگذاری با کلید عمومی بیش از آنکه جایگزینی برای رمزگذاری متقارن باشد, نقش مکمل آن را برای حل مشکلات توزیع کلید بازی می‌کند.

# سوء برداشت!

- دو تصور اشتباه دیگر درباره الگوریتم‌های کلید عمومی
  - رمزنگاری با کلید عمومی امن‌تر است!
    - در هر دو روش رمزنگاری امنیت به طول کلید وابسته است.
  - مسأله توزیع کلید در رمزنگاری با کلید عمومی برطرف شده است!
    - چگونه مطمئن شویم کلید عمومی لزوما متعلق به شخص ادعاکننده است؟!
    - پس توزیع کلید عمومی آسانتر است، ولی بدیهی و بدون مشکل نیست.

# فهرست مطالب

- مبانی رمزنگاری کلید عمومی
- مقایسه با رمزنگاری سنتی و متقارن
- **کاربردهای رمزنگاری کلید عمومی**
- الگوریتم رمز RSA
- الگوریتم تبادل کلید دیفی-هلمن
- الگوریتم رمز الجمل

# کاربردهای رمزنگاری کلید عمومی

- **رمزگذاری/ رمزگشایی:** برای حفظ محرمانگی
- **امضاء رقمی:** برای کنترل صحت و اصالت پیام و وارسی فرستنده پیام (پیوند دادن پیام با امضاء کننده) یا همان **عدم انکار**
- **توزیع کلید:** برای توافق طرفین بر روی یک کلید رمز متقارن (به عنوان کلید نشست)، قبل از برقراری ارتباط

# مقایسه رمز نامتقارن با رمز متقارن

- کلیدهای این نوع از الگوریتم‌ها بسیار طولانی‌تر از الگوریتم‌های رمز متقارن هستند.
  - الگوریتم RSA با پیمانه ۱۰۲۴ بیتی امنیتی در حد الگوریتم‌های متقارن با کلیدهای ۸۷ بیتی دارد.
- سرعت الگوریتم‌های کلید عمومی از الگوریتم‌های رمزگذاری متقارن پایین‌تر است.
  - RSA تقریباً ۱۰۰۰ بار کندتر از رمزهای متقارن (با امنیت یکسان) است.

# فهرست مطالب

- مبانی رمزنگاری کلید عمومی
- مقایسه با رمزنگاری سنتی و متقارن
- کاربردهای رمزنگاری کلید عمومی
- **الگوریتم رمز RSA**
- الگوریتم تبادل کلید دیفی-هلمن
- الگوریتم رمز الجمل

# کلیات الگوریتم رمزنگاری RSA

- توسط Rivest–Shamir –Adleman در سال ۱۹۷۷ در MIT
- مشهورترین و پرکاربردترین الگوریتم رمزگذاری کلیدعمومی
- مبتنی بر توان رسانی پیمانه‌ای
- امنیت آن ناشی از دشواری تجزیه اعداد بزرگ
- مستندات مربوط به آن تحت عنوان PKCS استاندارد شده است.

Public Key Cryptography Standards


Ronald Linn Rivest
(1947 – )


Adi Shamir
(1952 – )


Leonard Adleman
(1945 – )

# مبانی ریاضی RSA

- $\mathbb{Z}_n$ : مجموعه اعداد نامنفی کمتر از n
- $\mathbb{Z}_n^*$ : مجموعه اعداد طبیعی کمتر از n و اول نسبت به آن.

- مثال:

$$\mathbb{Z}_{۱۲} = \{۰, ۱, ۲, ۳, ۴, ۵, ۶, ۷, ۸, ۹, ۱۰, ۱۱\}$$

$$\mathbb{Z}_{۱۲}^* = \{۱, ۵, ۷, ۱۱\}$$

# نمادگذاری RSA

- n : پیمانه محاسبات
- e: نمای رمزگذاری
- d: نمای رمزگشایی
- M: پیام، عدد صحیح متعلق به $\mathbb{Z}_n$
- تابع RSA: تابع یکطرفه $C = M^e \bmod n$
- تابع معکوس: $M = C^d \bmod n$

# مبانی ریاضی RSA

- p و q دو عدد اول هستند.
- $\varphi(n)$: تعداد اعداد (کوچکتر از n) که نسبت به n اول است.
- **کلید عمومی: {e,n}**
- **کلید خصوصی: {d,n}**
- طول کلید: تعداد بیتهای پیمانه n

$n=p.q$

$\varphi(n)=(p-1).(q-1)$

$gcd(\varphi(n), e)=1, \quad 1<e<\varphi(n)$

$d.e=1\ [mod\ \varphi(n)], \quad d=e^{-1}\ [mod\ \varphi(n)]$

$C= M^e\ mod\ n, \quad M<n$

$M= C^d\ mod\ n = (M^e)^d\ mod\ n = M^{ed}\ mod\ n = M\ mod\ n$

# روند تولید کلید در RSA

۱– ابتدا دو عدد اول بزرگ $p$ و $q$ را به طور تصادفی انتخاب کن به گونه‌ای که $p \neq q$

۲– عدد $n$ و $\phi(n)$ را محاسبه کن $n=p.q$ و $\phi(n)=(p-1).(q-1)$

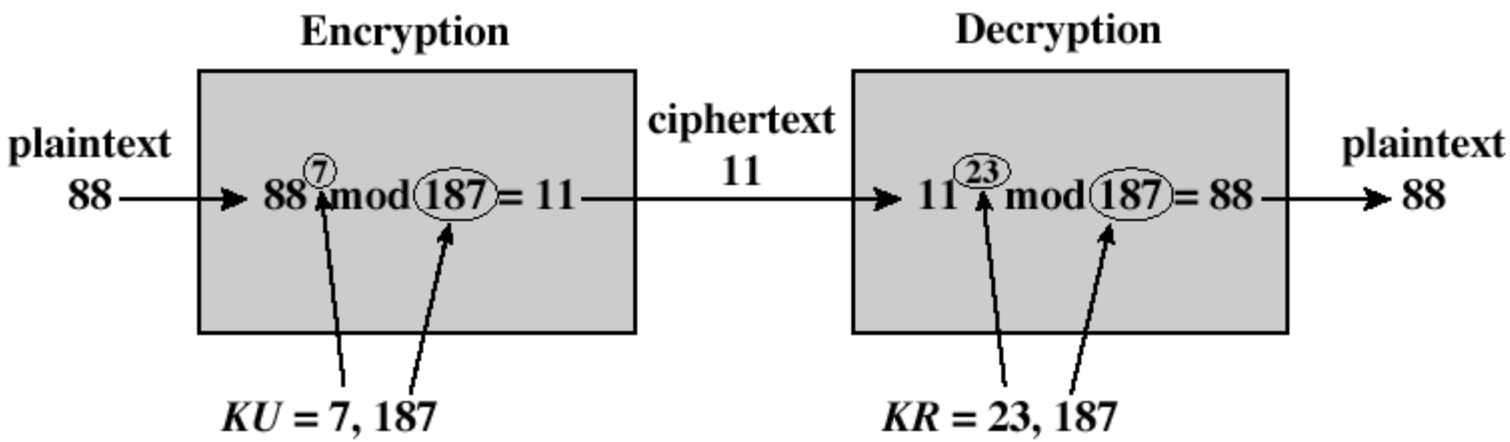
۳– عدد صحیح فرد $e$ کوچکتر از $\phi(n)$ را به گونه ای انتخاب کن که $gcd(e, \phi(n))=1$ باشد.

۴– $d$ را محاسبه کن $d \equiv e^{-1} [\mathrm{mod}\ \phi(n)]$

۵– زوج $PU=(e,n)$ را به عنوان کلید عمومی اعلام کن.

۶– زوج $PR=(d,n)$ را به عنوان کلید خصوصی ذخیره کن.

# RSA–مثال




p = 17, q = 11, n = p.q= 187

$\varphi(n)$ = 16.10 =160, pick e=7, d.e$\equiv$1 [mod $\varphi(n)$]
➔ d = 23

# روشهای کارا برای محاسبه نما

- برای محاسبه $a^b \pmod{n}$ الگوریتم‌های متفاوتی ابداع شده است...
  - فرض کنید $b_k b_{k-1} \ldots b_0$ نمایش مبنای ۲ عدد b باشد.
  - بنابراین خواهیم داشت:

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

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

# الگوریتم توان و ضرب

- بر این مبنا می‌توان الگوریتم زیر را طراحی نمود:

$c \leftarrow 0;\ \ d \leftarrow 1$

for $i \leftarrow k$ downto 0 ⟶ $k$= (size of $b$) - 1

  $c \leftarrow 2.c$ ⟶ $c$ is prefix of $b$

  $d \leftarrow d^2$ mod n

  if $b_i$=1

    then $c \leftarrow c+1$

      $d \leftarrow (d.a)$ mod $n$ ⟶ $d = a^c$ mod $n$

return $d$

# حمله به RSA

- حملات ریاضی
  - تجزیه پیمانه n و در نتیجه محاسبه $\varphi(n)$
- در حال حاضر سختی مساله فوق معادل سختی مساله تجزیه اعداد بزرگ حاصل از ضرب دو عامل اول است.
- الگوریتم‌های مختلفی برای مساله تجزیه ارائه شده است (بهترین آنها LS است).
- در حال حاضر RSA با کلید ۱۰۲۴ تا ۴۰۹۶ بیت امن است.

**Twenty Years of Attacks on the RSA Cryptosystem** 1999, by Dan Boneh

# حمله به RSA

## حمله کانال جانبی

- تاثیرات جانبی اجرای الگوریتم رمزگذاری یا رمزگشایی (مانند میزان توان مصرفی) می‌تواند اطلاعاتی را در مورد کلید افشا نماید.
- مثال: در الگوریتم ارایه شده در اسلایدهای قبل، هرگاه بیت $b_i$ از کلید یک باشد، یک عمل ضرب انجام می‌شود که منجر به مصرف بالاتر می‌شود و زمانی که صفر باشد، مصرف کمتری دیده می‌شود.

## راه‌های مقابله با حملات کانال جانبی

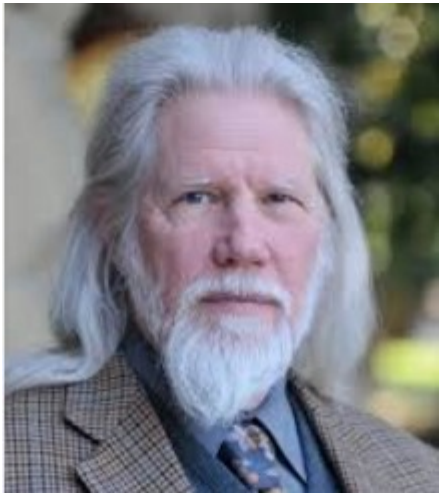
- حذف تاثیرات جانبی
- قرار دادن اعمال اضافی و گمراه کننده جهت تغییر تاثیرات جانبی

# فهرست مطالب

- مبانی رمزنگاری کلید عمومی
- مقایسه با رمزنگاری سنتی و متقارن
- کاربردهای رمزنگاری کلید عمومی
- الگوریتم رمز RSA
- **الگوریتم تبادل کلید دیفی-هلمن**
- الگوریتم رمز الجمل

# الگوریتم دیفی–هلمن

- توسط Diffie و Hellman در سال ۱۹۷۶ ارائه شد.
- برای تبادل کلید مورد استفاده قرار می‌گیرد.



Bailey Whitfield Diffie
(1944 – )

Martin Edward Hellman
(1945 – )

# الگوریتم دیفی-هلمن

- طرفین بر روی مقادیر q و α (به عنوان پارامترهای عمومی) توافق می‌کنند.
- q یک عدد اول و α یک مولد (primitive root) برای این عدد است.
- امنیت روش مبتنی بر دشواری مسأله لگاریتم گسسته است.
- مسأله لگاریتم گسسته: پیدا کردن $x$ با داشتن مقادیر

$$q, \quad \alpha, \quad y = \alpha^x \, mod \, q$$

عدد صحیح α در بازه [۱, $q$-۱]، **مولد یا ریشه اصلی (primitive root)** عدد اول $q$ است اگر تمام مقادیر $\alpha^x$ (mod $q$)، برای $x$ در بازه [۰, $q$-۲]، متفاوت از یکدیگر باشند.

# الگوریتم دیفی-هلمن

عمومی : α , q

**A**

مقدار تصادفی $X_A$ <q را انتخاب می‌کند

**B**

مقدار تصادفی $X_B$ <q را انتخاب می‌کند

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

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

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

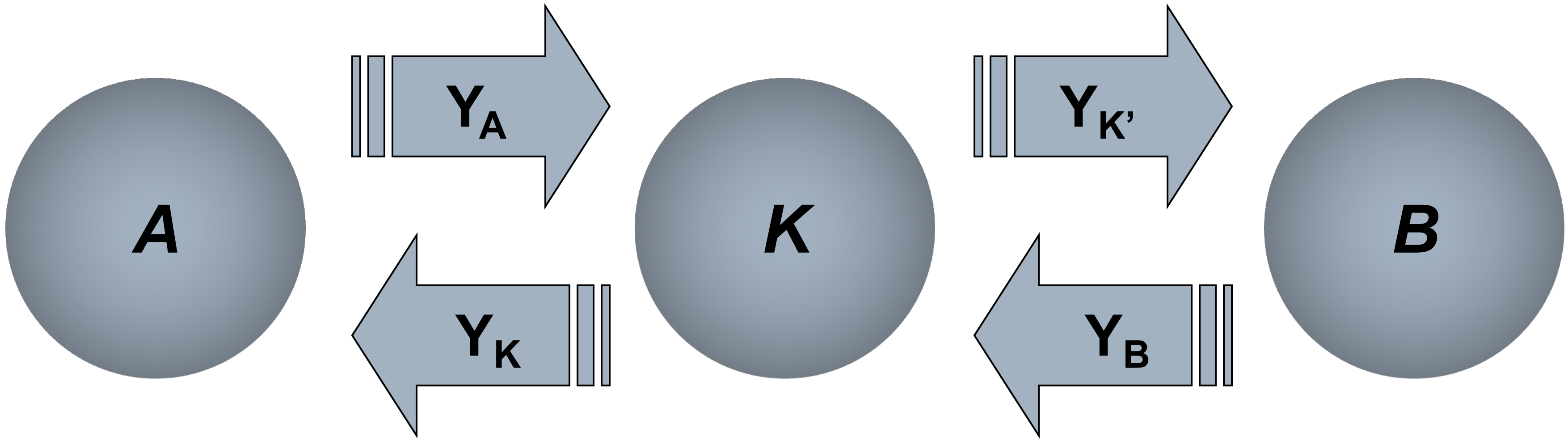
$$K_{AB} = (Y_A)^{X_B} \bmod q$$

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

# حمله مرد میانی

- مهاجم به عنوان کانال ارتباطی میان طرفین عمل می‌کند.
- از نوع حملات فعال محسوب می‌شود.
- الگوریتم دیفی-هلمن را تهدید می‌کند.

# حمله مرد میانی




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

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

A گمان می‌کند کلید $K_1$ را با B به اشتراک گذاشته است.

B گمان می‌کند کلید $K_2$ را با A به اشتراک گذاشته است.

# رفع مشکل تبادل کلید دیفی–هلمن

- طرفین باید قبل از شروع پروتکل، یک کلید طولانی مدت (LTK) را به اشتراک گذاشته باشند.

**LTK:** Long-Term Key

  - LTK می‌تواند متقارن یا نامتقارن باشد.
  - در حالت نامتقارن، طرفین کلید عمومی یکدیگر را دارند.

- **دیفی–هلمن احرازاصالت شده**
  **(ADH) Authenticated Diffie–Hellman**
  - از LTK برای کنترل صحت $\alpha^{X_A}$ و $\alpha^{X_B}$ استفاده می‌شود.
  - در صورت کنترل صحت، مهاجم نمی‌تواند حمله مرد میانی را اجرا کند.

# خاصیت محرمانگی پیشرو (Forward Secrecy)

- گاه به آن PFS هم گفته می‌شود (**P**erfect **F**orward **S**ecrecy).
- **تعریف:** در صورت لو رفتن LTK در زمان T، کلیدهای نشستی که قبل از زمان T تبادل شده‌اند امن بمانند.
- ADH دارای خاصیت PFS است، زیرا:
  - از LTK فقط برای کنترل صحت و نه محرمانگی استفاده می‌شود.
  - محرمانگی کلید نشست وابسته به LTK نیست.

# فهرست مطالب

- مبانی رمزنگاری کلید عمومی
- مقایسه با رمزنگاری سنتی و متقارن
- کاربردهای رمزنگاری کلید عمومی
- الگوریتم رمز RSA
- الگوریتم تبادل کلید دیفی-هلمن
- **الگوریتم رمز الجمل**

# رمز الجمل (ElGamal)

- ابداع توسط الجمل، رمزنگاری مصری-آمریکایی، در سال ۱۹۸۵
  - در ایران بیشتر با نام «الجمال» شناخته می‌شود.
  - الجمل دانشجوی دکترای هلمن در دانشگاه استنفورد بود.

- امنیت رمز الجمل مبتنی بر دشواری لگاریتم گسسته

طاهر الجمل
(۱۹۵۵ – )

# تولید کلید الجمل

- انتخاب پارامترهای عمومی $q$ و $\alpha$
- انتخاب عدد تصادفی $X_A$ به گونه‌ای که $1< X_A <q-1$
- محاسبه $Y_A = \alpha^{X_A} \bmod q$
- **کلید خصوصی:** $X_A$
- **کلید عمومی:** $\{q, \alpha, Y_A\}$

# رمزگذاری و رمزگشایی الجمل

- **رمزگذاری** پیام $M$ که در آن $0 \leq M \leq q-1$
  - انتخاب عدد تصادفی $r$ از $\mathbb{Z}_q$.
  - تولید کلید یکبار مصرف $k = {Y_A}^r \mod q$
  - رمزگذاری پیام به صورت یک زوج $C=(C_1, C_2)$

$$C_1 = \alpha^r \mod q \qquad C_2 = kM \mod q$$

# رمزگذاری و رمزگشایی الجمل

- **رمزگشایی** $C=(C_1, C_2)$ با استفاده از کلید خصوصی $X_A$:
  - بازیابی کلید یکبار مصرف $k = {C_1}^{X_A} \bmod q$
  - رمزگشایی پیام $M = (C_2 k^{-1}) \bmod q$

# کاربردهای الگوریتم‌های کلید عمومی

| الگوریتم | رمزگذاری/ رمز گشایی | امضاء رقمی | تبادل کلید |
|---|---|---|---|
| RSA | √ | √ | √ |
| Diffie-Hellman | × | × | √ |
| DSS (بعداً معرفی خواهد شد) | × | √ | × |
| ElGamal Encryption | √ | × | √ |

# پایان

## پست الکترونیکی

amini@sharif.edu

kharrazi@sharif.edu

یا ذاالامن والامان

# پیوست

# اثبات درستی RSA

# درستی RSA

- **Chinese Remainder Theorem**
  - If $n_1, n_2, \ldots, n_k$ are pairwise relatively prime and $n = n_1 n_2 \ldots n_k$, then for all integers x and a:
  - $x \equiv a \pmod{n_i}$ for $i = 1, 2, \ldots, k$

    *if and only if*

    $x \equiv a \pmod{n}$

- **Fermat's Theorem**
  - If p is prime, $a^{p-1} \equiv 1 \pmod{p}$

# درستی RSA

- Since *e* and *d* are multiplicative inverses modulo
  $\Phi(n) = (p-1)(q-1)$, So $ed = 1 + k(p-1)(q-1)$

- We prove that $M^{ed} = M \pmod{p}$, for all M
  - If $M \neq 0 \pmod{p}$
    - $M^{ed} = M\,(M^{p-1})^{k(q-1)} \pmod{p}$
    - $= M\,(1)^{k(q-1)} \pmod{p}$
    - $= M \pmod{p}$
  - If $M=0 \pmod{p}$, then $M^{ed} = M \pmod{p}$
- In the same way: $M^{ed} = M \pmod{q}$, for all M
- Thus: $M^{ed} = M \pmod{n}$ based on Chinese remainder theorem