



آزمایشگاه امنیت داده و شبکه  
<http://dnsl.ce.sharif.edu>



دانشگاه صنعتی شریف  
دانشکده مهندسی کامپیوتر

# درس ۷: امضای دیجیتال و زیرساخت کلید عمومی

محمد صادق دوستی

## □ مبانی امضای دیجیتال

□ امضای دیجیتال RSA و الجمل

□ زیرساخت کلید عمومی (PKI)

☞ مبانی PKI

☞ گواهی دیجیتال و مدیریت آن

☞ مؤلفه‌های PKI

☞ معماری PKI، رویه‌ها و خط‌مشی‌ها

□ چرا به امضای دیجیتال نیاز داریم؟ زیرا در صورت استفاده از رمز  
**مقارن:**

➡ **جعل توسط گیرنده:** گیرنده می تواند یک پیام جعلی را بسازد  
(با استفاده از کلید توافق شده) و آنرا به فرستنده نسبت دهد!

➡ **انکار توسط فرستنده:** فرستنده می تواند سناریوی فوق را بهانه  
قرار دهد و پیام فرستاده شده را منکر شود!

□ امکان تصدیق هویت فرستنده (و در صورت نیاز زمان و تاریخ ارسال)

□ تضمین عدم تغییر محتویات پیام

□ امکان تصدیق توسط طرف سوم (در صورت بروز اختلاف)

# نیازمندی‌های امضای دیجیتال

- ❑ رشته بیتی تولید شده وابسته به پیام اصلی باشد.
- ❑ از اطلاعات منحصر به فرستنده استفاده شود (جلوگیری از جعل و انکار)
- ❑ امضای دیجیتال صرفاً بر رمزنگاری نامتقارن (کلید عمومی) مبتنی است. در واقع برای پشتیبانی از سرویس عدم انکار، فرستنده و گیرنده نمی‌توانند از یک کلید مشترک استفاده کنند.
- ❑ به سادگی محاسبه شود و فضای کمی برای ذخیره نیاز داشته باشد.
- ❑ تشخیص و تأیید (verify) آن آسان باشد.
- ❑ جعل آن از نظر محاسباتی دست نیافتنی باشد.

## □ الگوریتم تولید کلید (Key Generation)

➡ به صورت تصادفی یک زوج کلید عمومی تولید می‌کند.

## □ الگوریتم امضا (Sign)

➡ پیام و کلید خصوصی فرستنده را به عنوان ورودی می‌گیرد و امضا را تولید می‌کند.

## □ الگوریتم تأیید امضا (Verification)

➡ پیام، امضا و کلید عمومی فرستنده را به عنوان ورودی می‌گیرد و تأیید (۱) یا عدم تأیید (۰) امضا را به عنوان خروجی برمی‌گرداند.

□ مبانی امضای دیجیتال

□ امضای دیجیتال **RSA** و الجمل

□ زیرساخت کلید عمومی (PKI)

➔ مبانی PKI

➔ گواهی دیجیتال و مدیریت آن

➔ مؤلفه‌های PKI

➔ معماری PKI، رویه‌ها و خط‌مشی‌ها

□ فرض کنیم  $(n, e)$  کلید عمومی و  $(n, d)$  کلید خصوصی RSA باشد.

□ امضای پیام  $m \in \mathbb{Z}_n$  با کلید خصوصی:

$$\sigma = m^d \pmod{n}$$

□ واریسی امضا با کلید عمومی:

$$m \stackrel{?}{=} \sigma^e \pmod{n}$$



□ امضای RSA سنتی به سادگی قابل جعل است.

👉 امضای پیام ° همواره ° و امضای پیام ۱ همواره ۱ است.

👉 با داشتن امضای  $\sigma_1$  و  $\sigma_2$  روی پیام‌های  $m_1$  و  $m_2$ ، می‌توان یک امضای جدید جعل کرد:

$$m = m_1 \times m_2 \pmod{n}$$

$$\sigma = \sigma_1 \times \sigma_2 \pmod{n}$$

□ راهکار: امضای چکیده پیام به جای خود پیام

👉 چکیده باید با یک تابع درهم‌ساز CR محاسبه شود. (چرا؟)

# امضای RSA با کمک توابع درهم‌ساز

□ امضای پیام  $m \in \mathbb{Z}_n$  با کلید خصوصی:

$$\sigma = (H(m))^d \pmod{n}$$

□ واریسی امضا با کلید عمومی:

$$H(m) \stackrel{?}{=} \sigma^e \pmod{n}$$

□ مشکل: طول خروجی توابع درهم‌ساز معمولاً خیلی کوچکتر از دامنه تابع RSA است.

➡ راهکار ۱: Pad کردن خروجی تابع درهم‌ساز

➡ راهکار ۲: استفاده از Full-Domain Hash (FDH)

□ RSA به گونه‌ای است که با یک زوج کلید می‌توان هم امضا و هم رمز نمود:

➡ به کارگیری  $(n, e)$  برای رمزگذاری و  $(n, d)$  برای رمزگشایی

➡ به کارگیری  $(n, d)$  برای امضا و  $(n, e)$  برای واری امضا

□ این کار اشتباه است و از لحاظ امنیتی مخاطراتی دارد.

□ باید دو زوج کلید RSA تولید شود: یک زوج برای رمزنگاری و زوج دیگر برای امضا.

# امضا با روش الجمل – تولید کلید

□ انتخاب عدد اول بزرگ  $p$

□ انتخاب  $g \in \mathbb{Z}_p^*$  به گونه‌ای که  $|\langle g \rangle_p| = q$

☞  $q$  باید اول و بزرگ باشد.

□ انتخاب عدد تصادفی  $\alpha$  از  $\mathbb{Z}_q$  و محاسبه  $h = g^\alpha \pmod{p}$

□  $p, q$  و پارامترهای عمومی (همه مقادیر آنها را می‌دانند).

□  $\alpha$  کلید خصوصی و  $h$  کلید عمومی.

# امضا با روش الجمل – امضای پیام

□ امضای پیام  $m \in \mathbb{Z}_q$ :

☞ انتخاب  $k$  به تصادف از  $\mathbb{Z}_q^*$  و محاسبه:

$$r \equiv g^k \pmod{p}$$

$$s \equiv (m - \alpha r)k^{-1} \pmod{q}$$

□ امضای پیام  $m$  عبارت است از زوج  $(r, s)$ .

# امضا با روش الجمل – واریسی صحت امضا

□ واریسی امضا  $(r, s)$  با داشتن کلید عمومی  $h$  و پیام  $m$  :

$$g^m \stackrel{?}{=} h^r r^s \pmod{p}$$

□ چرا روش واریسی درست کار می کند؟

👉 بر اساس روش تولید امضا داریم:

$$m \equiv \alpha r + sk \pmod{q}$$

👉 بنابراین:

$$g^m \equiv g^{\alpha r + ks} \equiv (g^\alpha)^r (g^k)^s \equiv h^r r^s \pmod{p}$$

# معایب امضای الجمل – ۱

□ همانند Textbook RSA، امضای الجمل نیز از توابع درهم‌ساز استفاده نمی‌کند.

□ امکان تولید امضاهای معتبر حتی بدون داشتن کلید خصوصی:

👉 **جعل با یک پارامتر:** مقدار دلخواه  $x \in \mathbb{Z}_q$  را در نظر بگیرید.

$$r \equiv g^x \times h \pmod{p}$$

$$s \equiv -r \pmod{q}$$

زوج  $(r, s)$  یک امضای معتبر برای پیام  $m \equiv xs \pmod{q}$  است.

$$g^m \equiv g^{xs} \pmod{p}$$

$$h^r r^s \equiv h^{-s} \times (g^x \times h)^s \equiv g^{xs} \pmod{p}$$

$$\rightarrow g^m \equiv h^r r^s \pmod{p}$$



## معایب امضای الجمل – ۲

□ امکان تولید امضاهای معتبر حتی بدون داشتن کلید خصوصی:

👉 **جعل با دو پارامتر:** مقادیر دلخواه  $x \in \mathbb{Z}_q$  و  $y \in \mathbb{Z}_q^*$  را در نظر بگیرید.

$$r \equiv g^x \times h^y \pmod{p}$$

$$s \equiv -r \times y^{-1} \pmod{q}$$

زوج  $(r, s)$  یک امضای معتبر برای پیام  $m \equiv xs \pmod{q}$  است.

$$g^m \equiv g^{xs} \pmod{p}$$

$$h^r r^s \equiv h^r \times (g^x \times h^y)^s \equiv g^{xs} \times h^{\overbrace{sy}^{sy = -r} + r} \equiv g^{xs} \pmod{p}$$

$$\rightarrow g^m \equiv h^r r^s \pmod{p}$$

□ همانند امضای RSA، بهترین راهکار این است که به جای امضا کردن پیام  $m$ ، مقدار  $H(m)$  را امضا کنیم.

□ در اینجا  $H$  یک تابع درهم‌ساز CR است.

□ به طور کلی، استفاده از  $H(m)$  به جای  $m$  یک راهکار جا افتاده در امضای دیجیتال است.

# خطای مهلك در امضای الجمل

□ اگر مقدار  $r$  تکراری باشد، می‌توان کلید خصوصی الجمل را یافت.

→ حتی در نسخه‌ای از الجمل که چکیده پیام را امضا می‌کند.

□ پیام‌های  $m_1$  و  $m_2$  با امضاهاى  $(r, s_1)$  و  $(r, s_2)$  که در آن:

$$s_i \equiv (H(m_i) - \alpha r)k^{-1} \pmod{q}$$

□ به شرط  $s_2 \neq s_1$  و  $\text{GCD}(r, q) = 1$  داریم:

$$k \equiv (H(m_1) - H(m_2))(s_1 - s_2)^{-1} \pmod{q}$$

$$\alpha \equiv (H(m_1) - k \times s_1)r^{-1} \pmod{q}$$

❑ DSS: Digital Signature Standard

❑ استاندارد شده توسط NIST FIPS 186

❑ مبتنی بر امضای الجمل

❑ RSA Digital Signature: استاندارد شده توسط

ISO 9776 

ANSI X9.31 

CCITT X.509 

- مبانی امضای دیجیتال
- امضای دیجیتال RSA و الجمل
- زیرساخت کلید عمومی (PKI)

➡ مبانی PKI

➡ گواهی دیجیتال و مدیریت آن

➡ مؤلفه‌های PKI

➡ معماری PKI، رویه‌ها و خط‌مشی‌ها

□ نکته اصلی در رمزنگاری نامتقارن:

☞ «هویت صاحب یک کلید عمومی چیست؟»

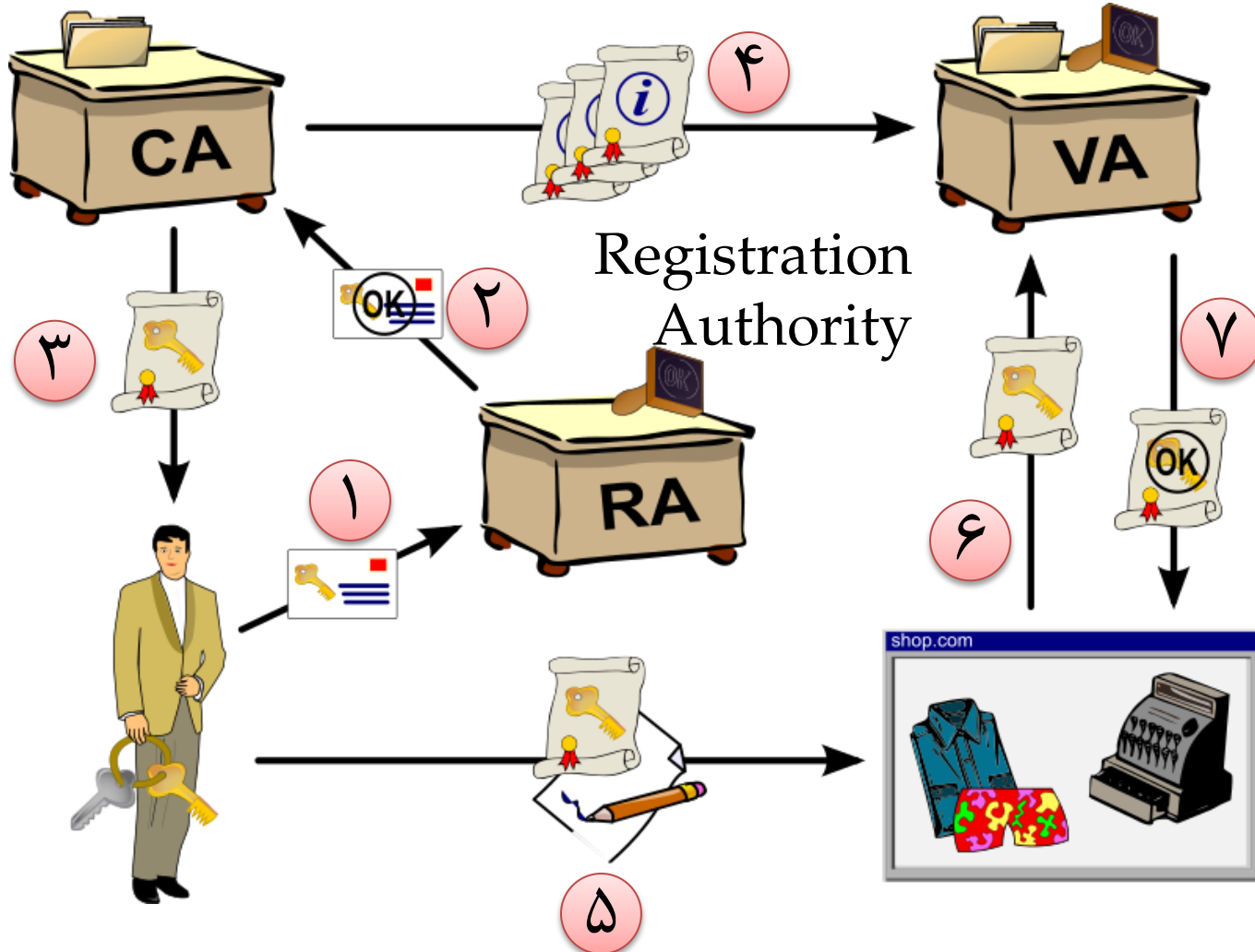
□ به عبارت دیگر، آیا یک کلید عمومی واقعاً به فردی که ادعا می‌کند تعلق دارد؟

□ برای هر کلید عمومی باید یک گواهی از یک مرجع معتبر وجود داشته باشد که متضمن تعلق آن به یک فرد باشد.

□ بنابراین نیاز به زیرساختی برای صدور گواهی و وارسی آن داریم که زیرساخت کلید عمومی (PKI) نام دارد.

Certification Authority

Validation Authority





□ مبانی امضای دیجیتال

□ امضای دیجیتال RSA و الجمل

□ زیرساخت کلید عمومی (PKI)

☞ مبانی PKI

☞ **گواهی دیجیتال و مدیریت آن**

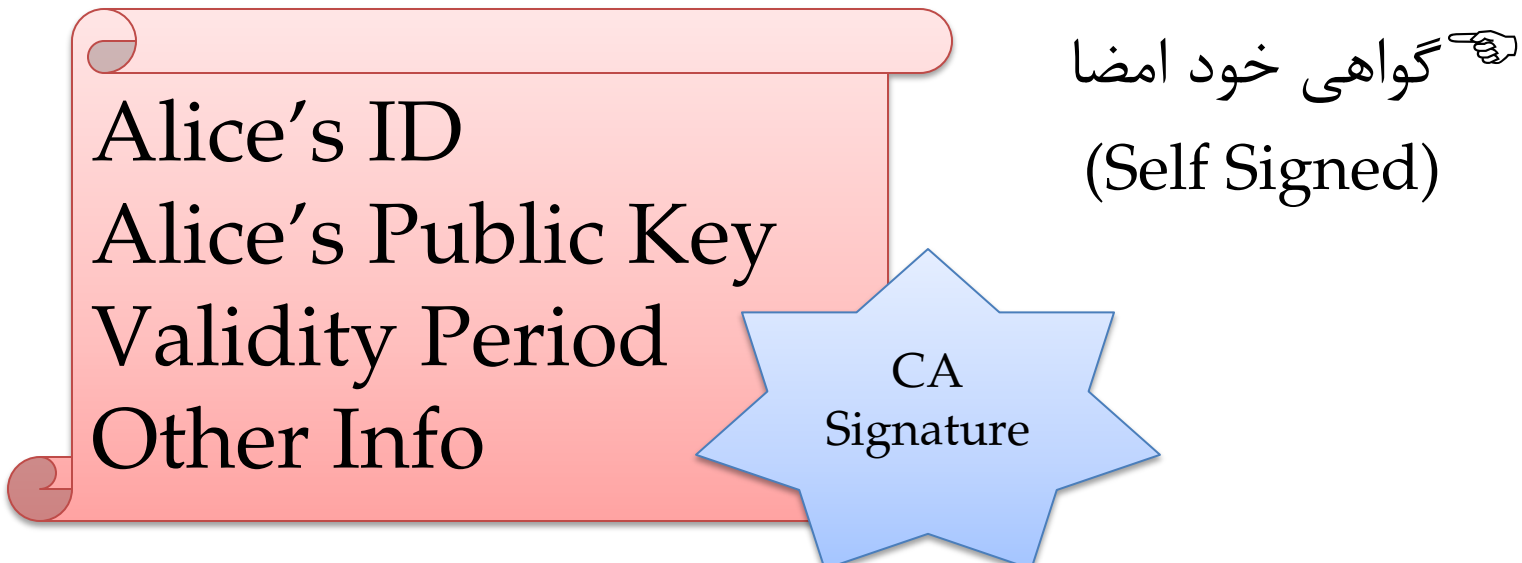
☞ مؤلفه‌های PKI

☞ معماری PKI، رویه‌ها و خط‌مشی‌ها

□ گواهی (Certificate) سندی رسمی است برای تضمین تعلق کلید عمومی به یک شناسه.

□ گواهی به وسیله یک مرکز مطمئن (CA) امضا شده است.

□ کلید عمومی CAهای مطمئن در سیستم عامل وجود دارد.



Alice's ID  
Alice's Public Key  
Validity Period  
Other Info

The diagram shows a self-signed certificate structure. On the left, a pink scroll contains the text: Alice's ID, Alice's Public Key, Validity Period, and Other Info. To the right of the scroll is a blue star-shaped box containing the text: CA Signature. Further to the right, the text 'گواهی خود امضا (Self Signed)' is written with a hand icon pointing to it.

گواهی خود امضا  
(Self Signed)

CA  
Signature

# مثال: گواهی‌ها در سیستم عامل ویندوز

certmgr - [Certificates - Current User\Trusted Root Certification Authorities\Certificates]

File Action View Help

Certificates - Current User

- Personal
- Trusted Root Certification Authorities
  - Certificates
- Enterprise Trust
- Intermediate Certification Authorities
- Active Directory User Objects
- Trusted Publishers
- Untrusted Certificates
- Third-Party Root Certification Authorities
- Trusted People
- Client Authentication Issuers
- Smart Card Trusted Roots

Issued To	Issued By	Expiration Date	Intended Purpose
QuoVadis Root Certification...	QuoVadis Root Certificatio...	3/17/2021	Server Authentici...
SecureTrust CA	SecureTrust CA	1/1/2030	Server Authentici...
Starfield Class 2 Certific...	Starfield Class 2 Certificati...	6/29/2034	Server Authentici...
Starfield Root Certificat...	Starfield Root Certificate A...	1/1/2038	Server Authentici...
StartCom Certification ...	StartCom Certification Aut...	9/18/2036	Server Authentici...
Symantec Enterprise M...	Symantec Enterprise Mobil...	3/15/2032	Code Signing
Thawte Premium Server...	Thawte Premium Server CA	1/1/2021	Server Authentici...
thawte Primary Root CA	thawte Primary Root CA	7/17/2036	Server Authentici...
thawte Primary Root C...	thawte Primary Root CA - ...	12/2/2037	Server Authentici...
Thawte Server CA	Thawte Server CA	1/1/2021	Server Authentici...
Thawte Timestamping ...	Thawte Timestamping CA	1/1/2021	Time Stamping
TÜRKTRUST Elektronik ...	TÜRKTRUST Elektronik Sert...	12/22/2017	Server Authentici...
UTN-USERFirst-Object	UTN-USERFirst-Object	7/9/2019	Encrypting File S...
VeriSign Class 3 Public ...	VeriSign Class 3 Public Pri...	7/17/2036	Server Authentici...
VeriSign Universal Root ...	VeriSign Universal Root Cer...	12/2/2037	Server Authentici...

Trusted Root Certification Authorities store contains 49 certificates.

□ صحت گواهی به راحتی قابل کنترل است. هر تغییری در آن به سادگی تشخیص داده می شود.

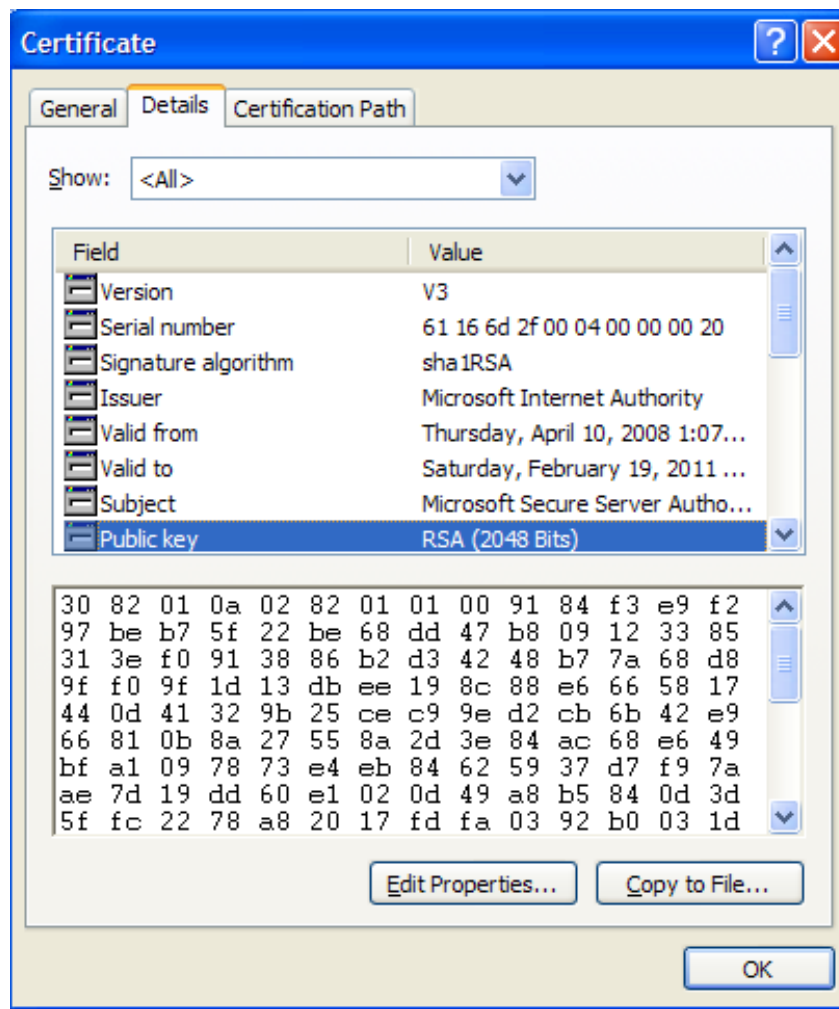
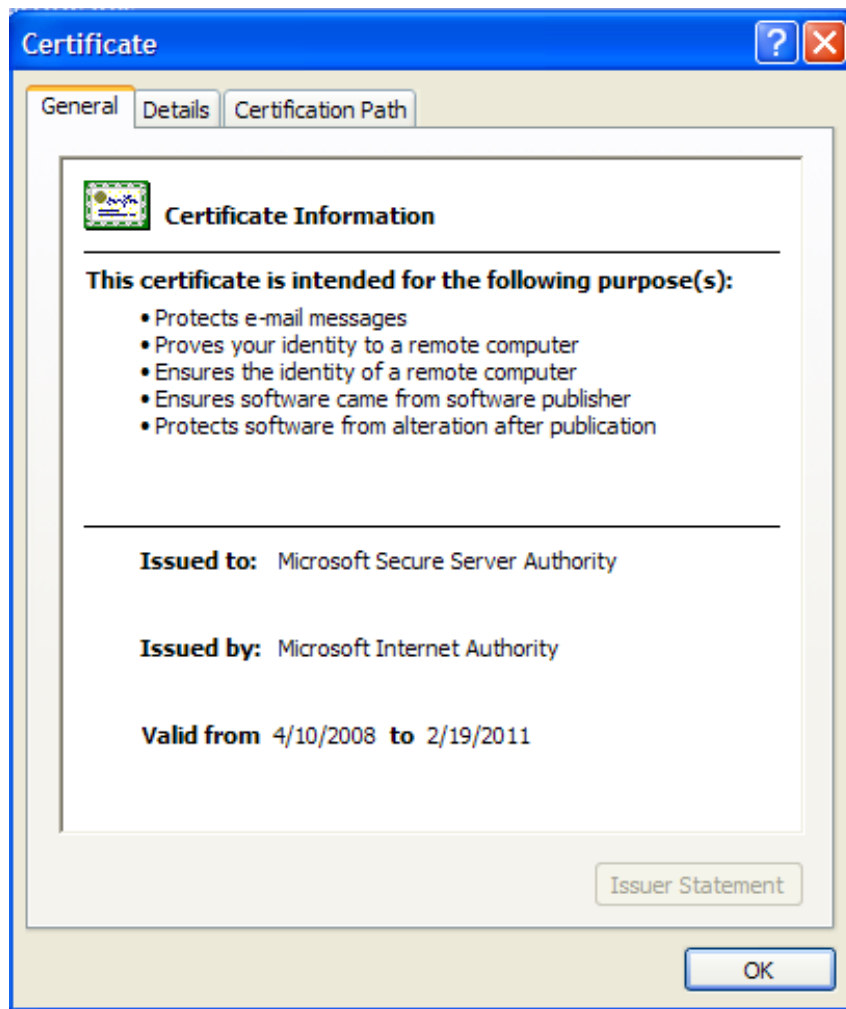
☞ به وسیله واری امضای دیجیتال روی آنها

☞ استثنا: گواهی های خود امضا ← نیازمند نگهداری امن

□ گواهی به شکل رمز نشده ارسال و ذخیره می شود.

□ برای واری گواهی به کلید عمومی CA نیاز داریم.

# نمونه گواهی کلید عمومی در ویندوز



□ دلایل ابطال گواهی:

☞ تغییر مشخصات موجودیتی که برایش گواهی صادر شده است؛

☞ گم شدن و یا لو رفتن کلید خصوصی موجودیت؛

☞ عدم تبعیت از سیاستهای مرکز صدور گواهی توسط موجودیت.

□ نیاز به تغییر کلید عمومی، ضرورت اطمینان از اطلاع همه دنیا از این تغییر.

## □ CRL: Certificate Revocation List

□ CA به طور دوره‌ای لیستی از گواهی‌های باطل شده را صادر می‌کند. برای بررسی اعتبار گواهی، لازم است CRL‌های صادر شده واریسی شوند.

## □ OCSP: Online Certificate Status Protocol

□ CA یک سرویس برخط ارائه می‌کند که می‌توان به کمک آن، معتبر بودن یا نبودن یک گواهی را بررسی کرد.

□ تاریخ ابطال، شماره سریال گواهی‌های نامعتبر، به همراه امضای CA در لیست گواهی نامعتبر (CRL) وجود دارد.

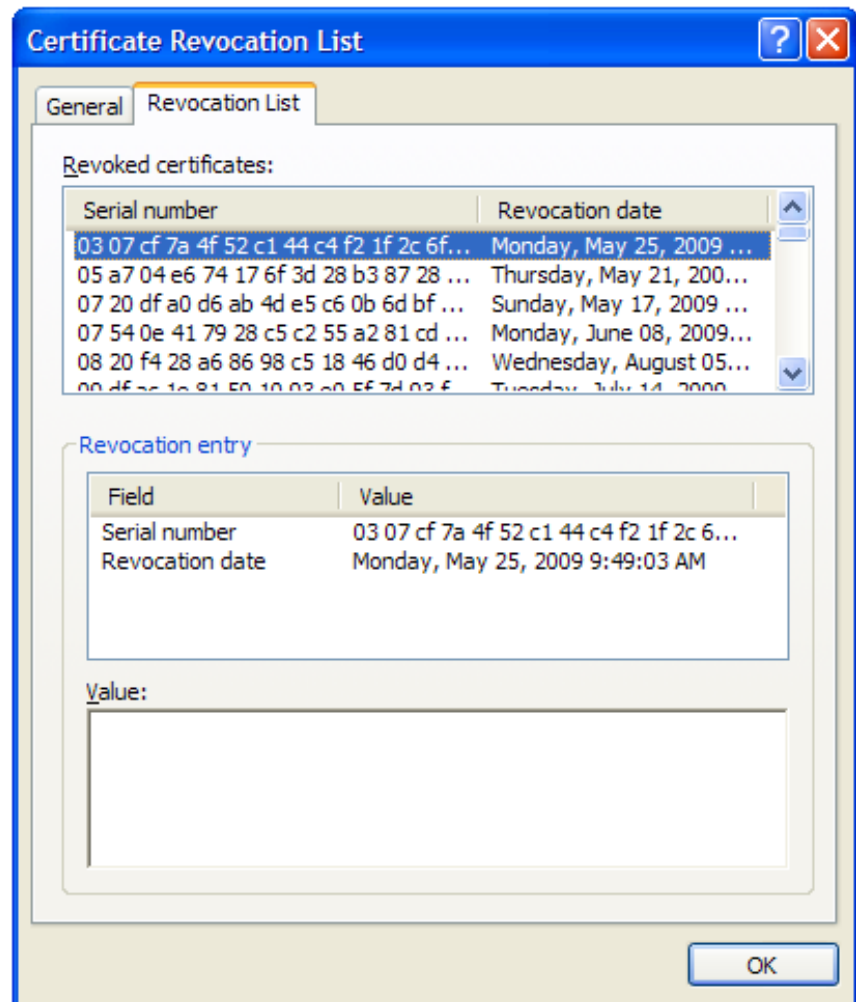
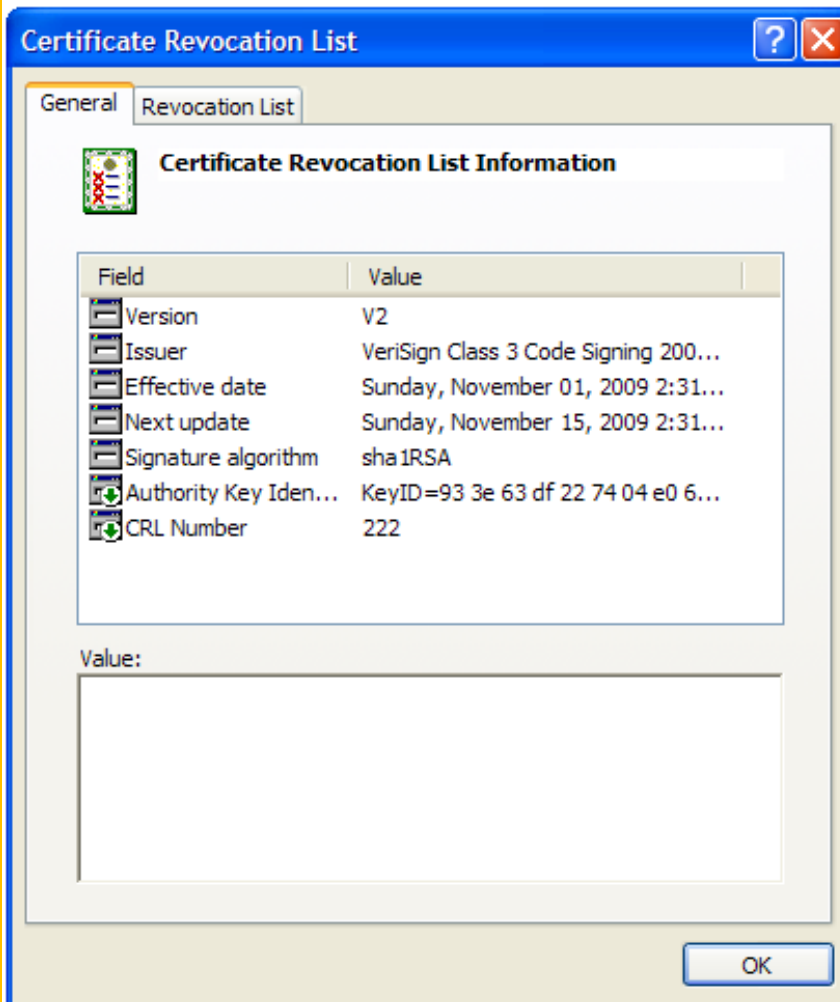
□ انواع CRL:

👉 **Full CRL**: در دوره‌های زمانی مشخص یک CA لیست کامل گواهی‌های نامعتبر را منتشر می‌کند.

👉 **Delta CRL**: اختلاف اخیرترین بروز رسانی و CRL جدید.



# عدم اعتبار گواهی – CRL



❑ به کارگزار OCSP، اصطلاحاً OCSP Responder گفته می شود.

❑ مزیت نسبت به CRL: به دلیل برخط بودن اطمینان بیشتری را از وضعیت فعلی گواهی فراهم می نماید.

❑ عیب نسبت به CRL:

➡ کارگزار OCSP می تواند از گواهی هایی که یک فرد استفاده می نماید اطلاع یابد. لذا حریم خصوصی فرد خدشه دار می شود.

➡ کارگزار OCSP باید همواره برخط بوده و کارایی بالایی داشته باشد.

□ کلید خصوصی ممکن است گم شود، که در این صورت داده‌های رمز شده غیر قابل دسترس می‌شوند.

□ سایر دلایل از دست رفتن کلید خصوصی:

👉 فراموشی کلمه رمز کلید خصوصی

👉 گم شدن، دزدیده شدن، و یا خرابی رسانه‌ای که کلیدها روی آن ذخیره شده است.

□ باید مرکزی برای بازیابی کلید وجود داشته باشد.

□ در صورتی که کلید خصوصی مورد استفاده در امضا در اختیار مرکز بازیابی کلید قرار بگیرد، انکارناپذیری خدشه دار می شود.

□ بنابراین به لحاظ نظری بهتر است دو زوج کلید برای هر کاربر وجود داشته باشد:

👉 **زوج کلید امضا:** عدم نیاز به پشتیبان. در صورت از بین رفتن کلید خصوصی، می توان زوج کلید جدیدی تولید کرد.

👉 **زوج کلید رمزنگاری:** نیازمند پشتیبان گیری

❑ نباید کلیدها ابدی باشند. پس باید:

➡ کلیدها در دوره‌های زمانی مشخصی به‌روز شوند.

➡ به‌روز رسانی کلید باید قبل از انقضا صورت پذیرد.

➡ سابقه زوج کلیدهای (رمزنگاری) قبلی را نگه داشت تا داده‌های رمز شده با زوج قبلی قابل رمزگشایی باشند.

➡ در نقطه مقابل، برای به‌روز رسانی کلیدهای امضا باید کاملاً کلید فعلی را نابود کرد!

□ مبانی امضای دیجیتال

□ امضای دیجیتال RSA و الجمل

□ زیرساخت کلید عمومی (PKI)

☞ مبانی PKI

☞ گواهی دیجیتال و مدیریت آن

☞ مؤلفه‌های **PKI**

☞ معماری PKI، رویه‌ها و خط‌مشی‌ها

□ کاربران یا دارندگان گواهی: کاربران انسانی، تجهیزات و هر آنچه که می‌تواند از گواهی استفاده نماید.

□ مرکز گواهی (CA): مسئول تولید، مدیریت، و ابطال گواهی.

□ مرکز ثبت نام (RA): مسئول دریافت درخواست گواهی و کنترل محتوای گواهی و اطمینان از هویت متقاضی.

□ انبار (Repository): توزیع گواهی‌ها و CRLها (حداکثر کارایی و دسترس‌پذیری را لازم دارد).

□ بایگانی (Archive): انبار طولانی‌مدت و امن برای بایگانی اطلاعات.

- به عنوان نماینده مورد اعتماد در PKI است و لذا شخص ثالث معتمد (Trusted Third Party) نامیده می شود.
- مجموعه ای از سخت افزار، نرم افزار، و اپراتورها.
- با دو صفت شناخته می شود: نام و کلید عمومی.



- صدور گواهی کاربران و یا دیگر CAها (تولید و امضا).
- نگهداری وضعیت گواهی‌ها و صدور CRL.
- انتشار گواهی‌ها و CRL موجود.
- نگهداری بایگانی اطلاعات وضعیتی از گواهی‌های صادره منقضی یا ابطال شده، به منظور تعیین اعتبار گواهی‌ها پس از انقضا.

- تأیید اینکه یک موجودیت (دارنده گواهی) کلید خصوصی متناظر با کلید عمومی موجود در گواهی را دارد.
- اگر کلید خصوصی CA لو برود، همه گواهی‌های صادره‌اش در معرض شک است.
- پس اولین وظیفه CA حفاظت از کلید خصوصی خودش است، حتی وقتی در حال پردازش است.
- وظیفه دیگر CA اطمینان از درستی گواهی و درستی ادعای درخواست‌کننده گواهی است.

- RA قبل از ارائه درخواست به CA اطلاعات لازم را جمع‌آوری و کنترل می‌کند: مراجعه شخص، تصدیق هویت.
- ممکن است زوج کلید توسط خود متقاضی تولید شده و فقط کلید عمومی در اختیار RA و CA قرار بگیرد.
- در غیر این صورت RA و (یا CA) می‌تواند زوج کلید لازم را در حضور متقاضی تولید نمایند.

□ استاندارد ITU-T برای PKI

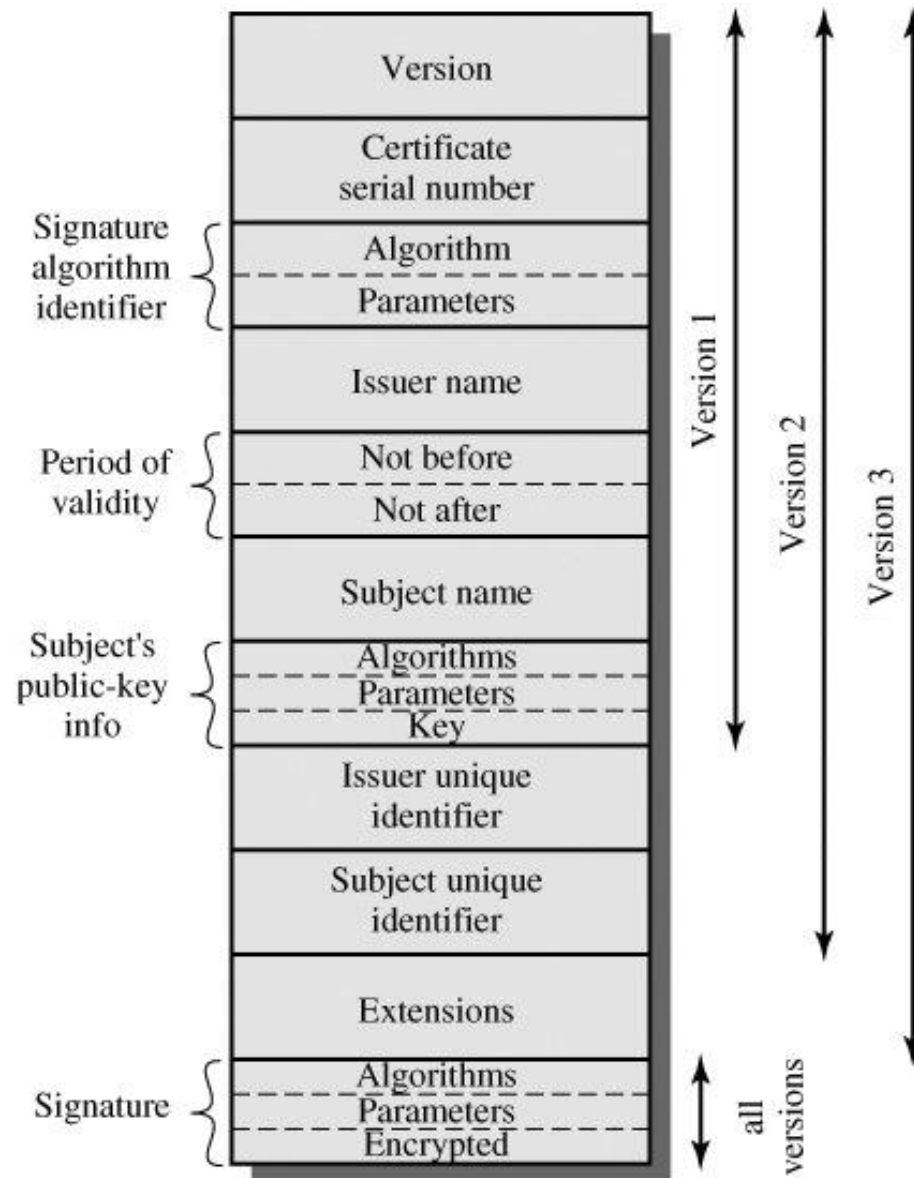
☞ بخشی از توصیه‌های سری X.500

□ گواهی X.509 در SSL/TLS، IPsec، S/MIME و SET استفاده شده است.

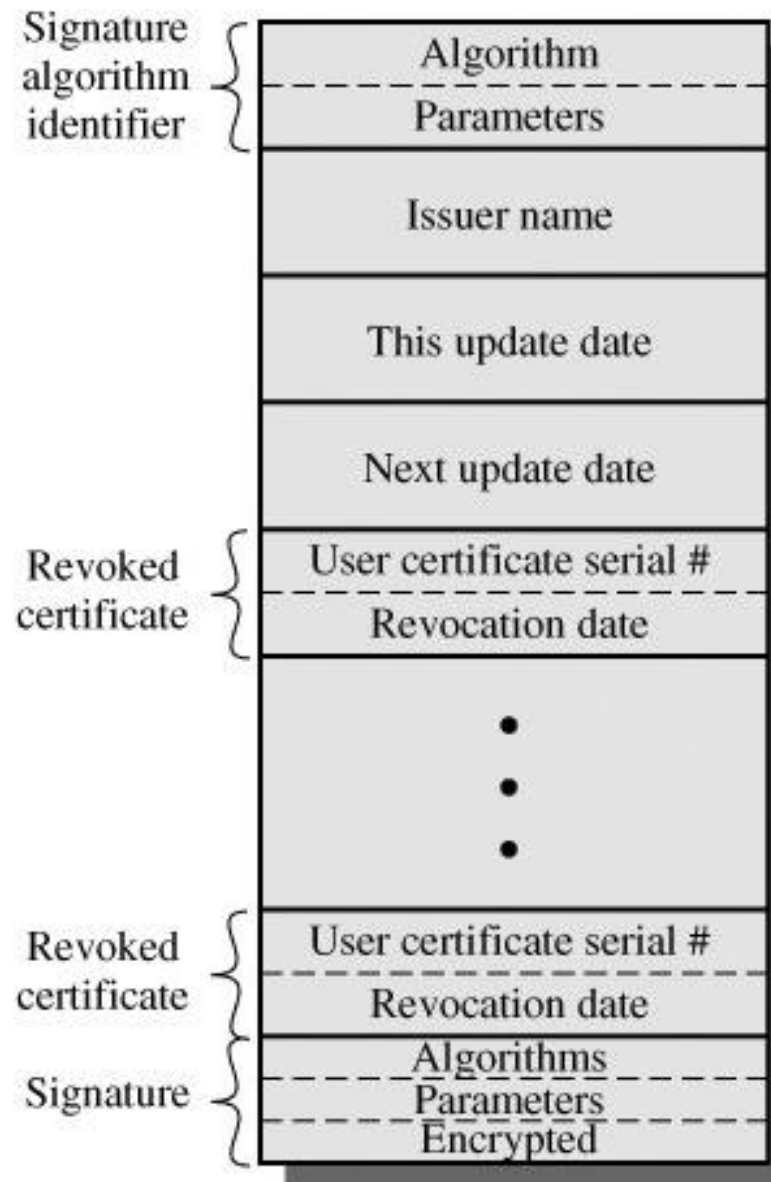
□  $CA \ll A \gg$  به معنای **گواهی** صادره CA برای کاربر A است.

□ همه کاربران در محدوده یک CA: وجود اعتماد مشترک و امکان واریسی گواهی صادره.

# ساختار گواهی دیجیتال X.509



# ساختار CRL در X.509



□ مبانی امضای دیجیتال

□ امضای دیجیتال RSA و الجمل

□ زیرساخت کلید عمومی (PKI)

➔ مبانی PKI

➔ گواهی دیجیتال و مدیریت آن

➔ مؤلفه‌های PKI

➔ معماری PKI، رویه‌ها و خط‌مشی‌ها

❑ مادام که دارندگان گواهی از یک CA گواهی گرفته باشند مسأله ساده است.

❑ معماری ساده PKI

👉 تنها یک CA؛ ایجاد Single Point of Failure

👉 هرگونه اشکال منجر به لطمه دیدن اعتماد و احتمالاً صدور مجدد گواهی‌ها.

❑ وقتی که دارندگان گواهی از CAهای مختلف گواهی گرفته باشند چگونه به هم اعتماد کنند؟

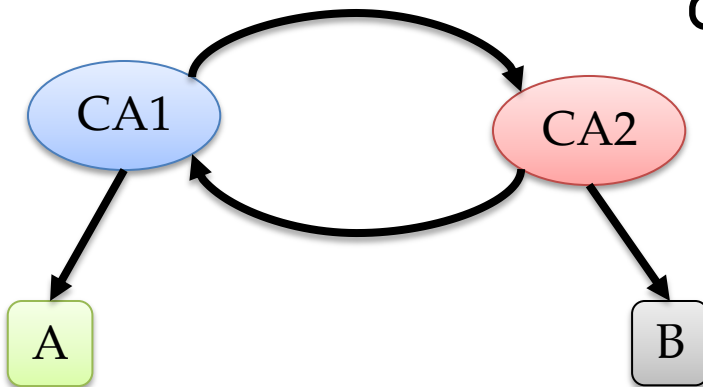


# گواهی ضربدري (Cross-Certificate)

□ گواهی ضربدري، گواهی‌ای است که یک CA برای CA دیگر صادر می‌کند تا گواهی‌های صادره توسط CA دوم توسط کاربران CA اول معتبر شناخته شوند.

□ با فرض صدور گواهی A و B توسط دو CA مختلف CA1 و CA2

CA1 <<CA2>>, CA2 <<B>>  
CA2 <<CA1>>, CA1 <<A>>



□ سه معماری مختلف برای PKI بزرگ

👉 سلسله مراتبی: در یک ساختار درختی

👉 توری (Mesh): ارتباط کامل ضربدري CAها با یکدیگر

👉 ترکیبی از دو مدل فوق: چند سلسله مراتب از CAها که ریشه آنها با یکدیگر ارتباط ضربدري دارند.

# مدل سلسله مراتبی

□ ساختار درختی از CAها

□ CA ریشه و مجموعه‌ای CA میانی

□ مزایا:

☞ توزیع کار و کاهش ریسک

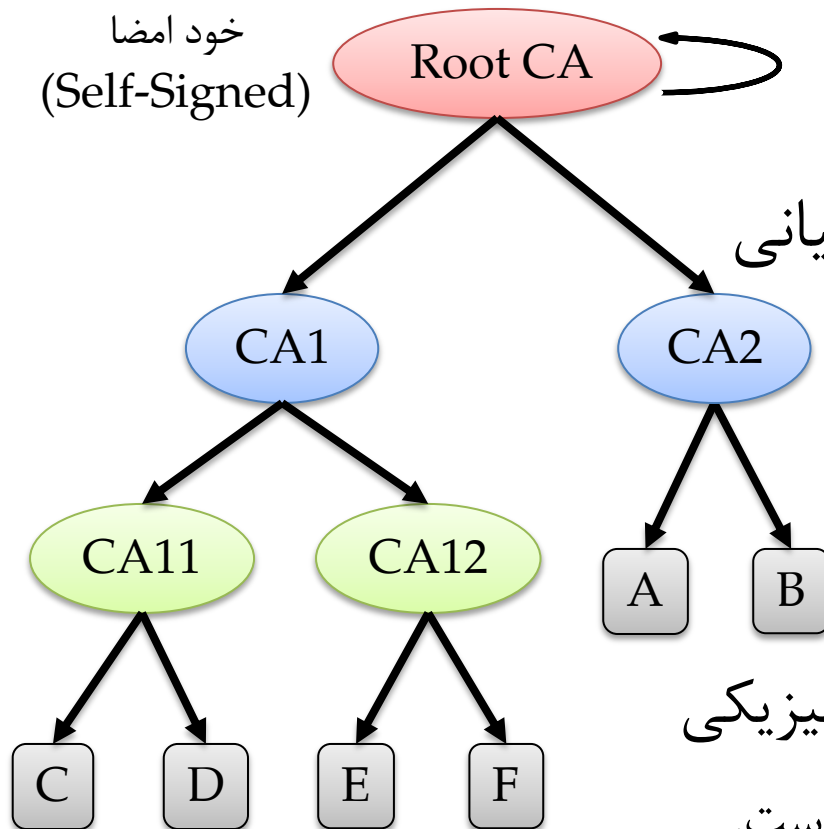
☞ کاهش هزینه برقراری امنیت فیزیکی

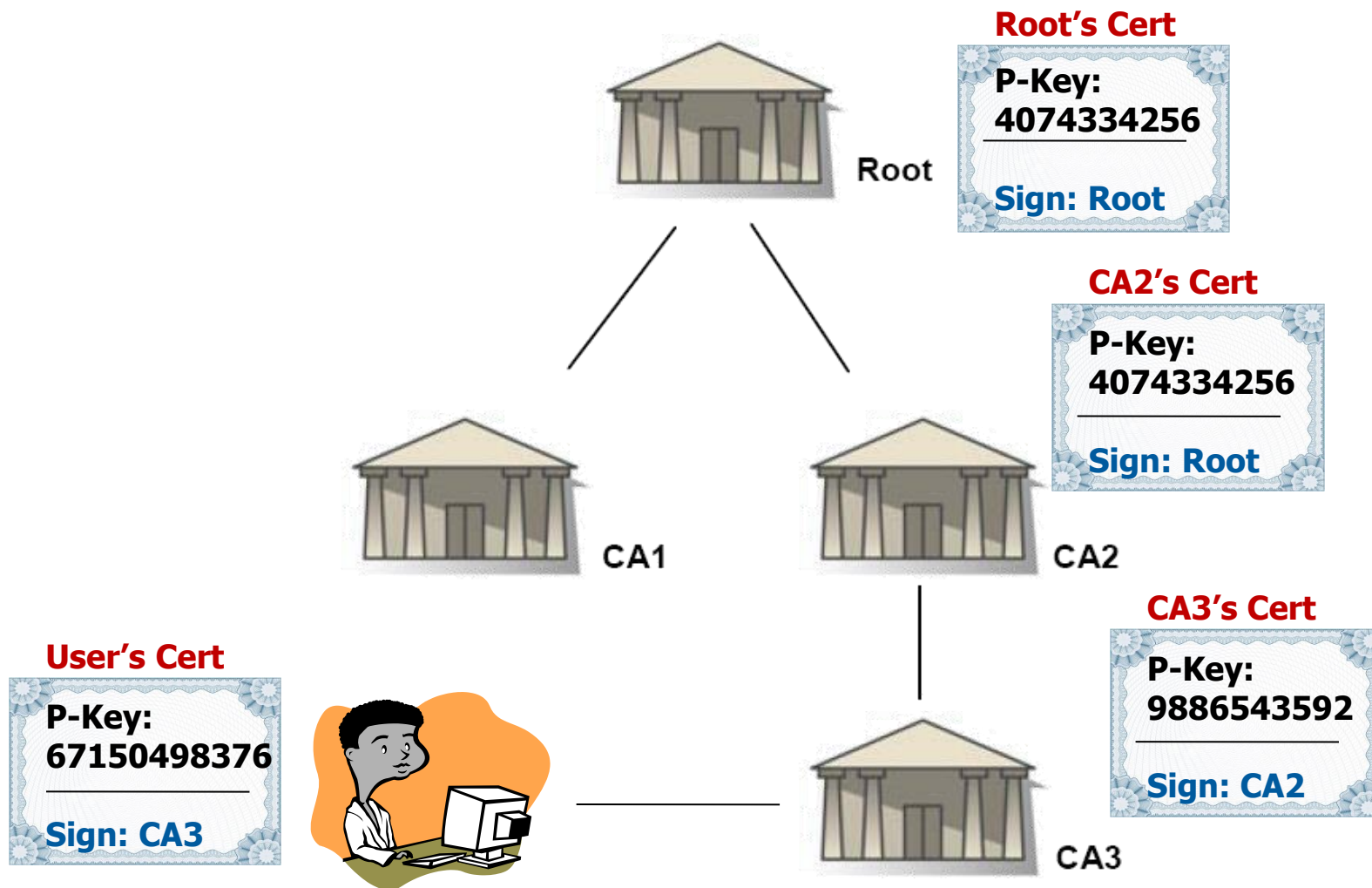
• صرفاً برای ریشه امنیت بالا نیاز است.

□ معایب:

☞ همه CAها را نمی توان در یک سلسله مراتب جای داد.

خود امضا  
(Self-Signed)





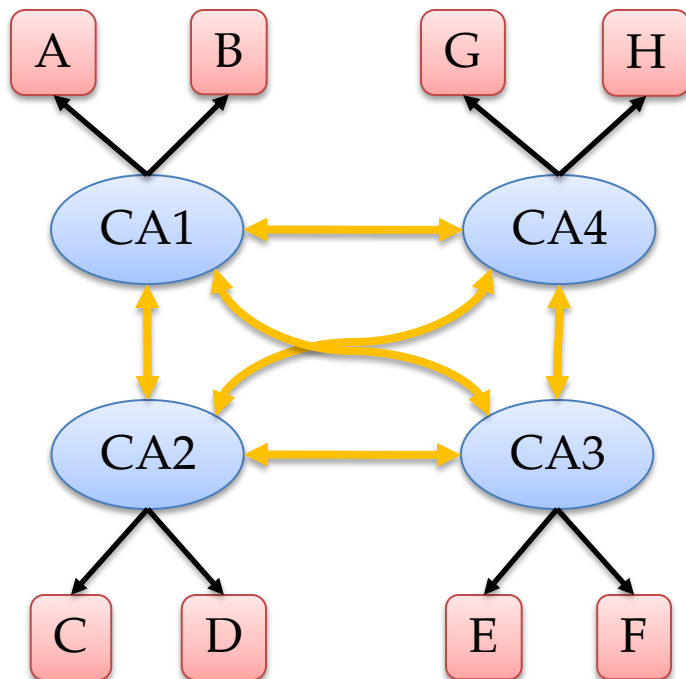
□ هر دو CA موجود در سیستم برای یکدیگر گواهی ضربدری صادر کنند.

□ مزایا:

☞ استقلال CA ها از یکدیگر

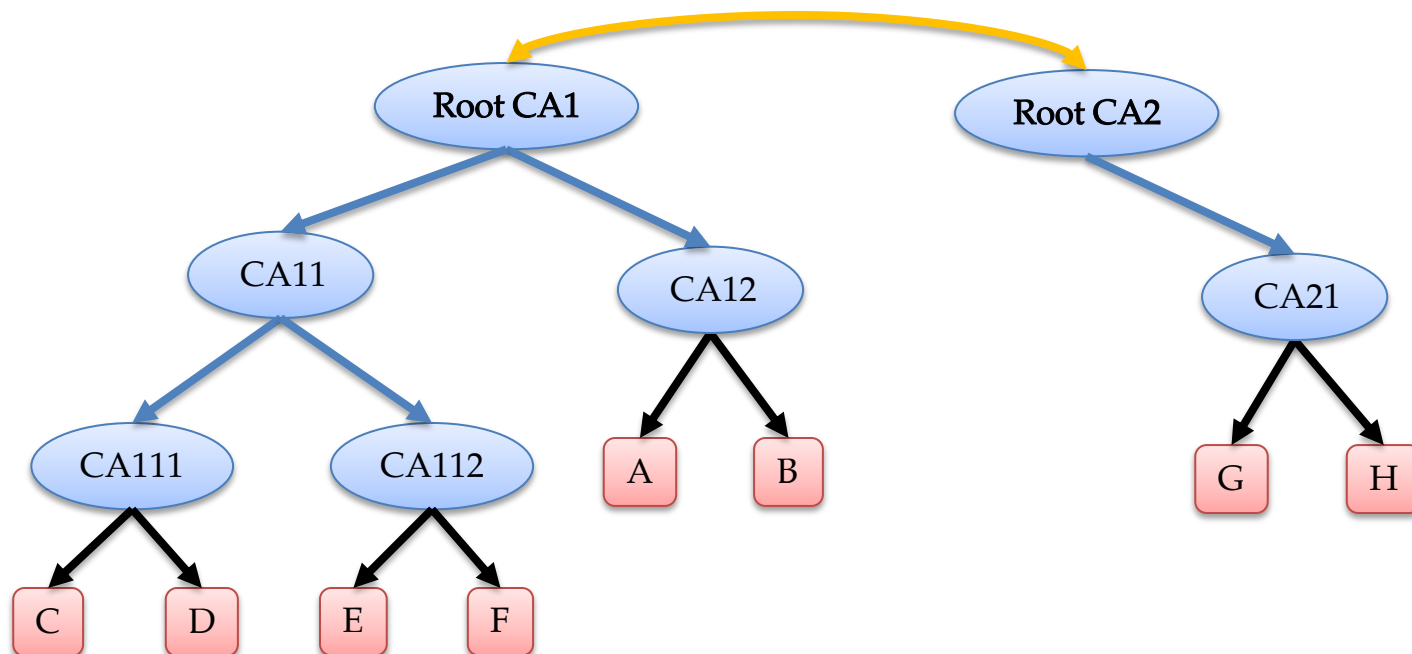
□ معایب:

☞ نیاز به صرف منابع و هزینه زیاد



□ ساختار درختی برای هر بخش

□ ارتباط درختها با یکدیگر از طریق گواهی ضربدری در سطح ریشه



□ برای داشتن PKI، وجود دو مستند ضروری است:

➡ سیاست نامه گواهی دیجیتال

CP: Certificate Policy

➡ آیین نامه اجرایی گواهی دیجیتال

CPS: Certificate Practice Statement

□ استاندارد فعلی برای این دو مستند RFC 3647 است.

□ CP یک مستند سطح بالا است که **مؤلفه‌های درگیر** در یک PKI مشخص، و **نقش و وظیفه** هر مؤلفه را تشریح می‌کند.

□ با مطالعه CP هر CA می‌توان میزان اعتماد به گواهی‌های صادره توسط آن CA را تعیین نمود.

□ CPS مستندی است که مطابق با CP یک مرکز تدوین شده و نحوه اجرایی شدن CP را بیان می‌کند.

➡ رویه‌های اجرایی صدور، انتشار، بایگانی، ابطال و صدور مجدد