



دانشگاه صنعتی امیرکبیر (پلی تکنیک تهران)

دانشکده مهندسی کامپیوتر و فناوری اطلاعات

# مبانی امنیت اطلاعات

(ترم بهمن ۹۸)

## PKC

نستوه طاهری جوان

[nastoooh@aut.ac.ir](mailto:nastoooh@aut.ac.ir)



## Public Key Cryptography

✓ نقطه ضعف اصلی روش های رمزنگاری متقارن  
نیاز به سِت کردن کلید مشترک و سِرّی

علاوه بر این، هر فرد برای تبادل داده با  $n$  نفر دیگر، باید  $n$  کلید سری و محرمانه را مدیریت کند!

✓ در مقابل روش های نامتقارن، موسوم به رمزنگاری کلید-عمومی، این نیاز اساسی را بر طرف می کنند.

○ در این روش ها، عملیات رمزگذاری با یک کلید انجام می شود و عملیات رمزگشایی با کلیدی دیگر!

○ استنتاج این کلیدها از روی یکدیگر غیر ممکن است.



## Public Key Cryptography

### ✓ روال کار در PKC

#### ○ ساختن کلید

- هر کس که تمایل به دریافت داده محرمانه است، باید یک جفت کلید بسازد. یک کلید با عنوان  $e$  برای رمزگذاری، و یک کلید با عنوان  $d$  برای رمزگشایی.

#### ○ انتشار کلید

- فرد گیرنده، کلید  $e$  را برای همه ارسال می کند. اما کلید  $d$  را باید محرمانه نزد خود نگهدارد.
- به همین دلیل به  $d$ ، کلید خصوصی و به  $e$  کلید عمومی گویند.

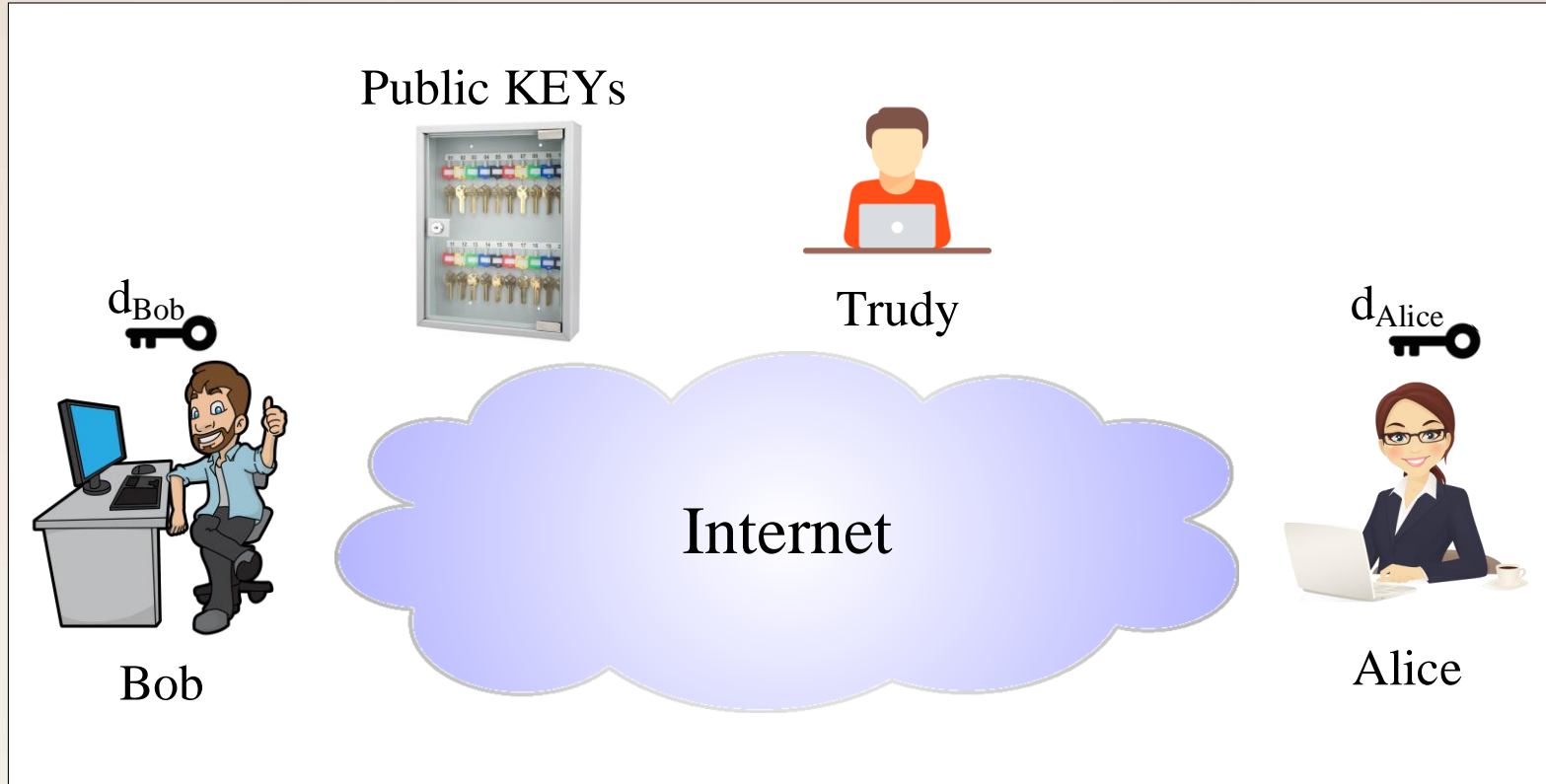
#### ○ رمزگذاری

- فرستنده داده محرمانه برای این گیرنده خاص، کلید عمومی گیرنده را پیدا کرده (یا از خودش دریافت می کند) و داده را با آن رمز کرده و ارسال می کند.
- داده ای که با کلید عمومی گیرنده رمز شده باشد، فقط با کلید خصوصی اش گشوده خواهد شد.



# Public Key Cryptography

روال کار در PKC ✓



شکل برای توضیح شفاهی



## Public Key Cryptography

✓ یادآوری و نتیجه گیری دو اصل اساسی در PKC

○ اصل اول: هر داده ای که با  $e$  یک فرد رمز شده باشد، تنها با  $d$  همان فرد گشوده خواهد شد.

○ اصل دوم:  $e$  و  $d$  از روی یکدیگر قابل استخراج نیستند.  
• با داشتن  $e$ ، نمی توان  $d$  را محاسبه کرد.

پس هر کسی باید یک کلید عمومی و یک کلید خصوصی داشته باشد، کلید عمومی را در اختیار همگان قرار می دهد تا دیگران به کمک آن داده های ارسالی برای وی را رمز کنند، و یک کلید را به صورت محرمانه و برای رمزگشایی داده های دریافتی استفاده می کند.



## Public Key Cryptography

✓ مزایای اصلی PKC در برابر رمزنگاری متقارن

- عدم نیاز به کانال امن جهت تبادل کلید
- هر فرد فقط در محرمانه نگه داشتن یک کلید کوشا باشد
- پشتیبانی از امضای دیجیتال (مباحث آینده درس)



## Public Key Cryptography

✓ معایب اصلی PKC در برابر رمزنگاری متقارن

- پیچیدگی و زمان بر بودن محاسبات رمزنگاری (در حد چند ده هزار برابر)
- در عمل، معمولا ciphertext از plaintext هم طول نیستند.
- استفاده از کلیدهای بسیار بزرگتر در عمل
- نیاز به اعتبار سنجی کلیدهای عمومی (مهم)



## Public Key Cryptography

✓ برخی از الگوریتم های رده PKC

- ElGamal (الجمال)
- کوله پشتی مرکل-هلمن
- کرامر-شاوپ
- ...
- و RSA (که به عنوان معروف ترین الگوریتم، با جزئیات کامل بررسی خواهیم کرد)





# RSA

## الگوریتم RSA (Rivest, Shamir, Adleman) ✓

○ معرفی در سال ۱۹۷۷

○ معروف ترین و پرکاربردترین الگوریتم رمزنگاری کلید عمومی (با اختلاف زیاد)

○ استفاده شده در:

- مستر کارت

- ویزا کارت

- پروتکل SET

- پروتکل SSL

- استفاده موثری در انواع امضاها دیجیتالی

- و ...





# RSA

## الگوریتم RSA ✓

○ ساخت کلید:

- انتخاب دو عدد اول بسیار بزرگ، با عنوان  $p$  و  $q$
- محاسبه  $n$  و  $z$  از روی  $p$  و  $q$

$$n = p * q$$

$$z = (p-1) * (q-1)$$

- انتخاب عدد  $d$  به نحوی که نسبت به  $z$  اول باشد. (هیچ عامل مشترکی نداشته باشند)
- انتخاب  $e$  به گونه ای که عبارت زیر صادق باشد

$$(e * d) \bmod z = 1$$

حال:

- زوج  $(e, n)$  کلید عمومی است.
- زوج  $(d, n)$  کلید خصوصی است.



## RSA

### ✓ الگوریتم RSA

#### ○ رمزگذاری:

- متن را به بلاک های مساوی با عنوان  $P$  تقسیم می کنیم.
- اندازه بلاک ها باید حداکثر  $k$  باشد.  $K$  بزرگترین عددی است که  $2^k < n$  که در صدق می کند.
- فرمول رمزنگاری هر بلاک به صورت زیر است:

$$C = P^e \text{ mod } n$$

#### ○ رمزگشایی

- فرمول رمزگشایی به صورت زیر است:

$$P = C^d \text{ mod } n$$



## RSA

✓ مثال از الگوریتم RSA

○ ساخت کلید:

- انتخاب دو عدد اول:  $p=3$  و  $q=11$
- محاسبه  $n=33$  و  $z=20$
- انتخاب  $d=7$  که نسبت به  $z=20$  اول است
- انتخاب  $e=3$  که در واقع  $7*3 \bmod 20 = 1$

➤ کلید عمومی ( $e=3, n=33$ )

➤ کلید خصوصی ( $d=7, n=33$ )



## RSA

✓ مثال از الگوریتم RSA

○ رمز گذاری

- فرض کنیم یکی از بلاک های داده اصلی برابر ۱۹ است. پس داریم:

$$C = 19^3 \bmod 33 = 28$$

○ رمزگشایی

- داریم  $C=28$ ، پس:

$$P = 28^7 \bmod 33 = 19$$



## RSA

✓ و اما ترودی!

○ ترودی زوج  $(e, n)$  را در اختیار دارد.

○ نیاز به محاسبه و به دست آوردن  $d$  دارد.

- ابتدا باید  $n$  را به عوامل اولش تجزیه کند. که با توجه به ماهیت  $n$ ، متأسفانه پاسخ آن یکتاست که همان  $p$  و  $q$  ابتدایی خواهد بود!
- بر اساس  $p$  و  $q$ ، می تواند به راحتی  $Z$  را محاسبه کند.
- بر اساس  $Z$  می تواند  $d$  را به دست آورد.
- تمام!

○ نکته: هنوز الگوریتم کارآمدی برای تجزیه اعداد بزرگ به عوامل اولشان پیدا نشده است. تنها راه: جستجو و آزمون!



## RSA

### ✓ جمع بندی RSA

- پس از گذشت ۴ دهه، هنوز RSA محبوب و پر کاربرد است.
- با کشف احتمالی یک الگوریتم کارآمد برای تجزیه اعداد به عوامل اولشان، کار RSA یکسره خواهد شد.
- تا آن زمان کافیست، همزمان با افزایش قدرت محاسباتی دنیا، فقط طول کلید را افزایش داد.
- برای درک مبانی نظری آن، مبحث لگاریتم گسسته، قضایای فرما و اوپلر و قضایای مربوط به اعداد اول را مرور کنید.

✓ الگوریتم های رمزنگاری نامتقارن دیگری نیز پیشنهاد شده اند، که هر یک بنابه دلایلی توفیق RSA را کسب نکرده اند.



## منابع

[1] William Stallings, “Cryptography and Network Security,” 7<sup>th</sup> ed.





پایان