Network Security
Elements of Applied Cryptography

Public Key Infrastructure

- **Certificati**
- **Le infrastrutture a chiave pubblica**
- **Standard X509v3**
- **Cenni alla normativa italiana**

Certificato

PROBLEMA. Rendere la chiave pubblica (e_A) di un soggetto (Alice) disponibile agli altri in modo tale che essi possono verificarne l'*autenticità* e la *validità*

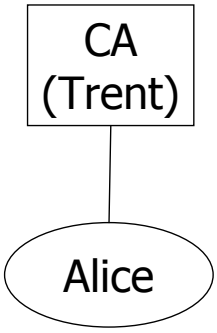
IPOTESI. Ogni soggetto può essere univocamente identificato per mezzo di un identificatore (distinguished name)

L'**autorità di certificazione** è una TTP fidata che attesta l'autenticità di una chiave pubblica

Il **certificato** (public-key certificate) è una struttura dati che lega l'identificatore unico di un soggetto alla sua chiave pubblica

Il certificato porta la firma digitale dell'autorità di certificazione

Creazione di un certificato



1. CA verifica l'identità di Alice
2. CA verifica che la chiave pubblica da certificare sia proprio quella di Alice
3. CA genera un certificato

$$\mathbf{C(T, A) = e_A, A, L, S_T(e_A, A, L)}$$

con **L** periodo di validità*

data part — signature part

* C(T, A) è denotato anche con T<<A>>, T{A}

Un certificato può anche specificare:

- informazioni aggiuntive sul soggetto
- informazioni aggiuntive sulla chiave (algoritmo,…)
- politica seguita per l'identificazione del soggetto, la generazione della chiave,…
- informazioni che facilitano la verifica della firma (algortimo,…)

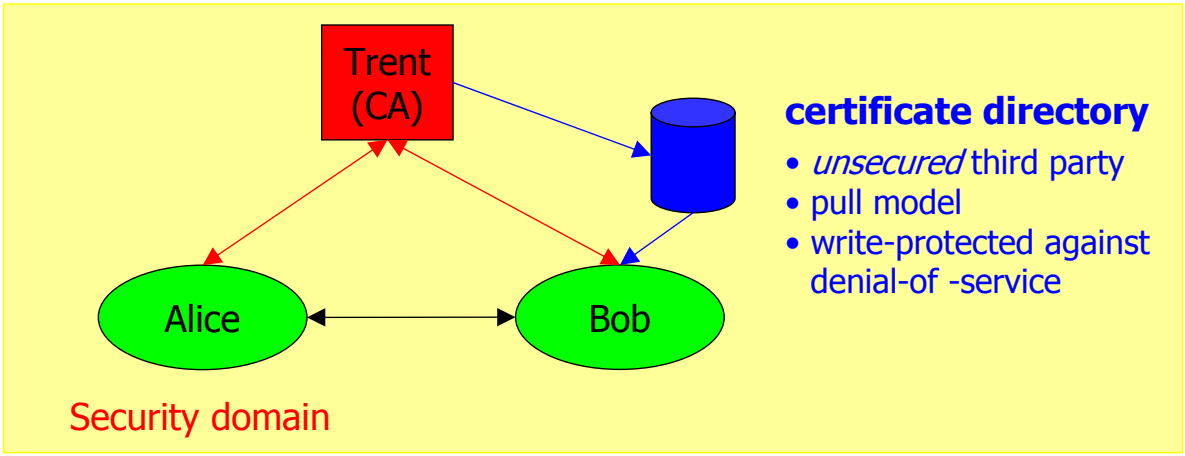
Creazione di un certificato

Per rilasciare un certificato la CA deve

- *verificare l'identità del soggetto*
  - Tipicamente, attraverso procedure non crittografiche
- *verificare l'autenticità della chiave*
  - *Due possibili scenari*
    - *Scenario 1*. La coppia di chiavi pubblica-privata è generata dalla CA ed è trasferita al soggetto in modo da preservarne l'autenticità e la segretezza
    - *Scenario 2*. La coppia di chiavi pubblica-privata è generata dal soggetto e la chiave pubblica è trasferita alla CA in modo da preservarne l'autenticità

      In questo caso la CA deve richiedere al soggetto una prova di conoscenza della corrispondente chiave privata (ad esempio, per mezzo di challenge-response)

Modello con singola CA



- Un **security domain** è un (sotto-)sistema sotto il controllo di una singola autorità di cui tutte le altre entità si fidano
- Il **certicate directory** è un database accessibile in sola lettura che memorizza certificati e che è gestito da una terza parte insicura

Uso e verifica di un certificato

1. Bob si procura la chiave pubblica di Trent (e_T) [*one time*]
2. Bob si procura l'identificatore unico *A* di Alice
3. Bob si procura un certificato *C*(*T*, *A*)
4. Bob verifica il certificato
   1. Bob verifica la validità della chiave di Trent
   2. Bob verifica che il certificato *C*(*T*, *A*) sia ancora valido
   3. Bob verifica la firma su *C*(*T*, *A*) usando la chiave pubblica di Trent
   4. Bob verifica che il certificato *C*(*T*, *A*) non sia stato revocato
5. Se tutte le la verifiche hanno esito positivo, allora Bob accetta e_A come la chiave autentica di Alice

Uso e verifica di un certificato

La certificazione si basa sul principio di ***delega della fiducia***

- Ogni soggetto che utilizza un certificato delega alla CA la fiducia di verificare l'identità di un altro soggetto e di attestare l'autenticità della rispettiva chiave
- Bob ha fiducia nell'autenticità della chiave pubblica di CA
- Attraverso il processo di verifica di un certificato, Bob **acquisisce transitivamente fiducia** nell'autenticità della chiave contenuta in un qualunque certificato firmato da CA

Certificato degli attributi

*Un certificato **attesta** il legame tra una chiave pubblica e l'identificatore di un soggetto **ma non specifica** il significato di questo legame, cioè non specifica per cosa può essere utilizzata la chiave*

- ***Il certificato degli attributi (attribute certificate)***
  - permette di legare una chiave a delle informazioni (*attributi*)
  - è firmato da una TTP detta *Autorità di Certificazione degli Attributi*: **$AC(R, A) = e_A, \alpha, S_R(\alpha, e_A), C(T, A),$** con α attributi
  - Esempi di attributo:
    - ✓ assegnare informazioni di autorizzazione ad una chiave;
    - ✓ vincolare l'uso di una firma digitale (utilizzabile in transazioni di una certa entità, di un certo tipo, ad una certa ora,…)

Revoca delle chiavi

Un certificato è **scaduto** (expired certicate) se è scaduto il periodo di validità della chiave

Se, per qualunque motivo, una chiave diventa invalida prima della scadenza, allora il relativo certificato deve essere **revocato**

- la chiave è stata compromessa
- l'utente ha cambiato ruolo, ha lasciato l'organizzazione, oppure non necessita più di un'autorizzazione

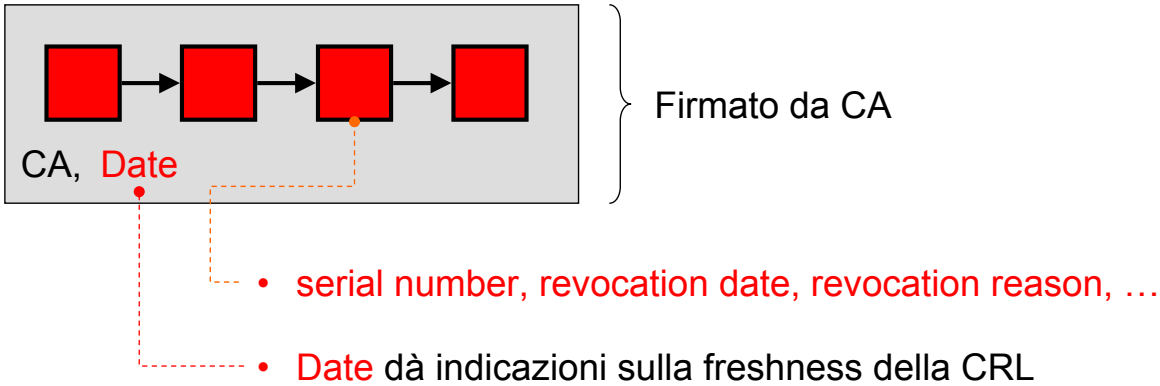
La revoca di un certificato deve essere

- **corretta**: la revoca può essere richiesta solo da chi è autorizzato: il possessore del certificato (*subject*) oppure chi lo ha emesso (*issuer*)
- **tempestiva**: la revoca di un certificato deve essere disseminata a tutti gli interessati il prima possibile

Gestione della revoca delle chiavi

- **Data di scadenza nei certificati**. Limita l'esposizione conseguente alla compromissione
- **Notifica manuale**. Gli utenti vengono avvisati out-of-band o tramite canali speciali. Questa soluzione non è scalabile
- **File pubblico di chiavi revocate**. Contiene le chiavi revocate; deve essere controllato prima di ogni utilizzo di una chiave.
  - **Certificate revocation list (CRL)** è un metodo per gestire un file pubblico di chiavi revocate.
- **Revocation certificates**. Soluzione alternativa a CRL;
  - è un certificato con il **revocation flag** attivo;
  - nella directory dei certificati, il certificato originale viene sostituito dal revocation certificate (certificato di revoca)

Certificate Revocation List



- Un certificato revocato rimane nella CRL fintanto che non scade
- Nella modalità *pull*, la CRL viene pubblicata ad intervalli regolari, anche se non ci sono modifiche, per evitare replay attack

Certificate Revocation List

- Quando la CRL diventa troppo grande, una soluzione è trasmetterla a pezzi (CRL segmenting)
  - delta-CRL: end-user mantiene la CRL in modo sicuro
  - partizionamento in base al motivo di revoca;
  - pre-assegnamento di un certificato all'*i*-esimo segmento di CRL lungo *nmax* (il certificato specifica il segmento a cui è stato pre-assegnato)

Certificate Revocation List

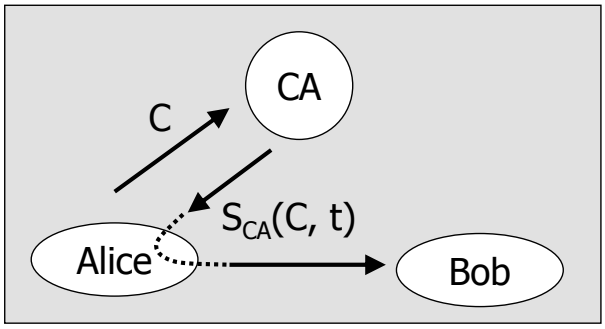
CRL permettono la verifica off-line dei certificati secondo un approccio simile alla gestione carte di credito revocate

Svantaggi

- Un avversario utilizza una chiave pubblica fino alla prossima distribuzione della CRL
- Spesso le CRL sono l'ultima componente ad essere realizzata e talvolta non vengono neanche realizzate
  - I primi browser non implementavano l'accesso a CRL
  - Caso Microsoft – Verisign: quando Microsoft cercò di accedere alla CRL di Verisign, si accorse che i certificati rilasciati da quest'ultima non avevano neanche il puntatore alla CRL

Approcci alternativi alla CRL

- CONTROLLO ON-LINE DEI CERTIFICATI secondo un approccio simile al controllo in linea delle carte di credito
  - Vantaggio: accuratezza
  - Svantaggio: poca affidabilità
- TIMELY-CERTIFICATION (short-term certificate). Bob richiede ad Alice un certificato recente; è Bob a specificare quanto deve essere recente il certificato



- *C*: certificato valido rilasciato da CA ad Alice
- $S_{CA}(C, t)$: copia "aggiornata" del certificato
- *t* marca temporale contenuta nell'intervallo di validità del certificato

Non-repudiation

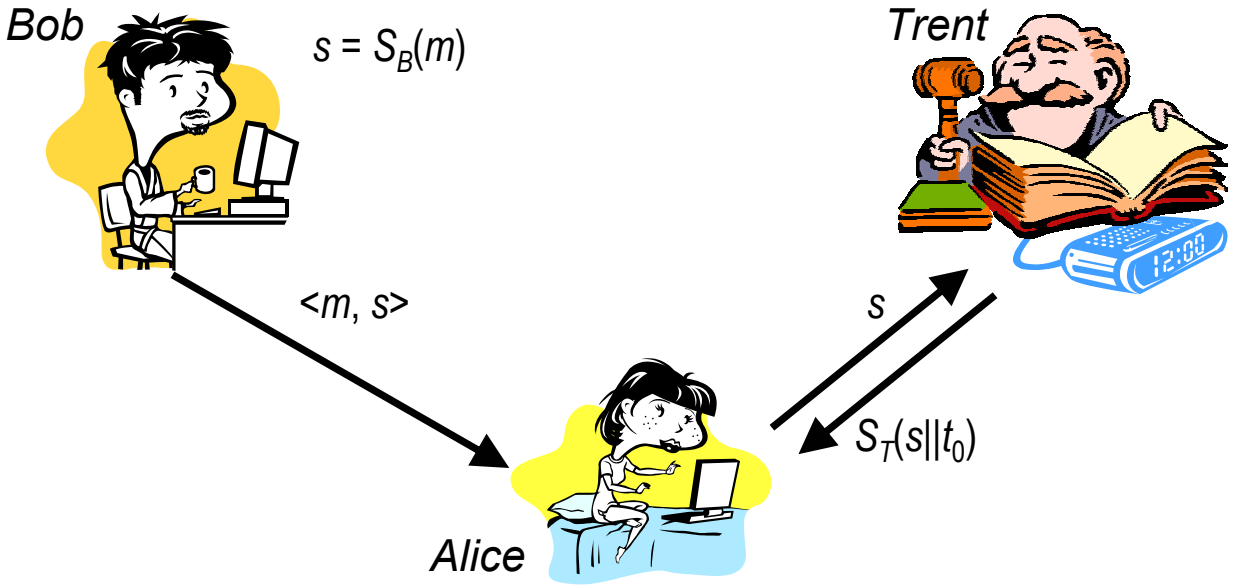
- Non-repudiation prevents a signer from signing a document and subsequently being able to successfully deny having done so.
- Non-repudiation vs authentication of origin
  - Authentication (based on symmetric cryptography) allows a party to convince **itself** or a **mutually trusted party** of the integrity/authenticity of a given message at a given time t_0
  - Non-repudiation (based on public-key cryptography) allows a party to convince **others** at any time $t_1 \geq t_0$ of the integrity/authenticity of a given message at time t_0

Alice's digital signature for a given message depends on the message and a secret ***known to Alice only (the private key)***

Non-repudiation

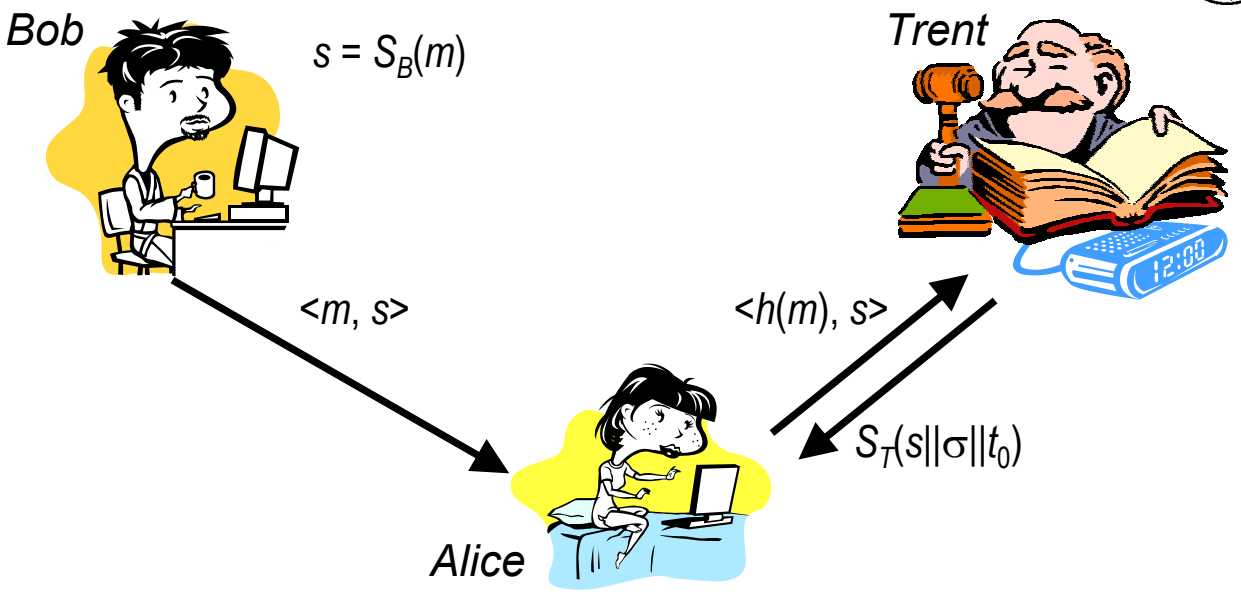
- Data origin authentication as provided by a digital signature is valid only while the *secrecy* of the signer's *private key* is maintained
- A threat that must be addressed is a signer who *intentionally* discloses his private key, and thereafter claims that a *previously* valid signature was forged
- This threat may be addressed by
  - *preventing direct access to the key*
  - *use of a trusted timestamp agent*
  - *use of a trusted notary agent*

Trusted timestamping service



- Trent certifica che la firma digitale ***s*** *esiste* all'istante t_0
- La firma digitale ***s*** (non) è valida se l'istante ***t*** a cui viene denunciato che S_B è compromessa è (minore) maggiore di t_0

Trusted Notary Service

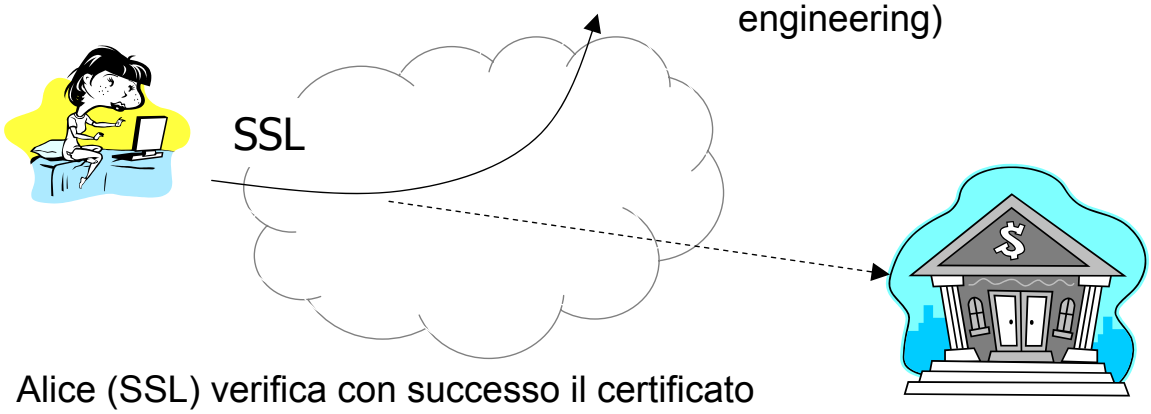


- Trent certifica che un certo *statement* σ sulla firma s è vero all'istante t_0
  esempio: σ la firma è valida

Il certificato è quello giusto?

Tipicamente, nei sistemi SSL-based, l'autenticazione del cliente è basata su PWD

- L'avversario dispone di un certificato valido
- L'avversario induce Alice a connettersi ad un sito sotto il suo controllo (spoofing, social engineering)



- Alice (SSL) verifica con successo il certificato dell'avversario,
  stabilisce la connessione ed
  invia la propria PWD all'avversario

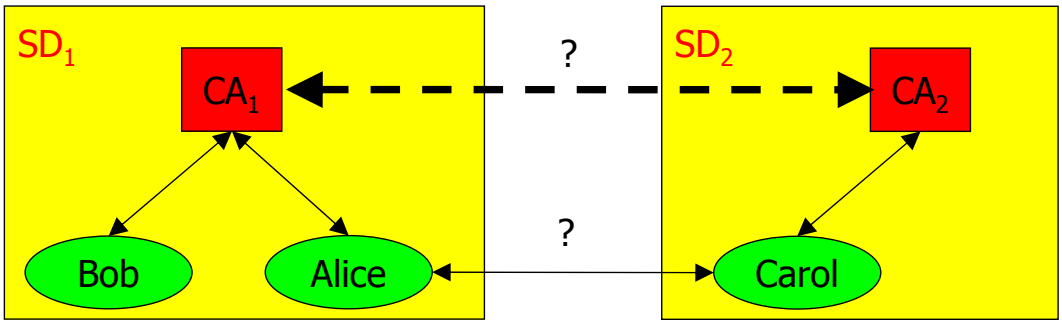
Il certificato è quello giusto?

Il problema è che SSL opera ad un livello più basso di quello applicativo

➔ *È l'applicazione che deve (indurre l'utente a) verificare che il nome richiesto sia uguale al nome contenuto nel certificato verificato*

- *Esempio: Netscape*
  - Il browser *notifica* all'utente se l'URL specificato dal browser e quello contenuto nel certificato del server sono diversi
  - L'utente decide se proseguire la connessione oppure no (interfaccia utente!!!)
  - *In linea di principio non è detto che il controllo eseguito dal browser Netscape sia sufficiente per ogni tipo di applicazione Web-based*

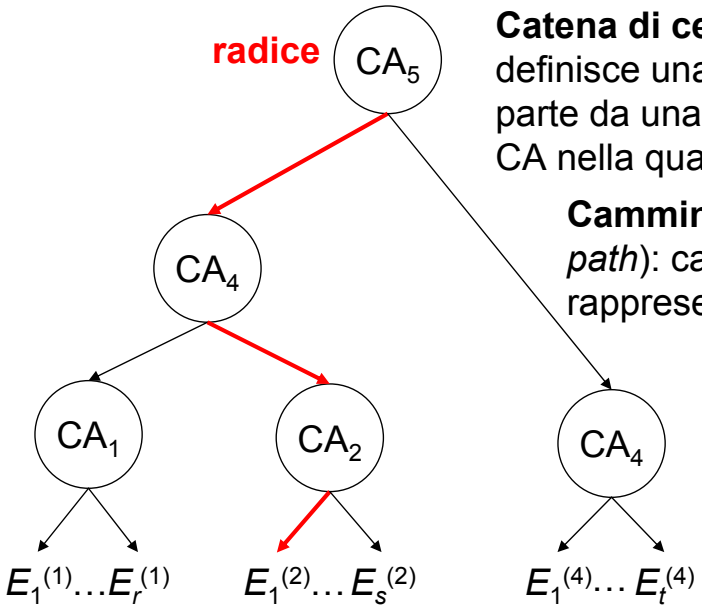
CA multiple e modelli di fiducia



- Affinché entità appartenenti a diversi domini di sicurezza possano interagire è necessario che sia possibile stabilire una ***relazione di fiducia*** **(*trust relationship*)** tra le rispettive CA
- Le relazioni di trust tra le CA permettono di determinare come i certificati rilasciati da una CA possono essere utilizzati e verificati da entità certificate da un'altra CA

Modello di trust centralizzato

Alice vuole verificare il certificato $CA_2\langle\langle E_i^{(2)}\rangle\rangle$



Catena di certificati (*chain of certificates*): definisce una catena ininterrotta di fiducia che parte da una CA di cui Alice si fida e termina nella CA nella quale Alice vuole ottenere fiducia

Cammino di certificazione (*certification path*): cammino diretto nel grafo che rappresenta il trust model

Nel modello centralizzato ogni entità conosce a priori la chiave pubblica della radice

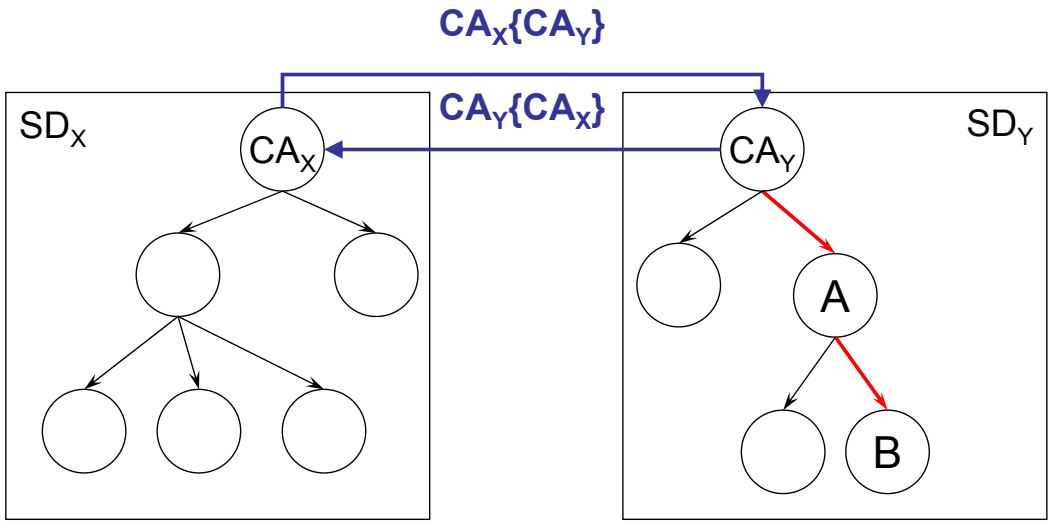
Catena di certificati per CA_2: $CA_5\{CA_4\}CA_4\{CA_2\}$

Modello centralizzato: considerazioni

- Il modello centralizzato definisce un unico dominio di sicurezza
- Tutto il trust del sistema dipende dalla chiave pubblica della radice
- Una catena di certificati è necessaria anche per due entità che stanno sotto la stessa CA (e che rilascia i loro certificati) poiché ciascuna entità ha fiducia diretta solo nella radice
- Le catene di certificati tendono ad essere lunghe
- È un modello di trust innaturale

  un modello più naturale: il trust inizia da un nodo locale (CA padre) piuttosto che da un nodo remoto (CA radice)

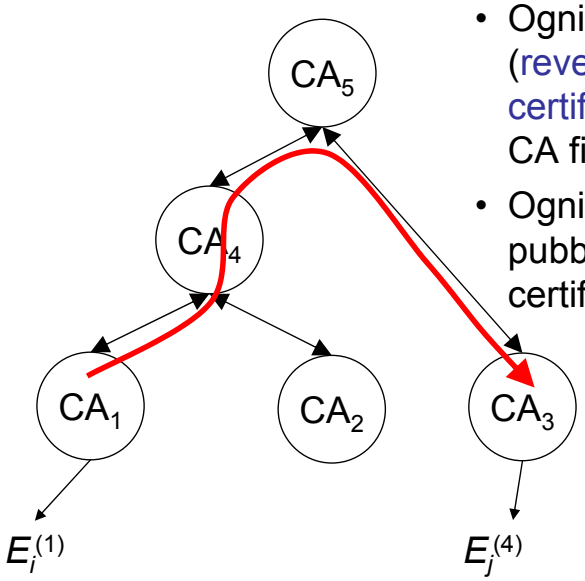
Modello a radici multiple



I certificati incrociati (cross-certificate) permettono ad una entità in SD_X (SD_Y) di ottenere fiducia (trust) nei certificati rilasciati in SD_Y (SD_X) da CA_Y (CA_X)

Catena di certificati per B: $CA_X\{CA_Y\}CA_Y\{A\}A\{B\}$

Gerarchia con certificati inversi

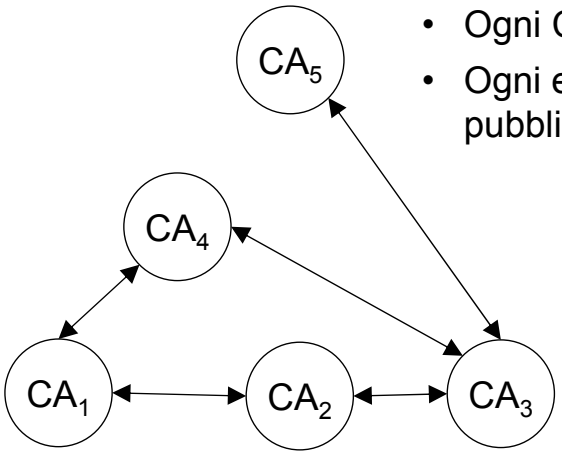


- Ogni CA crea un certificato inverso (reverse certificate) per la CA padre ed un certificato diretto (forward certificate) per la CA figlio
- Ogni entità conosce a priori la chiave pubblica della CA che ha creato il suo certificato

- Questa soluzione dà luogo a catene "lunghe" anche tra CA che comunicano frequentemente
- Questa soluzione può essere migliorata attraverso la cross-certificazione

Catena di certificati per CA_3: $CA_1\{CA_4\}CA_4\{CA_5\}CA_5\{CA_3\}$

Modello distribuito



- Ogni CA può cross-certificare ogni altra CA
- Ogni CA può certificare gli end user
- Ogni entità conosce a priori la chiave pubblica della propria CA locale

Catena di certificati per CA_3: $CA_1\{CA_2\}CA_2\{CA_3\}$

Vincoli nei modelli di trust

- Se CA_X cross-certifica CA_Y, la fiducia che CA_X ripone in CA_Y si **propaga transitivamente** a tutte le CA raggiungibili da CA_Y
- CA_X può **limitare** questa propagazione imponendo dei **vincoli** sui cross-certificati che essa firma
  - **Limitazione della lunghezza**
    la catena di certificati che segue il cross-certificate può avere una lunghezza massima predeterminata
  - **Limitazione dell'insieme dei domini validi**
    le CA nella catena di certificati che segue il cross-certificate devono appartenere ad un insieme predeterminato di CA

Certificato X.509 (RFC 3280)

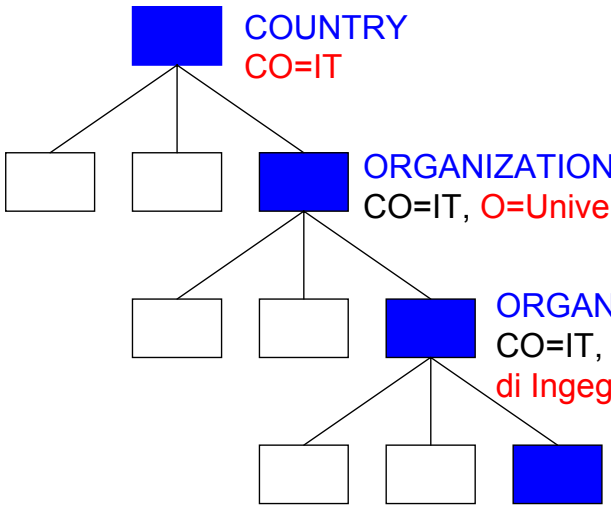
Struttura costituita dai seguenti campi:

1. Version
2. Serial number
3. Signature algorithm identifier
4. Issuer distinguished name
5. Validity interval
6. Subject distinguished name
7. Subject public key information
8. Issuer unique identifier (v=2,3)
9. Subject unique identifier (v=2,3)
10. Extensions (v=3)
11. Signature

- *Serial number* del certificato deve essere unico rispetto all'issuer
- *Distinguished name*, identificatore unico
- *Signature algorithm identifier* specifica l'algoritmo e la chiave pubblica dell'issuer
- *Subject public key information* specifica l'algoritmo, i parametri e la chiave pubblica del subject
- *Signature* di hash dei campi 1-10

Distinguished names (X.500)

L'organizzazione gerarchica "garantisce" l'unicità dei nomi ed "induce" l'organizzazione delle CA



COUNTRY
CO=IT

ORGANIZATION
CO=IT, O=University of Pisa

ORGANIZATIONAL UNIT
CO=IT, O=University of Pisa, OU=Dipartimento di Ingegneria della Informazione

COMMON NAME
CO=IT, O=University of Pisa, OU=Dipartimento di Ingegneria della Informazione, CN=Gianluca Dini

Estensioni

Le estensioni sono classificate come

- *Informazioni sulle chiavi e sulle politiche*
  - Identificatore della chiave dell'issuer
  - Identificatore della chiave del soggetto
  - Utilizzo della chiave
  - Periodo di utilizzo della chiave privata
  - Politiche riguardanti i certificati
  - Corrispondenze tra politiche
- *Attributi del certificato relativi al soggetto ed a chi lo emette*
  - Nome alternativo del soggetto
  - Nome alternativo dell'entità che ha emanato il certificato
  - Attributi relativi alla directory del soggetto
- *Vincoli sul percorso di certificazione*
  - Vincoli di base
  - Vincoli sul nome
  - Vincoli sulla politica

Esempio: https://www.mps.it

Certificate name
www.mps.it
Consorzio Operativo Gruppo MPS
Terms of use at www.verisign.com/rpa (c)00
Florence
Italy, IT

Issuer
VeriSign Trust Network
www.verisign.com/CPS Incorp.by Ref. LIABILITY LTD.(c)97 VeriSign

Details
Certificate version: 3
Serial number: 0x652D0F8ADAB4C7B168A27BBD1C3E9D9D
Not valid before: Mar 2 00:00:00 2004 GMT
Not valid after: Mar 2 23:59:59 2005 GMT
Fingerprint: (MD5) CA CA 88 08 EC D0 8E 49 A6 9A 66 C4 69 31 E0 AE
Fingerprint: (SHA-1) 82 64 CB 69 F0 43 86 43 FF B4 55 D4 25 EF 51 60 65 46 D3 87

continua

Esempio: https://www.mps.it

Public key algorithm: rsaEncryption
Public-Key (1024 bit):
Modulus:
00: E1 80 74 5E E7 E5 54 8B DF 6D 00 95 B5 96 27 AC
10: 66 93 E0 49 B9 6F 5B 73 53 1C BE 1C EB 47 64 B2
20: 12 95 70 E6 CD 50 67 02 88 E3 EE 9D B1 91 49 C8
30: 8D 58 19 4B 86 8F C0 2E 65 E8 F2 D4 82 CC 55 DB
40: 43 BC 66 DA 44 2F 53 B3 48 4B 37 15 F3 AB 67 C1
50: 69 B4 53 23 19 30 1A 19 23 7F 28 E0 E3 C0 6B 18
60: FF 84 C4 AC A9 74 28 DB FF E9 48 CA 75 D5 35 D6
70: 46 FB 7D D4 A7 3F A1 4B 00 60 14 DC D5 00 CF C7
Exponent:
01 00 01
Public key algorithm: sha1WithRSAEncryption
00: 23 A6 FE 90 E3 D9 BB 30 69 CF 43 2C FD 4B CF 67
10: D7 3C 46 22 9A 08 DB 05 1D 45 DC 07 F3 1E 4D 1F
20: 4B 11 23 5B 42 91 14 95 25 88 1F BD 60 E5 6F 84
30: 44 70 7A 95 EC 30 E4 46 4F 37 87 F1 B2 FA 45 04
40: 6F 7C BE 97 25 C7 20 E7 F3 90 55 51 99 3A 72 35
50: 40 F2 E8 E3 36 3A 7D 58 61 9C 91 D6 AC 34 E7 E8
60: 09 27 64 4F 2C 4C C2 D2 A3 32 DB 2B 7E F0 B6 F3
70: 69 96 E4 2B C3 2B 42 ED CA 2C 3C C8 F5 AA E6 71

continua

Esempio: https://www.mps.it

Extensions:
X509v3 Basic Constraints: CA:FALSE
X509v3 Key Usage: Digital Signature, Key Encipherment
X509v3 CRL Distribution Points:
URI:http://crl.verisign.com/Class3InternationalServer.crl
X509v3 Certificate Policies:
Policy: 2.16.840.1.113733.1.7.23.3
CPS: https://www.verisign.com/rpa
X509v3 Extended Key Usage: Netscape Server Gated Crypto, Microsoft Server Gated Crypto, TLS Web Server Authentication, TLS Web Client Authentication
Authority Information Access:
OCSP - URI:http://ocsp.verisign.com
Unknown extension object ID 1 3 6 1 5 5 7 1 12:
0_.].[0Y0W0U..image/gif0!0.0...+.....k...j.H.,{..0%.#http://logo.verisign.com/vslogo.gif

Esempio: https://www.mps.it

Certificate name
VeriSign Trust Network
www.verisign.com/CPS Incorp.by Ref. LIABILITY LTD.(c)97 VeriSign

Issuer
VeriSign, Inc.
Class 3 Public Primary Certification Authority
US

Details
Certificate version: 3
Serial number: 0x254B8A853842CCE358F8C5DDAE226EA4
Not valid before: Apr 17 00:00:00 1997 GMT
Not valid after: Oct 24 23:59:59 2011 GMT
Fingerprint: (MD5) BC 0A 51 FA C0 F4 7F DC 62 1C D8 E1 15 43 4E CC
Fingerprint: (SHA-1) C2 F0 08 7D 01 E6 86 05 3A 4D 63 3E 7E 70 D4 EF 65 C2 CC 4F

Esempio: https://www.mps.it

Public key algorithm: rsaEncryption
Public-Key (1024 bit):
Modulus:
00: 6F 7B B2 04 AB E7 34 4F 9C 53 A7 02 B2 90 4F 22
10: F9 3A 3C 5A 8B 51 2B FE CB 42 95 30 70 FE 8A B2
20: D3 1D C1 B8 5A 49 5C F7 39 4E 4D B7 F3 3B 09 F1
30: FA E5 28 93 3E 30 F5 63 AA 43 71 27 56 FE A3 BB
40: CA C4 6C 75 B2 32 C1 07 D9 DD 25 40 F5 5C A9 D4
50: 15 0A 34 9A ED 42 97 EA BD F1 B2 55 45 73 3C AA
60: E7 B6 5B 6C 4C F0 AA 3B 36 E6 BC D3 05 D4 BF E1
70: 2B 65 A2 25 39 18 85 1F 7D 02 19 D6 E8 80 82 D8
Exponent:
01 00 01
Public key algorithm: sha1WithRSAEncryption
00: 08 01 EC E4 68 94 03 42 F1 73 F1 23 A2 3A DE E9
10: F1 DA C6 54 C4 23 3E 86 EA CF 6A 3A 33 AB EA 9C
20: 04 14 07 36 06 0B F9 88 6F D5 13 EE 29 2B C3 E4
30: 72 8D 44 ED D1 AC 20 09 2D E1 F6 E1 19 05 38 B0
40: 3D 0F 9F 7F F8 9E 02 DC 86 02 86 61 4E 26 5F 5E
50: 9F 92 1E 0C 24 A4 F5 D0 70 13 CF 26 C3 43 3D 49
60: 1D 9E 82 2E 52 5F BC 3E C6 66 29 01 8E 4E 92 2C
70: BC 46 75 03 82 AC 73 E9 D9 7E 0B 67 EF 54 52 1A

Esempio: https://www.mps.it

Extensions:
X509v3 Basic Constraints: CA:TRUE, pathlen:0
X509v3 Certificate Policies:
Policy: 2.16.840.1.113733.1.7.1.1
CPS: https://www.verisign.com/CPS

X509v3 Extended Key Usage: TLS Web Server Authentication, TLS Web Client Authentication, Netscape Server Gated Crypto, 2.16.840.1.113733.1.8.1
X509v3 Key Usage: Certificate Sign, CRL Sign
Netscape Cert Type: SSL CA, S/MIME CA
X509v3 CRL Distribution Points:
URI:http://crl.verisign.com/pca3.crl

Assurance: il caso di Verisign

- **Verisign distribuisce tre classi di certificati; ogni classe definisce sia l'uso appropriato sia le procedure di autenticazione**

- **Class 1 Certificates**. Class 1 Certificates offer the lowest level of assurances within the VTN. The Certificates are issued to individual Subscribers only, and authentication procedures are based on assurances that the Subscriber's distinguished name is unique and unambiguous within the domain of a particular CA and that a certain e-mail address is associated with a public key. Class 1 Certificates are appropriate for digital signatures, encryption, and access control for non-commercial or low-value transactions where proof of identity is unnecessary.
- **Class 2 Certificates**. Class 2 Certificates offer a medium level of assurances in comparison with the other two Classes. Again, they are issued to individual Subscribers only. In addition to the Class 1 authentication procedures, Class 2 authentication includes procedures based on a comparison of information submitted by the certificate applicant against information in business records or databases or the database of a VeriSign-approved identity proofing service. They can be used for digital signatures, encryption, and access control, including as proof of identity in medium-value transactions.
- **Class 3 Certificates**. Class 3 Certificates provide the highest level of assurances within the VTN. Class 3 Certificates are issued to individuals and organizations for use with both client and server software. Class 3 individual Certificates may be used for digital signatures, encryption, and access control, including as proof of identity, in high-value transactions. Class 3 individual Certificates provide assurances of the identity of the Subscriber based on the personal (physical) presence of the Subscriber before a person that confirms the identity of the Subscriber using, at a minimum, a well-recognized form of government-issued identification and one other identification credential. Class 3 organizational Certificates are issued to devices to provide authentication; message, software, and content integrity and signing; and confidentiality encryption. Class 3 organizational Certificates provide assurances of the identity of the Subscriber based on a confirmation that the Subscriber organization does in fact exist, that the organization has authorized the Certificate Application, and that the person submitting the Certificate Application on behalf of the Subscriber was authorized to do so. Class 3 organizational Certificates for servers also provide assurances that the Subscriber is entitled to use the domain name listed in the Certificate Application, if a domain name is listed in such Certificate Application.

Nomi e certificati

Una CA deve attestare l'identità di un'entità, ma…

…quando si riceve un certificato, "quanto" ci si può fidare che l'identità corrisponda proprio al possessore della chiave

- Authentication policy, specifica il livello di autenticazione che la CA richiede per verificare un'identità
- Issuance policy, specifica le entità a cui la CA assegna certificati (individui, server, …)
- Queste politiche devono essere pubbliche
- Se una CA rilascia certificati ad altre CA, le politiche adottate da quest'ultime devono essere più restrittive
- Il livello di fiducia non è quantificabile, ma può essere stimato in base alla politica della CA ed il rigore con cui tale politica è seguita

Assurance

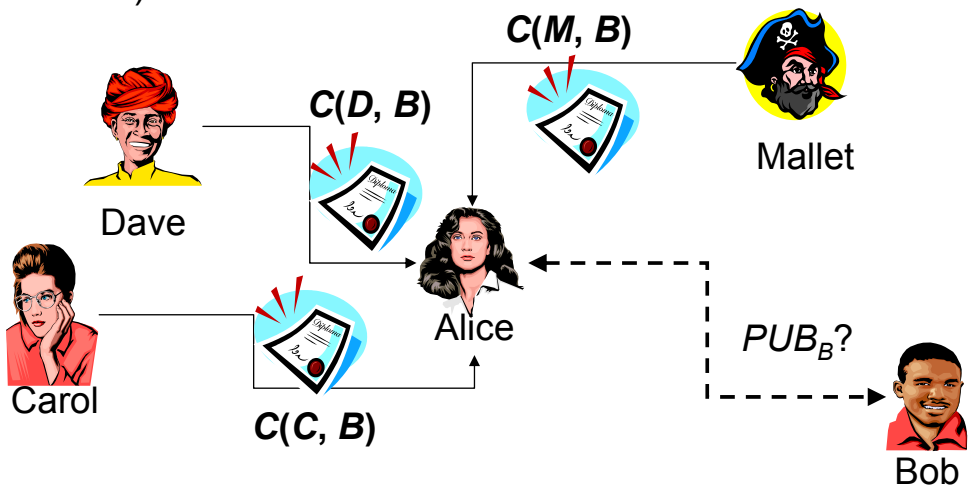
- Il livello di fiducia non può essere quantificato precisamente
- La specifica, il progetto e l'implementazione di un sistema concorrono a determinare "quanta" fiducia riporre in un sistema (**assurance**)
- ESEMPIO: un medicinale prodotto da una casa farmaceutica nota ed onorabile, consegnato alla farmacie in un contenitore con sigillo di sicurezza e venduto con tale sigillo ancora integro è considerato affidabile

  Le basi per tale fiducia sono le seguenti
  - I test del Ministero che autorizza la commercializzazione del medicinale solo se esso supera certi standard clinici e di utilità
  - Commissioni di controllo che assicurano che il processo produttivo soddisfi determinati standard industriali
  - La presenza del sigillo di sicurezza

Pretty Good Privacy (PGP)

È il singolo utente che decide quanta fiducia riporre in un certificato

"*PGP is for people who prefer to pack their own parachutes*"
(P. Zimmerman)



In base al numero di certificati e la fiducia in ciascun individuo, Alice definisce il proprio livello di fiducia in PUB_B

Certificato PGP

Certificato ::= ⟨public key packet⟩ { ⟨signature packet⟩ }

Public key packet
- Version
- Creation time
- Validity period
- Public key algorithm and parameters
- Public key

Signature packet
- Identifier
- Key identifier of the signer
- Public key algorithm
- Hash algorithm
- Signature
- …

PGP: Validità e livello di fiducia

Il livello di fiducia in una chiave può essere

nella propria chiave

- Implicit trust

nelle chiavi di altri

- Complete trust
- Marginal trust
- No trust (untrusted)

È l'utente che decide il livello di trust da assegnare ad una chiave

Una chiave può essere:

- Valid
- Marginally Valid
- Invalid

Una chiave è ***valida*** se è stata firmata da una chiave di livello ***Complete trust*** oppure da almeno due chiavi di livello ***Marginal***

Certificati: PGP vs X.509

- In X.509, una chiave pubblica è firmata solo una volta;
  In PGP, una chiave pubblica può essere firmata più volte
- In X.509,
  un certificato è implicitamente associato ad un certo livello di trust che dipende dalla politica della CA e dal rigore con cui CA applica tale politica
  In PGP,
  ogni firma è associata ad un livello di trust esplicito;
  le firme su di una stessa chiave possono avere livelli di trust differenti;
  il significato di un livello di trust dipende dal contesto

CA in-house o commerciale?

Per un azienda, è meglio realizzare la propria CA oppure rivolgersi ad una CA commerciale?

Rapporto costo-convenienza

- Processo di certificazione di "alta qualità" comporta costi più elevati
- Processo di certificazione di "bassa qualità" comporta rischi più elevati

Soluzione in-house

- Vantaggi: completo controllo del processo di certificazione; l'azienda valuta i propri rischi e sceglie la soluzione che ritiene più appropriata
- Svantaggi
  - Costi dell'infrastruttura necessaria
  - Scala limitata a causa del modello di trust a radici multiple

Soluzione outsourcing

- Vantaggio: scala (i certificati sono accettati dalla maggior parte dei browser)
- Svantaggio: delega di fiducia; CA tipicamente non si assumono responsabilità finanziarie in caso di errori (C*ertification Practices Statement*)

CA in-house o commerciale?

Confronto tra certificati e carte di credito

- Una carta di credito, come un certificato, offre un servizio di third-party autentication
- I certificati, come le carte di credito si sono evoluti attraverso le fasi seguenti:
  - Fase pionieristica
  - Fase in-house
  - Fase di accettazione incrociata
  - Fase di consolidazione
- L'emittente della carta di credito si assume la responsabilità ultima del pagamento
- Le carte di credito si sono dimostrate un business

Normativa italiana

- DPCM 8 febbraio 1999: Regole tecniche per la formazione, la trasmissione, la conservazione, la duplicazione, la riproduzione e la validazione, anche temporale, dei documenti informatici ai sensi dell'art. 3, comma 1, del Decreto del Presidente della Repubblica 10 novembre 1997, n. 513.
- Direttiva 1999/93/CE del Parlamento europeo e del Consiglio del 13 dicembre 1999 relativa ad un quadro comunitario per le firme elettroniche (G.U. delle Comunità europee L. 13 del 13 dicembre 1999)

I mille problemi della firma digitale

- Difficoltà di accesso allo strumento (usabilità)
- Babele dei sistemi di firma (interoperabilità)
- Incompatibilità dei sistemi di verifica (interoperabilità)
- Dubbi sulle garanzie di paternità (identificazione, sospensione, revoca)
- Precarietà nel tempo degli effetti della firma digitale (non-ripudio)
- Assenza di un rigoroso sistema di conservazione del documento sottoscritto (non ripudio)

I mille problemi della firma digitale

- **Difficoltà di accesso allo strumento** (usabilità)
  - pochi certificatori
  - scarsa assistenza
  - nessuna formazione
- **Babele dei sistemi di firma** (interoperabilità)
- **Incompatibilità dei sistemi di verifica** (interoperabilità)
- **Dubbi sulle garanzie di paternità** (identificazione, sospensione, revoca)
  - "accertamento con certezza"
  - accesso semplice, on-line ed in tempo reale alle CRL

I mille problemi della firma digitale

- **Precarietà nel tempo degli effetti della firma digitale** (non-ripudio)
  - marcatura temporale (art. 60, Reg. Tec.)
- **Assenza di un rigoroso sistema di conservazione del documento sottoscritto** (non ripudio)
  - servizio di deposito documentale (art. 59, Reg. Tec.)
  - tra 50, 100, 200 anni sarà possibile (leggere e) verificare un documento digitale sottoscritto?

Verifica della firma digitale

- **Busta elettronica** (PKCS#7) contenente il documento D

$$\langle D, S_A(h(D)), Cert_A \rangle$$

- **Verifica**
  - apertura della busta
  - verifica di $S_A(\ldots)$
  - verifica di $Cert_A$
    - il certificatore è abilitato (presente nella lista del CNIPA)
    - il certificato non deve essere
      scaduto, revocato, o sospeso
      (art. 23, TU sulla documentazione amministrativa)

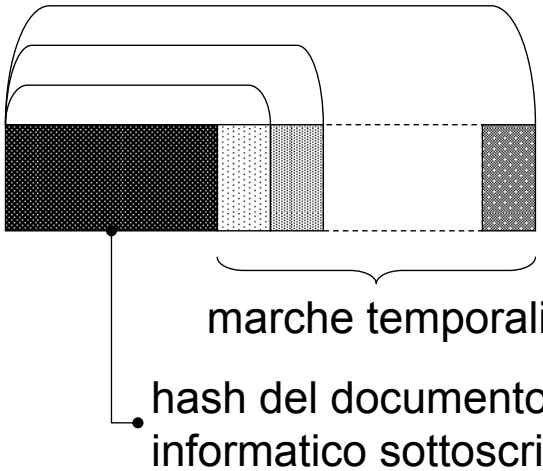
Marcatura temporale come garanzia di validità

- Il documento sottoscritto con firma digitale diventa a tutti gli effetti un documento sottoscritto con la stessa efficacia di una firma autografa
- Un documento deve essere conservato nel tempo
- Un documento sottoscritto con firma digitale ha, per sua natura, una durata limitata nel tempo
  - un certificato ha una durata limitata bel tempo (art. 4, Reg. Tec.) non superiore a tre anni (art. 22, comma f. TU)
  - un certificato può essere revocato (art. 29, comma 1)
  - un certificato può essere sospeso (art. 33)
- Chi vuol far valere l'efficacia del documento sottoscritto con firma digitale deve provare che questo è stato sottoscritto quando il certificato era valido (art. 23, TU) (*continua*)

Marcatura temporale come garanzia di validità

- In mancanza di una prova di anteriorità rispetto ad un evento pregiudizievole (scadenza, revoca, sospensione), il documento non avrebbe più efficacia come documento sottoscritto (art. 60, Reg. Tec.) potendosi immaginare almeno un'efficacia di riproduzione meccanica (art. 2712 c.c.)
- L'art. 60 del Reg. Tec. indica la marcatura temporale come mezzo per provare la **data certa** di un documento informatico

Estensione temporale della validità di una firma



- La marca temporale contiene varie informazioni (art. 53, reg. tec.) tra cui la data e l'ora della sua creazione (art. 55 ss., reg. tec.)
- La hash del documento informatico e e la marca temporale sono sottoscritti con la chiave di marcatura temporale del fornitore del servizio
- La marca viene restituita al richiedente entro un minuto

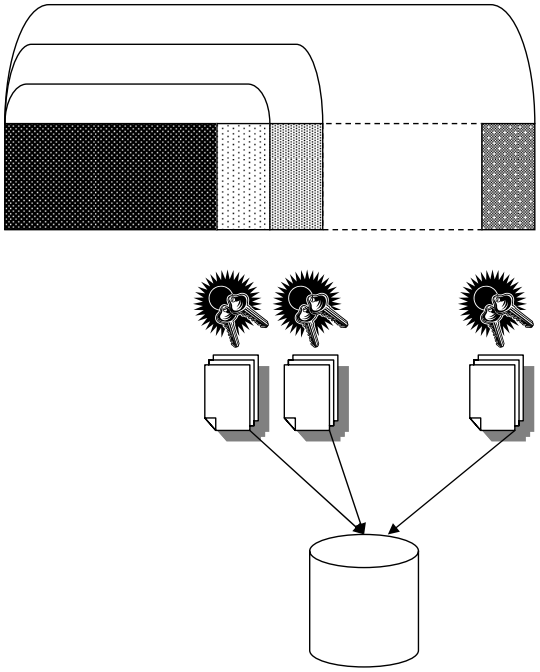
- Le chiavi di marcatura temporale sono di 1024 bit ed hanno una durata di un anno

Estensione temporale della validità di una firma

- **Alcuni problemi con il sistema di marcatura temporale**
- È complicato accertare il momento della scadenza
- Incompatibilità dei sistemi di verifica
- Costi
- Brevità delle marche temporali

Estensione temporale della validità di una firma

Ricostruzione del trust passato



- È necessario poter accedere alle vecchie chiavi, ai vecchi certificati ed alle vecchie CRL per poter verificare la catena delle marche temporali
- Un modo più semplice e meno costoso per garantire l'associazione tra documento e marca temporale: il certificatore potrebbe fornire un servizio di conservazione documentale