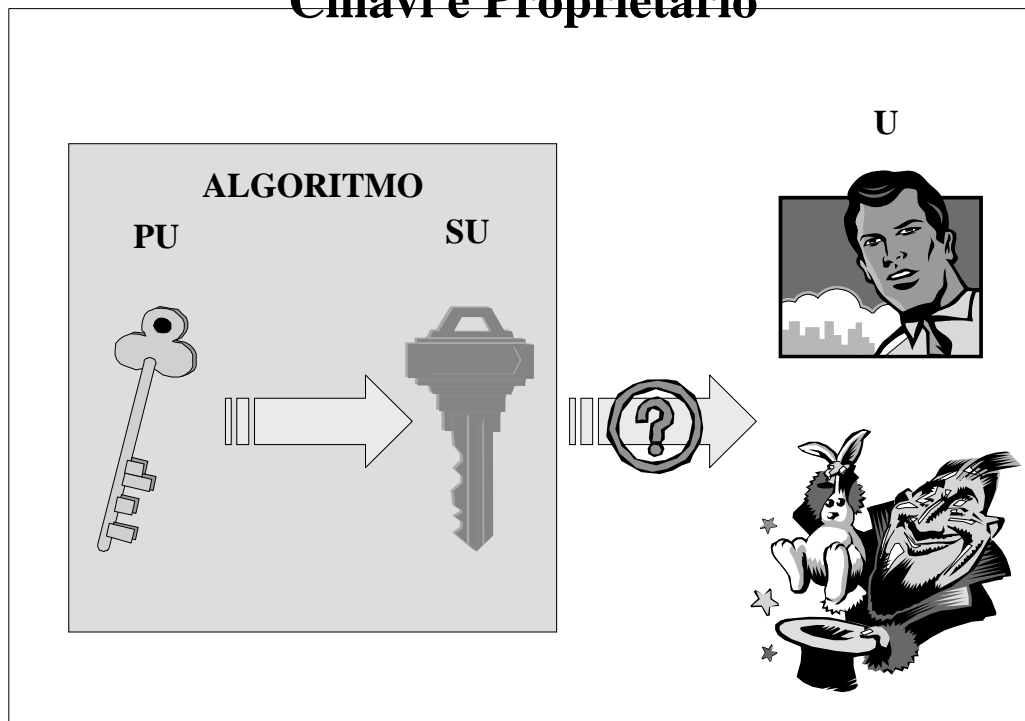


## Autenticità della chiave pubblica

- l'attacco dell'uomo in mezzo
- il certificato
- la PKI

## Chiavi e Proprietario



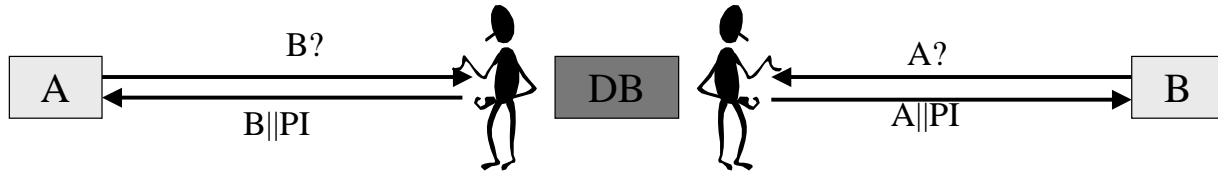
□ R28: “prima d’impiegare una chiave pubblica bisogna o essere certi dell’identità del suo proprietario o poterla verificare”

# Attacco dell'uomo in mezzo

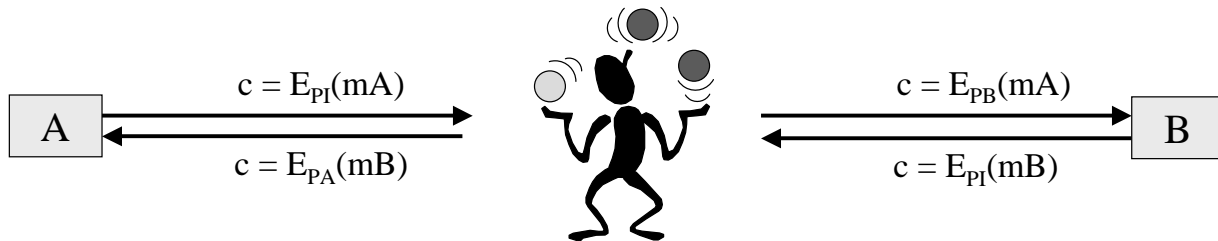
1 - Registrazione



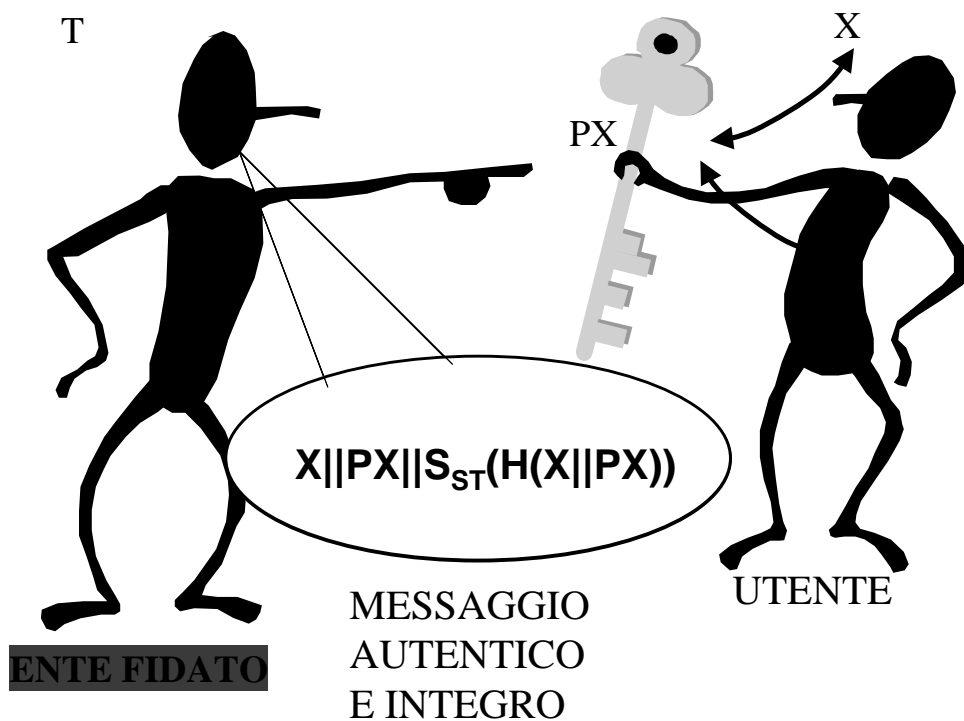
2 - Intercettazione delle interrogazioni e falsificazione delle risposte



3 - Intercettazione, decifrazione, cifratura ed inoltra.



# Autenticazione di una chiave pubblica



# Preparazione ed uso di un certificato

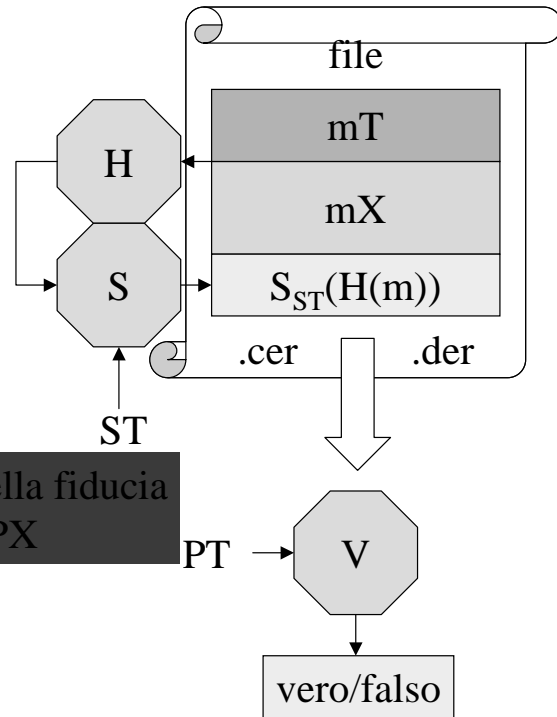
X identificato in modo sicuro

- $mX = X || X || PX || PX$
- $mT = T || IT$
- $m = mT || mX$

- $H(m)$
- $S_{ST}(H(m))$
- $Cert(PX, T) = m || S_{ST}(H(m))$

PT nota in modo certo

- $V_{PT}(Cert(PX, T))$

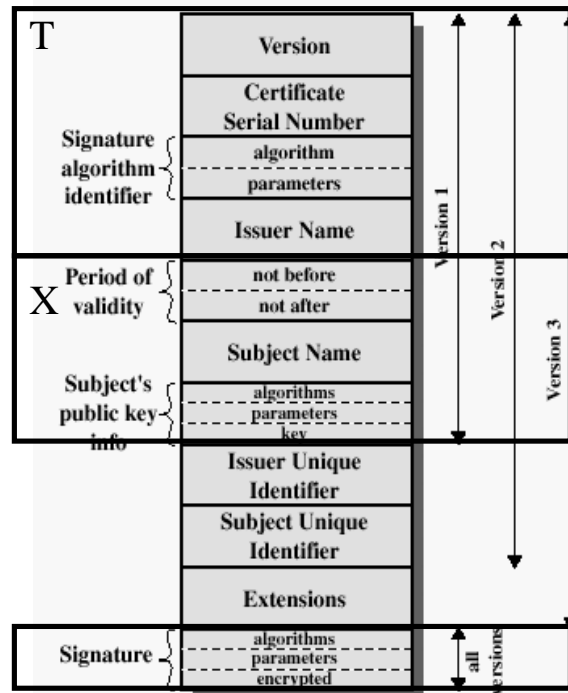


## Il Certificatore

T?

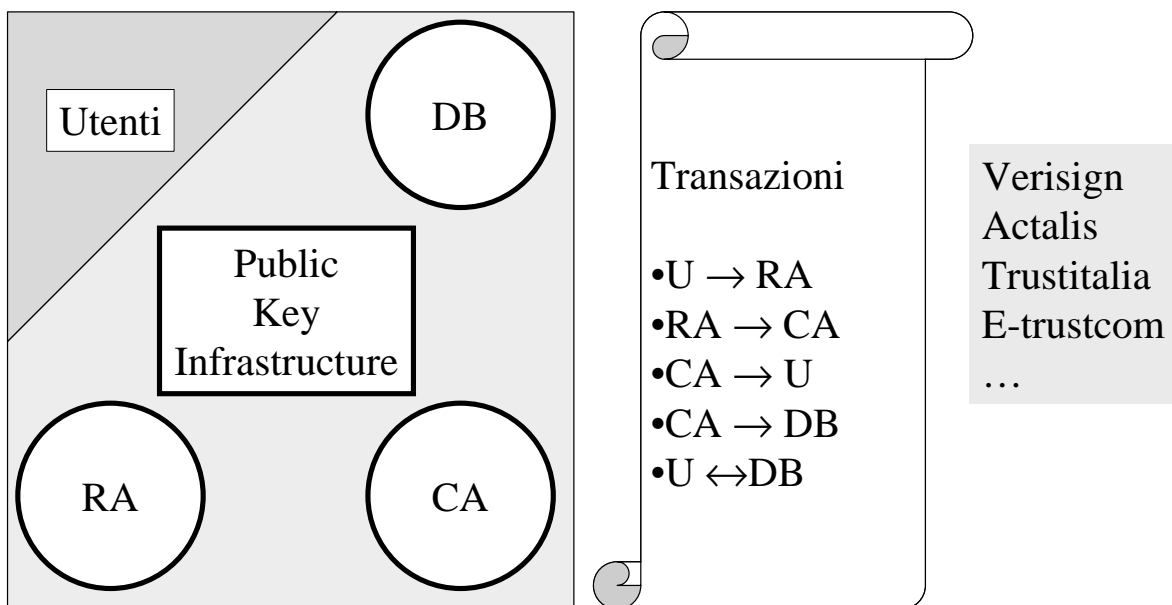
- ENTE ufficialmente riconosciuto  
Certification Authority o CA (X.509)
- Qualsiasi UTENTE  
(PGP)

# X.509 Formats

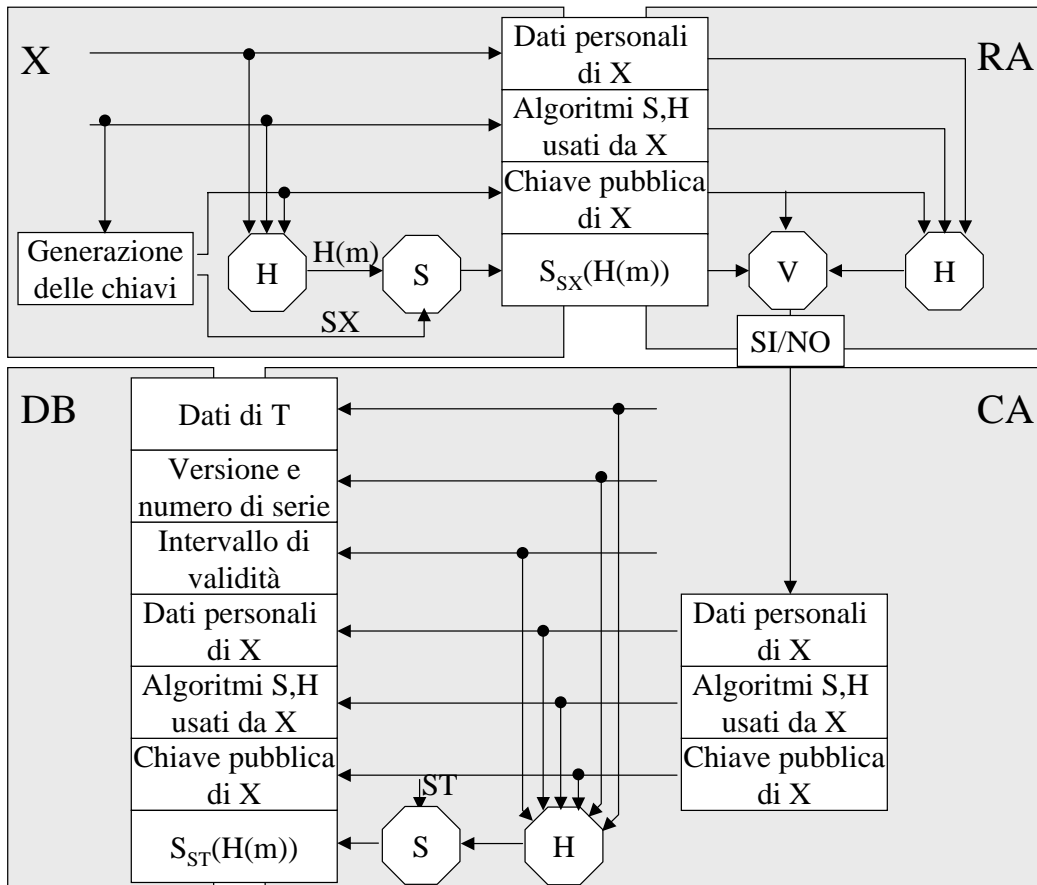
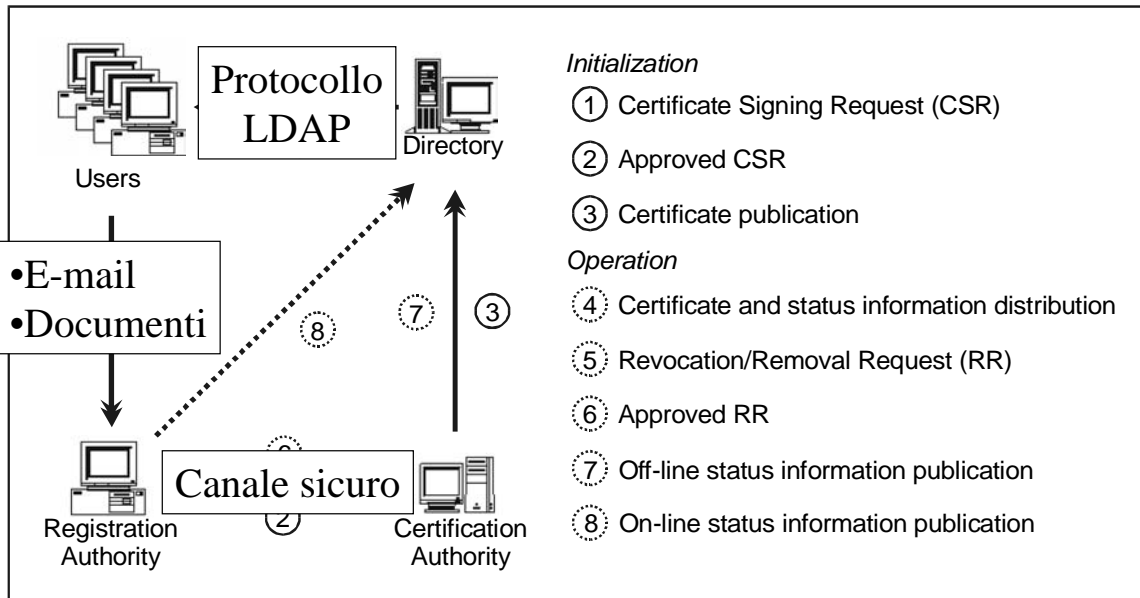


(a) X.509 Certificate

## PKI: RA, CA e Directory



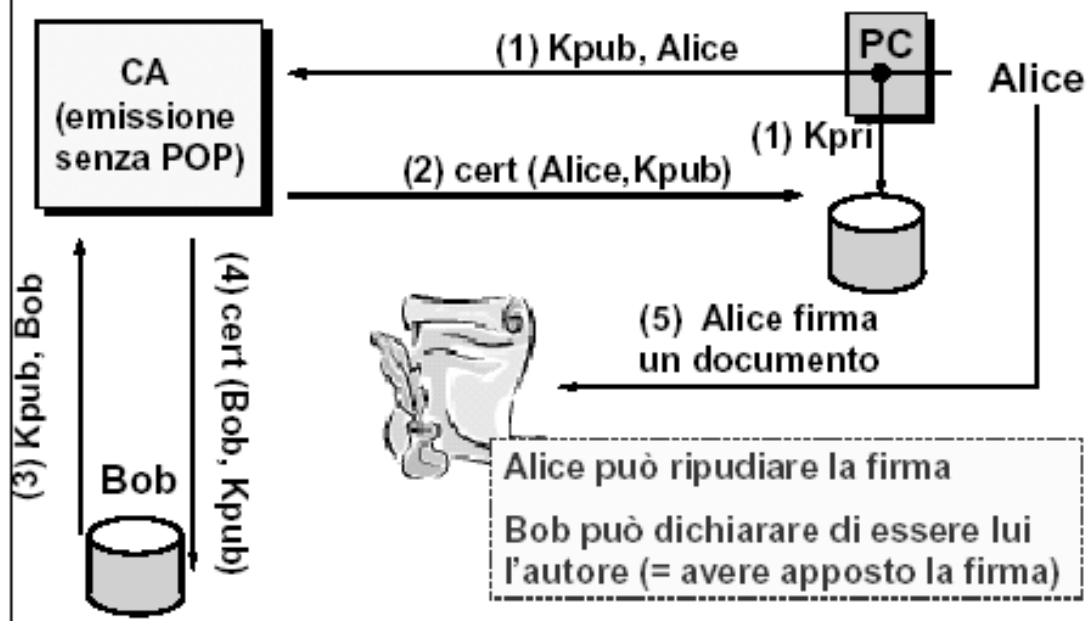
# PKI: RA, CA e Directory



## Proof-Of-Possession (POP)

- POP = la CA ha sufficienti garanzie circa il possesso della chiave privata da parte dell'entità che richiede un certificato (il Subject)
- emissione di certificati senza POP può permettere vari attacchi
  - situazione diversa a seconda dell'uso della chiave (POP fondamentale per garantire il non-ripudio)
  - POP non sempre critico nel caso di cifratura

## Assenza di POP-possibili rischi



## Contromisure

- **metodo migliore: POP a tempo di firma**
  - il firmatario inserisce un riferimento al certificato (es. un hash) tra le informazioni firmate
  - la firma è funzione quindi anche del certificato
  - attualmente non supportato dai protocolli di sicurezza
- **soluzione alternativa: la CA emette un certificato solo se ha la prova che il richiedente possiede la chiave privata**

## Metodi per il POP (I)

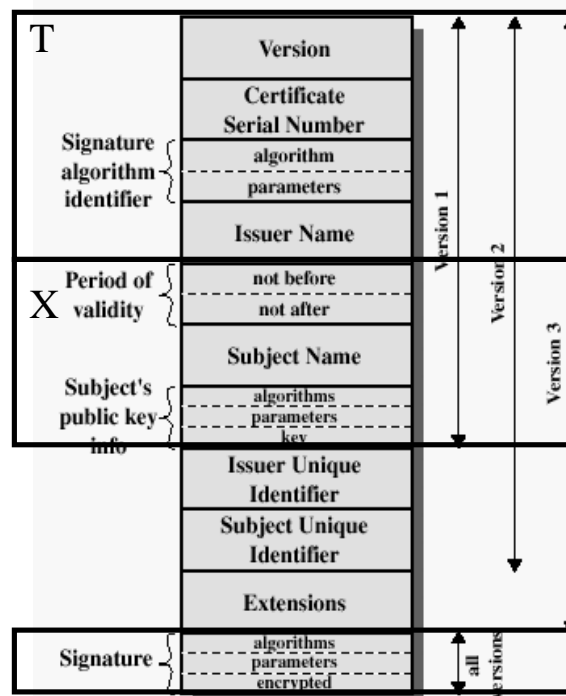
- **metodi OOB**
  - chiavi generate da CA/RA e consegnate in token sicuri (es. smart-card, USB crypto-token); fa quindi fede il possesso del token
  - politiche di key-recovery/key-backup (molto rischioso!!!): la CA mantiene una copia di tutte le chiavi private – come le protegge efficacemente?

## Metodi per il POP (II)

### ■ metodi on-line

- per chiavi di firma e cifratura: possibile usare formati auto-firmati (PKCS-10, SPKAC): la CA verifica la firma ed ottiene il POP
- per chiavi di cifratura (no firma)
  - utilizzo di protocolli challenge - response che comportino un'operazione di decifratura (=uso della chiave privata)
  - certificato restituito in forma cifrata (successiva revoca se il certificato non è usato)

## X.509 Formats



(a) X.509 Certificate



# Standard Certificate Extensions (1.)

- *version 3 introduces a mechanism whereby certificates can be extended, in a standardized and generic fashion, to include additional information;*
- *certificates are not constrained to only the standard extensions and anyone can register an extension with the appropriate authorities (e.g., ISO);*
- *standard extensions for public key certificates can be separated into the following groups:*
  - *key information;*
  - *policy information;*
  - *user and CA attributes;*
  - *certification path constraints*

# Standard Certificate Extensions (2.)

- ***authority key identifier***: *specifies a unique identifier of the key pair used by the CA to sign the certificate;*
- ***subject key identifier***: *serves much the same purpose as the authority key identifier;*
- ***key usage***: *specifies the intended use(s) of the key. The following list represents the settings for the key usage field: certificate signing (e.g., a CA key pair), CRL signing, digital signature, symmetric key encryption for key transfer, data encryption (other than a symmetric key);*
- ***private key usage period***: *specifies the date on which the signing private key expires for a user's digital signature key pair*

# Standard Certificate Extensions (3.)

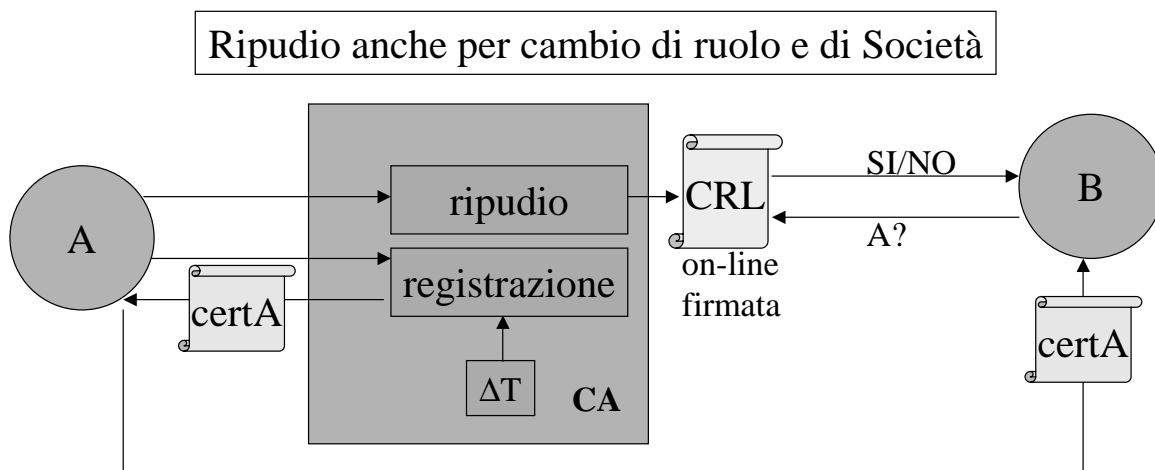
- **subject alternative name:** specifies one or more unique names for the certificate subject; the permissible name forms are Internet e-mail address, Internet IP address, , web URL
- the policy information extensions provide a mechanism for the CA to distribute information regarding the ways a particular certificate should be used and interpreted;
- **certificate policies:** specifies the policies under which the certificate was issued to the user and/or the types of uses applicable to the certificate; certificate policies are represented by specially-formatted numbers, known as object identifiers;

## Vita e ripudio di una chiave pubblica

R29: “Quando uno ha il sospetto, o la certezza, che la sua chiave segreta sia stata violata, deve

- rinunciare ad impiegarla,
- notificare immediatamente il ripudio alla CA
- registrarne una nuova”.

Per prevenire il rischio CA da una vita limitata ad ogni chiave.



# Modelli di Notifica di Revoca

- ❑ pull method
- ❑ push model
- ❑ online status checking

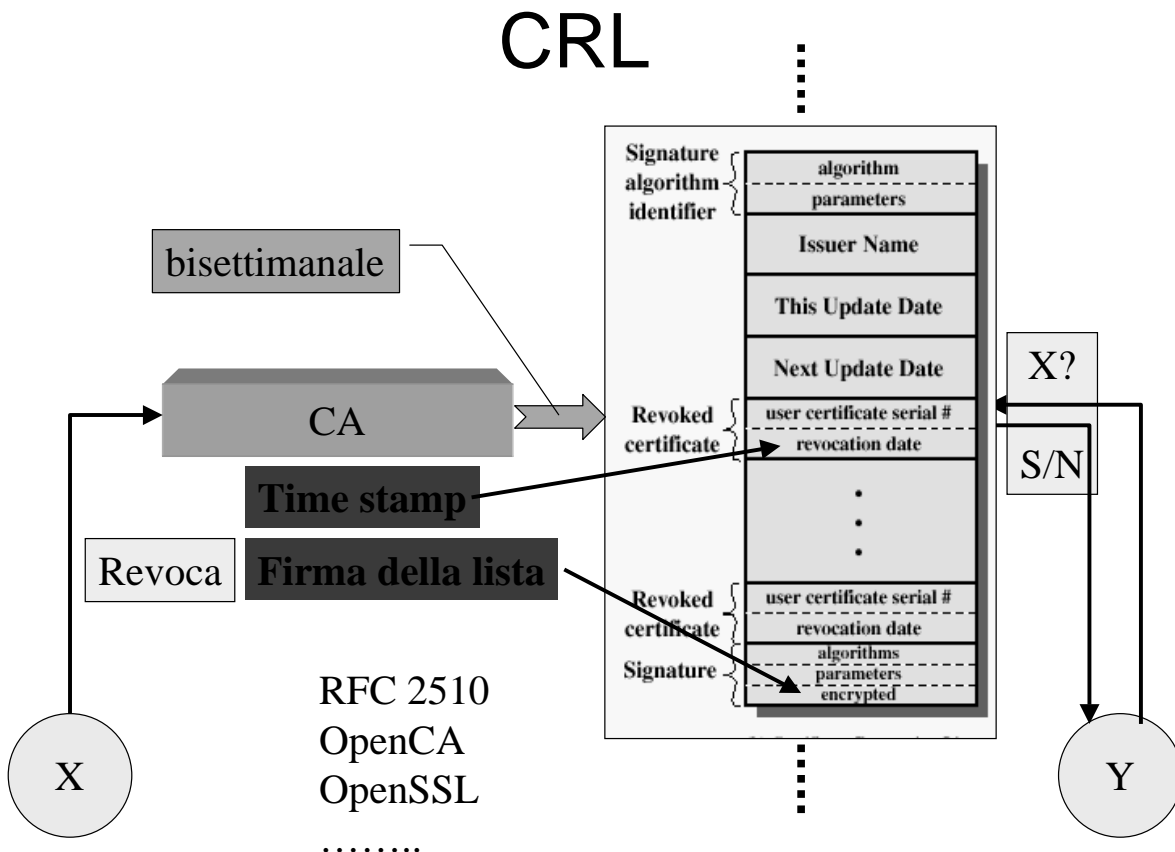
## Schemi di Notifica della Revoca:

### Schemi off-line:

Certificate Revocation List  
Certification / Revocation System  
Certificate Revocation Tree  
...

### Schemi on-line:

On-line Certificate Status Protocol



# Estensioni delle CRL

- **general extensions:** *CRL number, reason code*
  - *key compromise;*
  - *CA compromise;*
  - *affiliation change;*
  - *cessation of operation;*

## Come ridurre la dimensione delle CRL?

- **partizionamento delle CRL (usando opportunamente il CRLDP):**
  - cert con SN < 1000 hanno CRLDP= crl\_1\_1000.der
  - cert con 1000 < SN < 2000 hanno CRLDP= crl\_1001\_2000.der
- **usando deltaCRL**
  - una CRL base (n. N)
  - una o più delta CRL (differenza rispetto a N)

## Gestione efficiente delle CRL

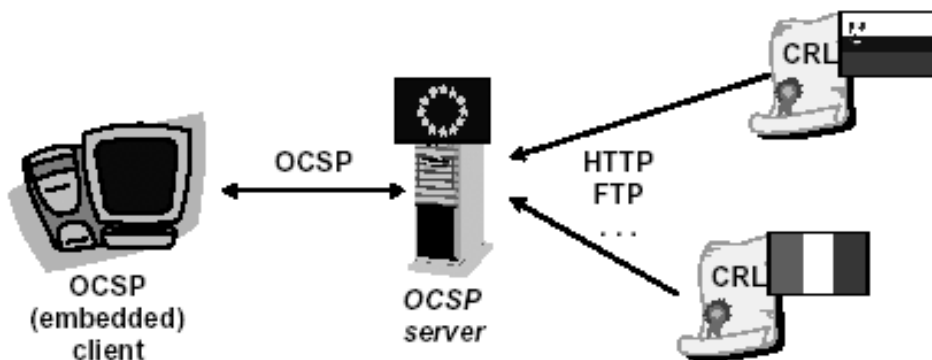
- **problema: le CRL possono diventare molto grosse e quindi onerose da scaricare e da esaminare**
- **varie soluzioni:**
  - eliminare la revoca dopo la prima CRL successiva alla scadenza del certificato
  - pubblicare CRL complete (Base CRL) e poi solo le differenze (Delta CRL)
  - partizionare le CRL in tanti gruppi (es. per ogni mille certificati emessi) usando CRL DP

## OCSP

- **RFC-2560: On-line Certificate Status Protocol**
- **standard IETF-PKIX per verificare in linea se un certificato è valido:**
  - good
  - revoked
    - revocationTime
    - revocationReason
  - unknown
- **risposte firmate dal server (non dalla CA!)**
- **certificato del server non verificabile con OCSP!**

## Architettura di OCSP

- **possibili risposte pre-calcolate**
  - diminuisce il carico sul server ... ma rende possibili attacchi di tipo replay!
- **possibile attingere informazioni non da CRL**



## Modelli di OCSP responder

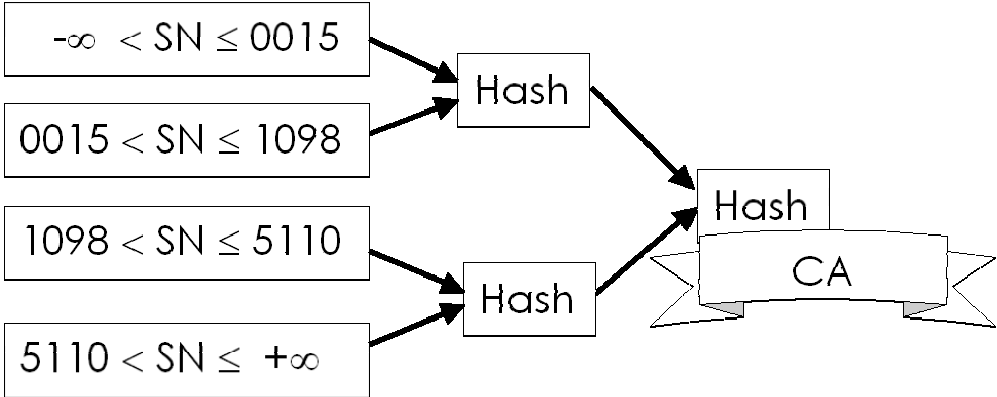
- **Trusted Responder**
  - il server OCSP firma le risposte con una coppia chiave:cert indipendente dalla CA per cui sta rispondendo
  - responder aziendale o TTP pagata dagli utenti
- **Delegated Responder**
  - il server OCSP firma le risposte con una coppia chiave:cert diversa in base alla CA per cui sta rispondendo
  - TTP pagata dalle CA

# Certification/ Revocation System

| Serial Number | Status            |
|---------------|-------------------|
| 00000001      | OK(SN,t,CA)       |
| 00000002      | OK(SN,t,CA)       |
| ...           | OK(...,....,....) |
| 13434899      | OK(SN,t,CA)       |
| 13434900      | REV(SN,CA)        |
| 13434901      | OK(SN,t,CA)       |
| ...           | OK(...,....,....) |

Certificates status at time  $t$

# Certificate Revocation Tree



# Performance Evaluation Criteria

- Timeliness
- Involved computational load
- Communication traffic induced on the network

| PARAMETRO             | PAROLE CHIAVE       |   |
|-----------------------|---------------------|---|
| Prestazioni           | Lato Amministratore | Picco di Carico e Picco di Richiesta<br>Carico Medio e Richiesta Media<br>Distribuzione del Carico<br>Ritardo<br>Dimensioni |
|                       | Lato Utente         | Dimensioni<br>Ritardo Massimo<br>Carico Computazionale<br>Banda   |
| Tempestività          |                     | Tempo Massimo tra revoca e distribuzione  |
| Scalabilità           | Lato Amministratore | Complessità dello schema  |
| Sicurezza             |                     | Autenticità<br>Integrità<br>Confidenzialità<br>Non-Ripudio  |
| Standard              |                     | Standard<br>Proprietario<br>Teorico<br>Implementato   |
| Espressività          |                     | Granularità dell'informazione di revoca   |
| Gestione dello schema | Lato Amministratore | Automatizzato<br>Archiviazione sicura<br>Complessità  |
| On-line vs. Off-line  | Lato Amministratore | Frequenza delle connessioni   |



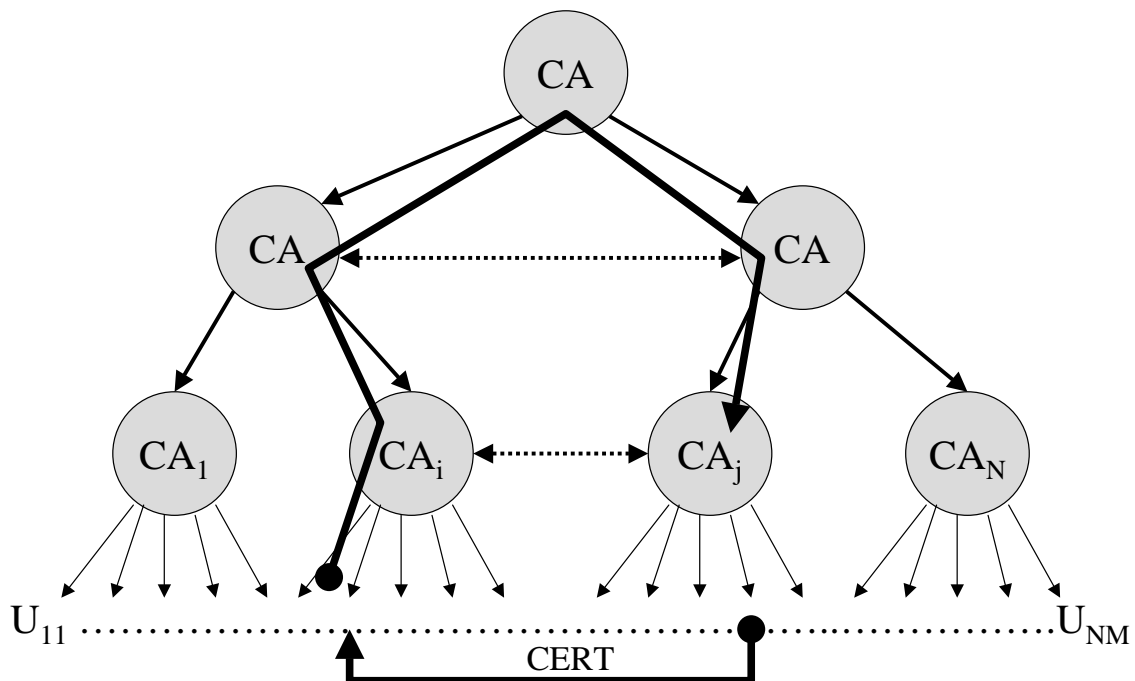
# Problemi di PKI

- RA sempre disponibile
- CA rapida anche nella gestione della CRL
- Collo di bottiglia (n° max di utenti)
- Ente degno di fiducia
- Interrogazione della CRL
- Vita della chiave di firma

## Gerarchia di Autorità di Certificazione

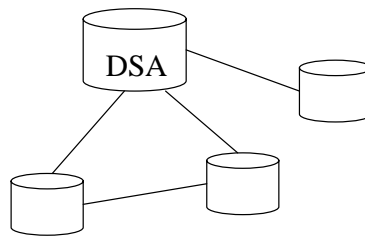
DPR 513/97

AIPA → [cnipa.gov.it](http://cnipa.gov.it)



# What is a Directory?

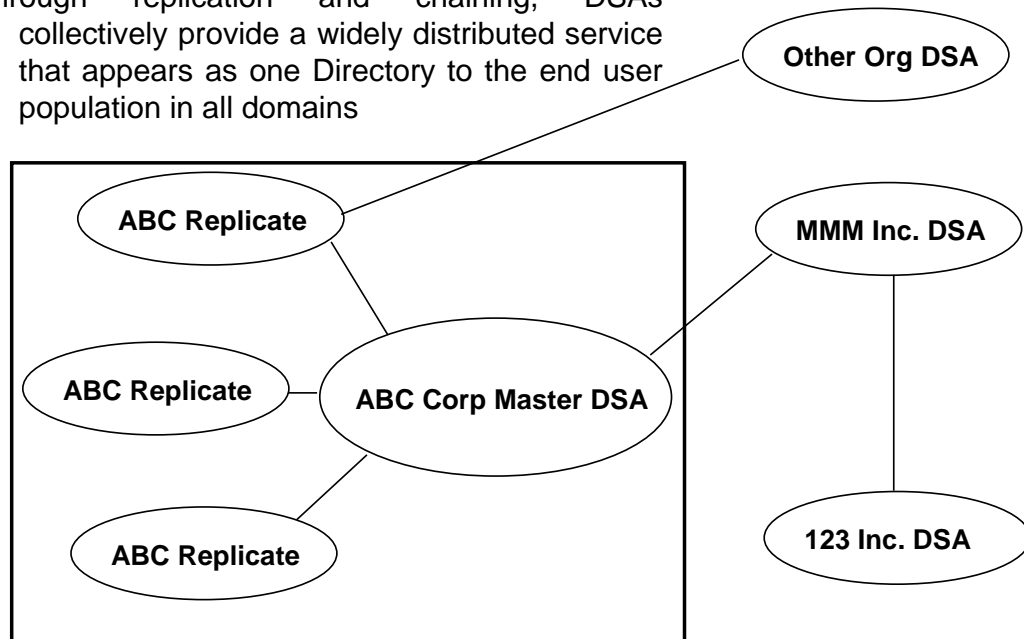
- *Intended as a global repository of information*
  - *Used for:*
    - *electronic mail*
    - *cryptographic exchanges*
    - *telephone systems*
- *In essence the Directory is a distributed database, capable of storing information about people and objects in various nodes or servers distributed across a network*



## X.500 - The Directory

### DUAs and DSAs

Through replication and chaining, DSAs collectively provide a widely distributed service that appears as one Directory to the end user population in all domains



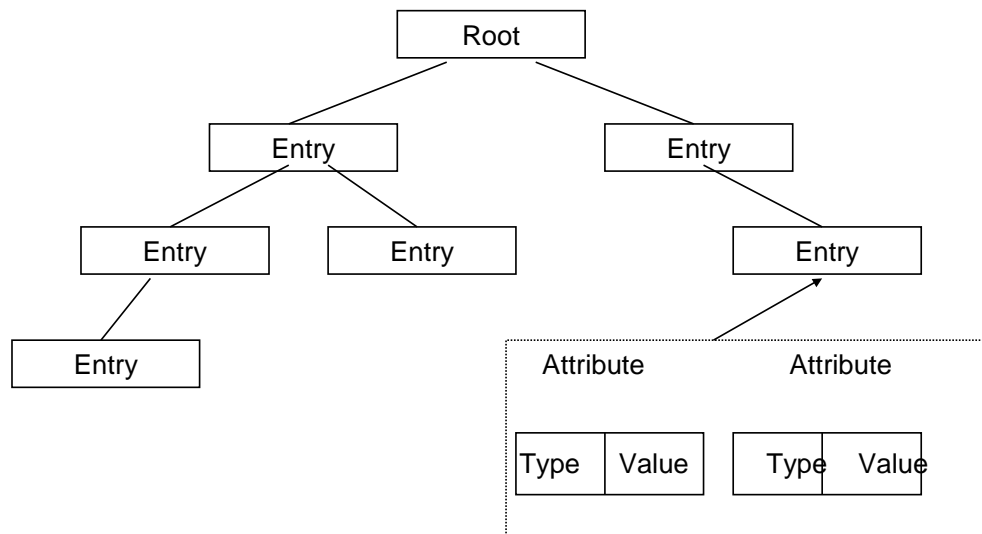
# Information Model

- *determines the form and character of information in the directory;*
- *the information model is centered around **entries**, which are composed of attributes;*
- *each **attribute** has a type and one or more values;*
- **object class:** *defines which attributes are required and allowed in an entry;*
- *entries are arranged in a tree structure and divided among servers in a geographical and organizational distribution*

|                  |                   |
|------------------|-------------------|
| Common Name      | Rebecca Montanari |
| Telephone Number | 02/57401235       |

} Entry

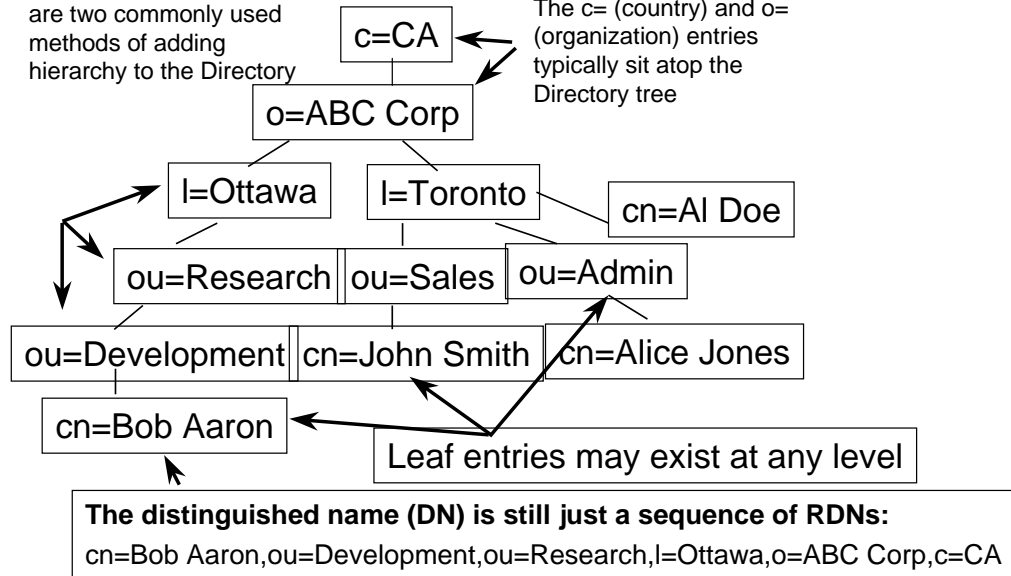
## Directory Structure (1.)



# Directory Structure (2.)

l= (locality) and ou= (organizational unit) entries are two commonly used methods of adding hierarchy to the Directory

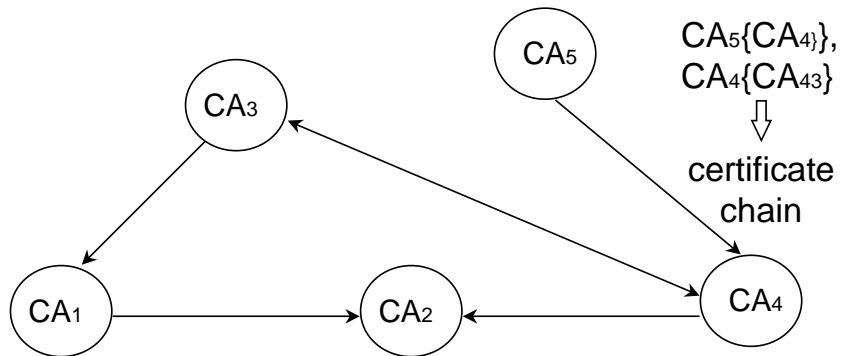
The c= (country) and o= (organization) entries typically sit atop the Directory tree



# Trust Models

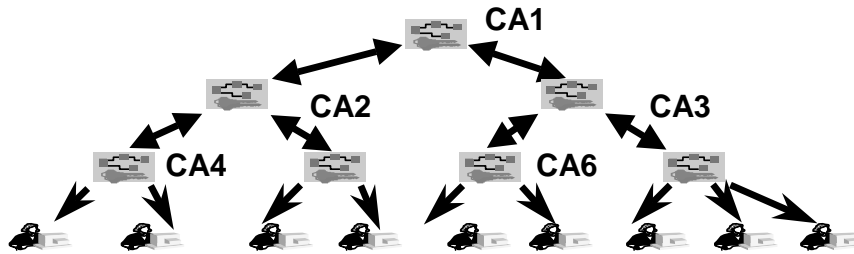
certificate chains and certification paths:

$A\{P_5\} \Rightarrow B\{P_3\}$



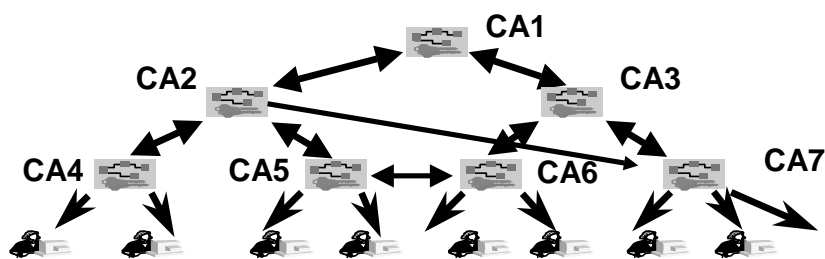
- **centralized** trust model;
- **distributed** trust model

# General Hierarchical Structure



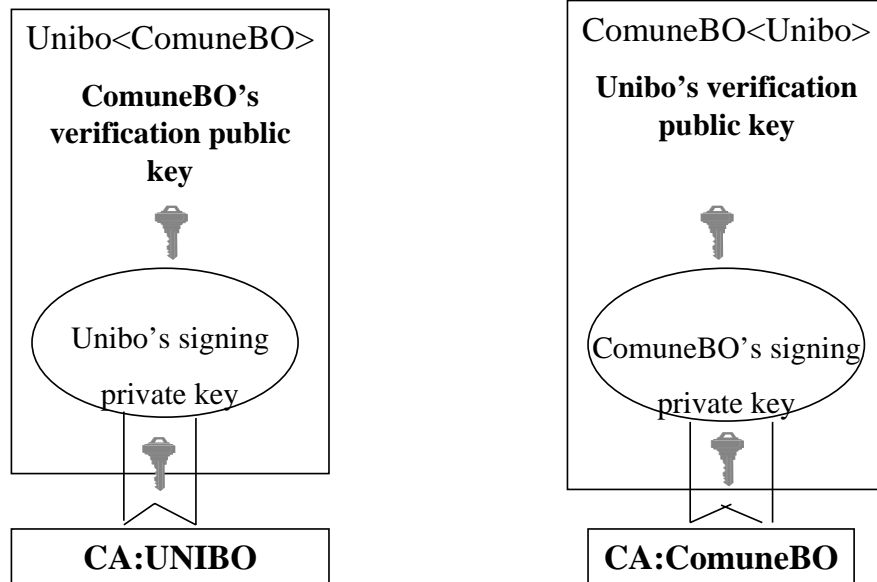
- *it is easy to construct a certification path between any pair of end-entities, regardless of how each end-entity determines which CA it is prepared to accept as root CA;*
- *this model scales reasonably well; provides means for constructing reasonably short certification paths;*
- *complicating factor is trust*

# General Hierarchical Structure with additional links

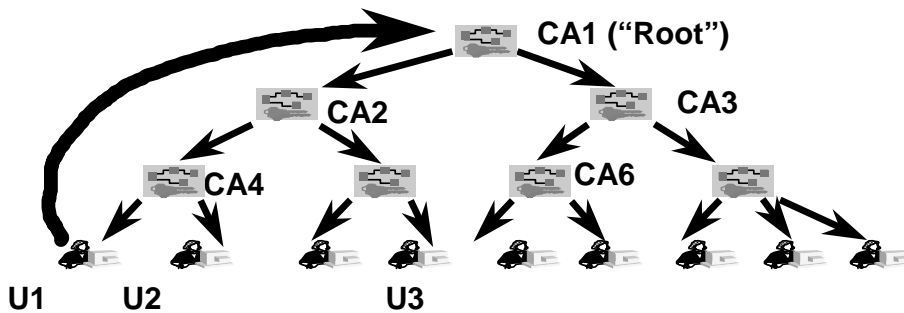


- *added links are called cross-certificates*

# Cross-Certificates



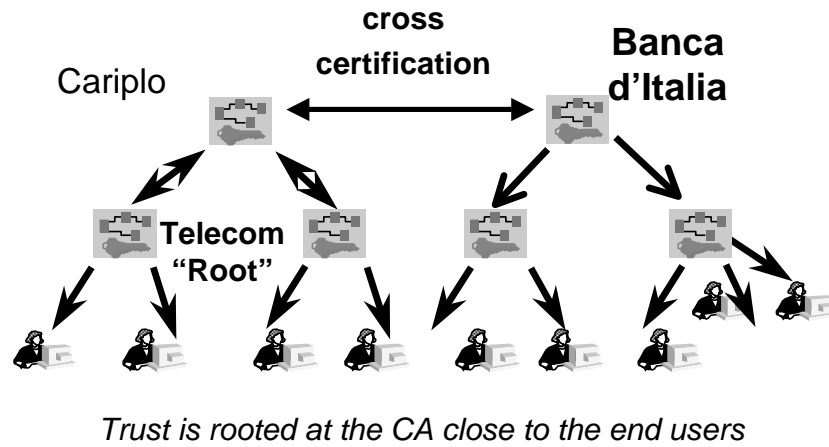
# Top-down Hierarchical Structure



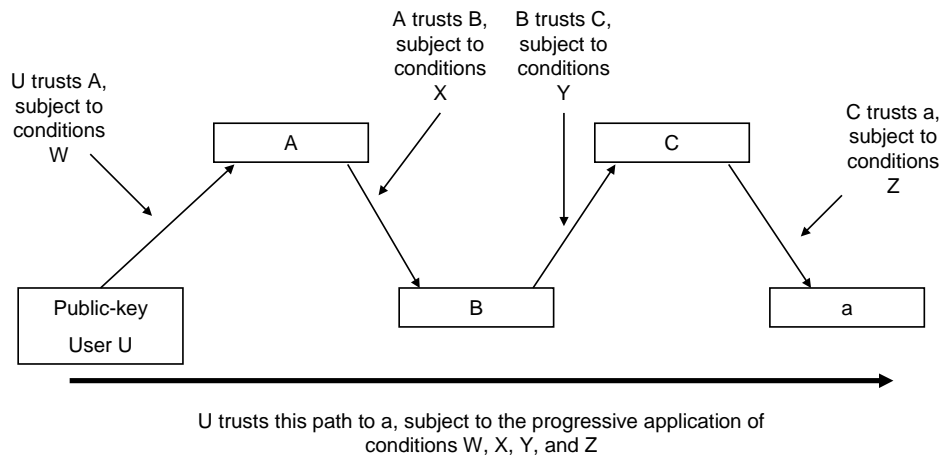
### Drawbacks:

- all trust depends on the root key;
- certificate chains also for two entities on the same CA;
- certificate chains long in deep hierarchies.

# Distributed Trust Model



# Progressive-Constraint Trust Model



# X.509 Certificate Policies

- **certificate policy:** a named set of rules that indicates the applicability of a certificate to a particular community and/or class of application with common security requirements;
- **policy mapping:** only applies to cross-certificates; provides a mechanism for the signing CA to map its policies to the policies of the CA specified in the cross-certificate
- **policy constraints:** it is used in cross-certificates the administrator can specify the set of acceptable policies in a certificate chain extending from a cross-certificate; can specify whether or not all certificates in a chain must meet a specific policy;
- .....

## Certification Path Discovery

- the certification path discovery problem is as follows: I need to find a certification path between a particular remote user's public key and any one of the set of root public key that I already know securely
- given a CA's name, a service to retrieve certificates for that CA's public keys issued by other Cas, it is possible to find a certification path by working back progressively from the target user's certificate toward a root key, as follows:
  - **step 1:** given a certificate issued by CA X, determine the set of Cas that have issued certificates for the public key of X;
  - **step 2:** if one of the CAs from the Step 1 is a known root authority, the required certification path is found, otherwise proceed to Step3;
  - **step 3:** for each CA found in Step 1, repeat the Step 1 procedure, treating that CA as CA X



# Certification Path Validation

- *given that a suitable certification path has been found, it is then necessary to validate that path. This involves such actions as:*
  - *verifying the digital signature on each certificate;*
  - *checking that the names in the certificates are consistent with a valid certification path, that is, the subject of every certificate is the issuer of the next certificate;*
  - *checking that the validity periods of all certificates correctly span the time for which validity is being checked;*
  - *checking that each certificate has not been revoked. This may be a complex process;*
  - *checking that the required certificate policies are indicated in the certificates;*

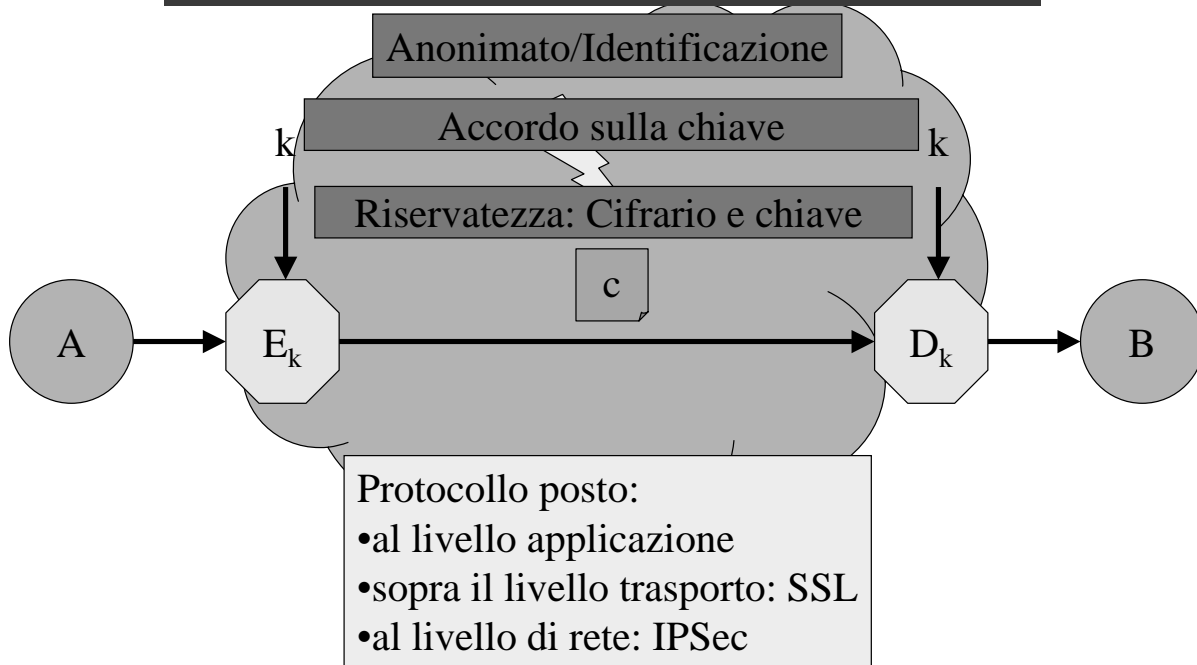


## **Certificati e reti sicure**

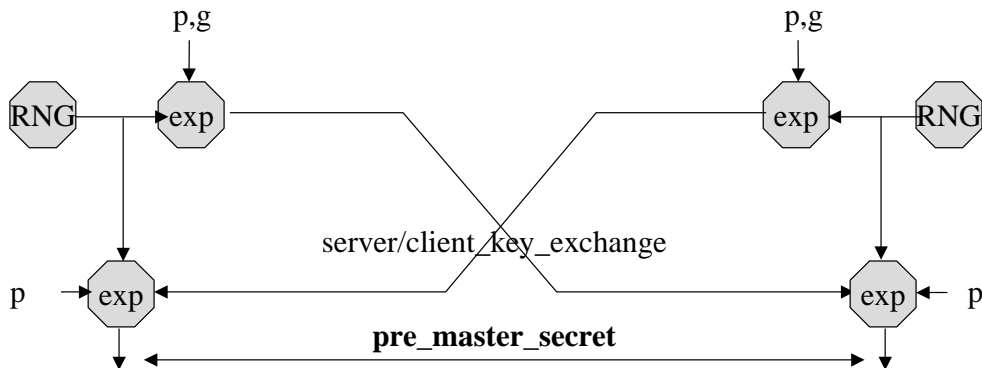
- **fixed D-H**
- **ephemeral D-H**
- **IKE**

# I certificati e le reti sicure

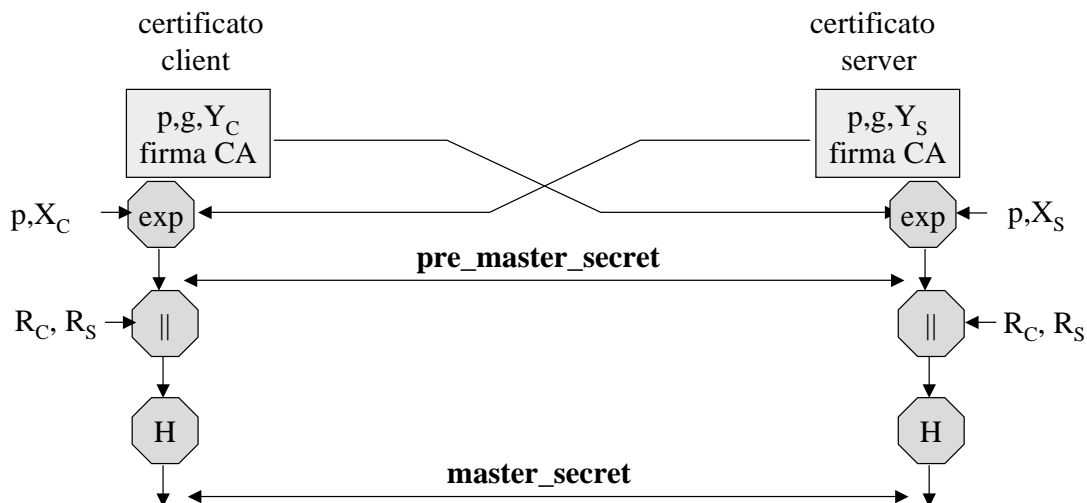
Problema: A e B vogliono scambiarsi informazioni sicure in assenza di accordi precedenti



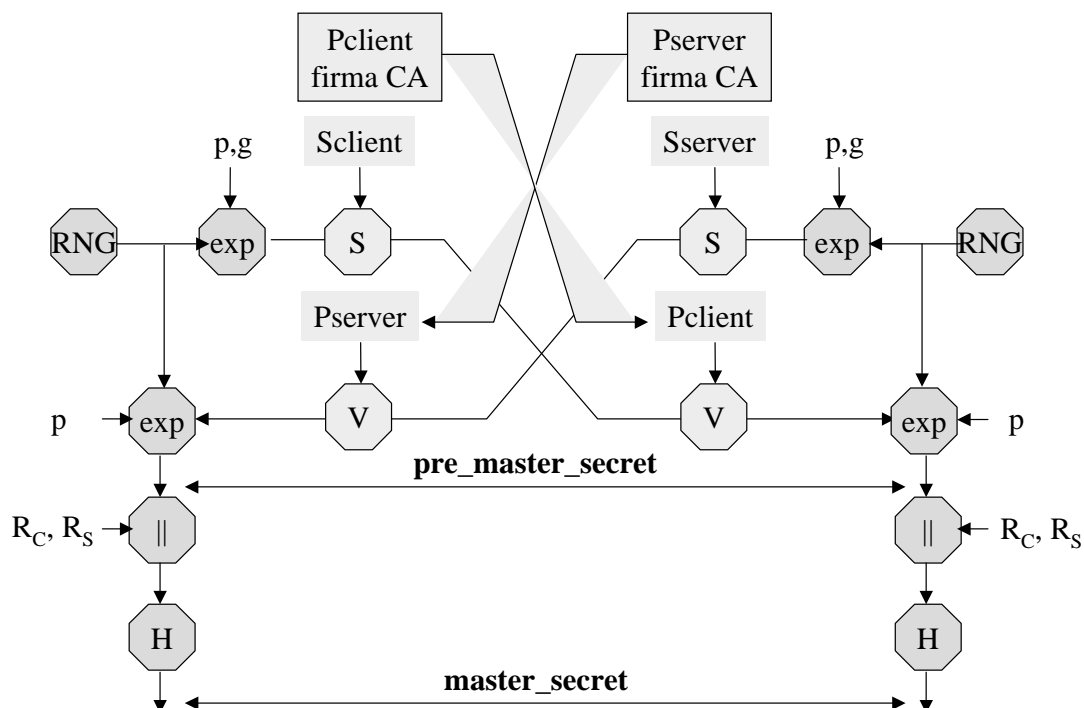
## L'accordo sul segreto: anonymous Diffie-Hellman



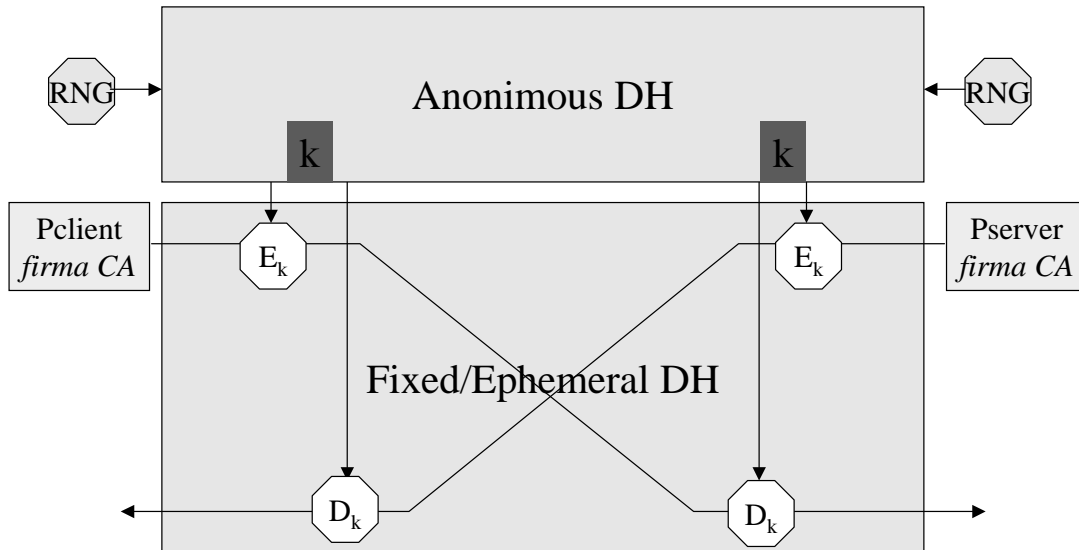
# L'accordo sul segreto: fixed Diffie-Hellman



# L'accordo sul segreto: ephemeral Diffie-Hellman



# Accordo sul segreto e privacy: IKE (Internet Key Exchange)



## IKE: anonimato ed identificazione

