Sistemi Distribuiti
Corso di Laurea in Ingegneria

Prof. Paolo Nesi

PARTE 15: Sicurezza in Rete

Department of Systems and Informatics

University of Florence
Via S. Marta 3, 50139, Firenze, Italy
tel: +39-055-4796523, fax: +39-055-4796363

Lab: DISIT, Sistemi Distribuiti e Tecnologie Internet

nesi@dsi.unifi.it, paolo.nesi@unifi.it
http://www.disit.dsi.unifi.it/

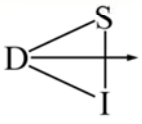
Sicurezza in rete

- Introduzione
- Tecnologie per la sicurezza
- Algoritmi di crittografia
- Firme digitali
- Scenari ed applicazioni

Evoluzione della Sicurezza

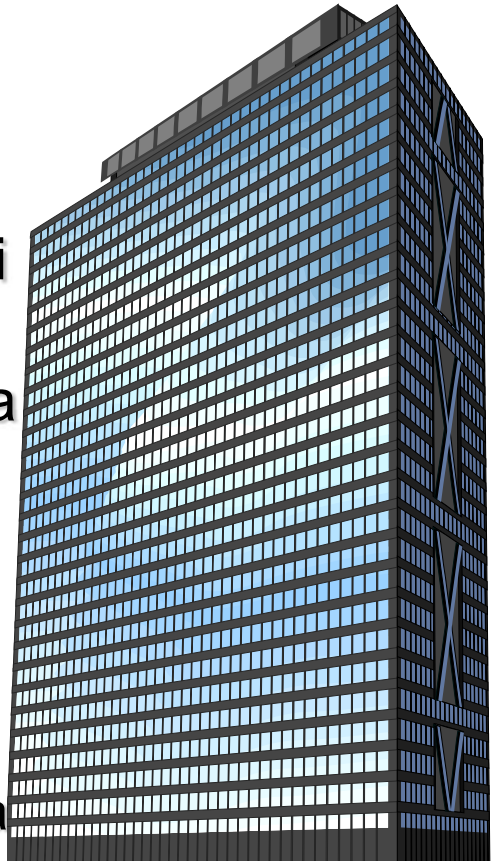
	1965-75	*1975-89*	*1990-99*	*Current*
Platforms	Multi-user timesharing computers	Distributed systems based on local networks	The Internet, wide-area services	The Internet + mobile devices
Shared resources	Memory, files	Local services (e.g. NFS), local networks	Email, web sites, Internet commerce	Distributed objects, mobile code
Security requirements	User identification and authentication	Protection of services	Strong security for commercial transactions	Access control for individual objects, secure mobile code
Security management environment	Single authority, single authorization database (e.g. /etc/ passwd)	Single authority, delegation, repli-cated authorization databases (e.g. NIS)	Many authorities, no network-wide authorities	Per-activity authorities, groups with shared responsibilities

- Con la **rete globale** sono apparse nuove esigenze per garantire la sicurezza dei servizi forniti.

Cosa significa sicurezza

- Sicurezza “informatica” e sicurezza “nel mondo reale”
- Una azienda desidera che le sue risorse siano rese accessibili specificando opportune limitazioni
- Per esempio, desidera che all’edificio dell’azienda abbiano accesso solamente i dipendenti e i visitatori previsti di autorizzazione
- Inoltre vuole definire gruppi di dipendenti in modo da diversificare l’accesso ai documenti
- Meccanismi per controllare l’accesso all’edificio:
  - Il portiere dell’edificio che controlla i badges all’ingresso
  - Badges per dipendenti e visitatori
  - Una guardia di sicurezza e porte a chisura automatica
- Meccanismi per l’accesso ai documenti:
  - Responsabile dell’archivio che controlla il gruppo di appartenenza



Politiche o meccanismi di sicurezza

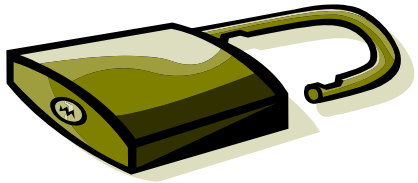
- Le politiche di sicurezza vengono realizzate mediante i meccanismi di sicurezza
- I meccanismi da soli non garantiscono la sicurezza
- Le politiche sono indipendenti dalle tecnologie usate

- La distinzione tra politiche e meccanismi è molto utile in fase di progetto
- Risulta spesso difficile garantire che i meccanismi previsti implementano la politica di sicurezza desiderata

Politiche o meccanismi di sicurezza 2

- Tornando all'azienda…

L'applicazione di una serratura al portone prinicipale dell'edificio non garantisce la sicurezza se non viene associato un'adeguata politica: "la serratura viene chiusa in ogni situazione in cui non sia presente il personale di sorveglianza"



Modello di sicurezza

- Elementi chiave:
  - Processi
  - Risorse
  - Interfacce
  - Client
  - Principals
  - Rete
- I **processi** contengono (*encapsulation*) oggetti (linguaggi di programmazione) e altre **risorse** definite dal sistema
- I processi consentono l'accesso ai **client** attraverso le loro **interfacce**
- I **principals** (utenti o altri processi) possono essere autorizzati a operare su di una determinata risorsa
- I processi interagisco attraverso una **rete** condivisa tra molti utenti

Modello di sicurezza 2

- Elementi chiave:
  - Attackers
  - Messaggi

- I nemici (**attackers**) possono accedere alla rete
- possono leggere o copiare tutti i **messaggi** trasmessi attraverso la rete
- possono arbitrariamente inserire messaggi indirizzati a qualunque destinazione simulando che provengano da una qualunque sorgente della rete

Sistema distribuito: esempi di minacce

- In molte architetture di rete è semplice
  - creare un programma che ottenga copie dei messaggi trasmessi sulla rete
  - se i clients non provvedono ad autenticare il server, che un programma possa inserirsi è spacciarsi per il processo server richiesto cosicché il client trasmetta le informazioni ignaro della loro reale destinazione
  - che un programma esegua richieste fraudolente a scapito di un sistema insicuro, a seguito di violazione dei suoi dati

Classificazione delle minacce

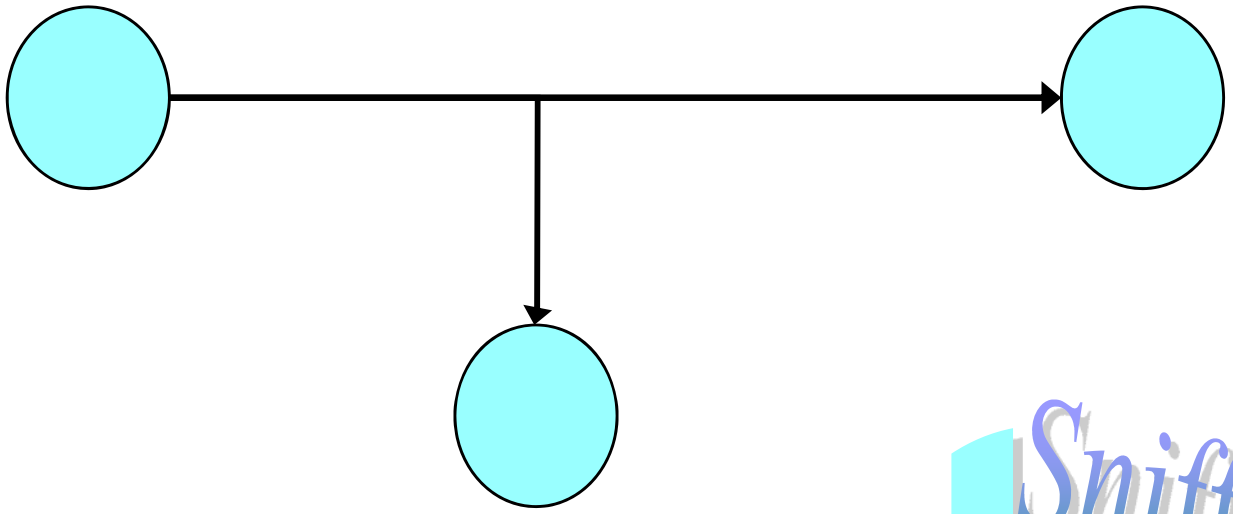
- Lo scopo principale della sicurezza è consentire l'accesso alle risorse ed alle informazioni soltanto ai principals autorizzati
- Le minacce sono contenute in tre classi:
  - Leakeage
    - Accesso ad informazioni del sistema senza autorizzazione
  - Tampering
    - Modifica non-autorizzata delle informazioni
  - Vandalism
    - Interferenza al corretto funzionamento del sistema senza guadagno da parte di chi la attua

Tipologie di attacco al sistema

- Gli attacchi ad un sistema distribuito contano su
  - dall'ottenere o meno l'accesso a canali di comunicazione esistente
  - dal creare nuovi canali di comunicazione che figurano come autorizzate
- Si distinguono nelle seguenti tipologie:
  - Eavesdropping
  - Masquerading
  - Message tampering
  - Replayng
  - Denial of service

Eavesdropping

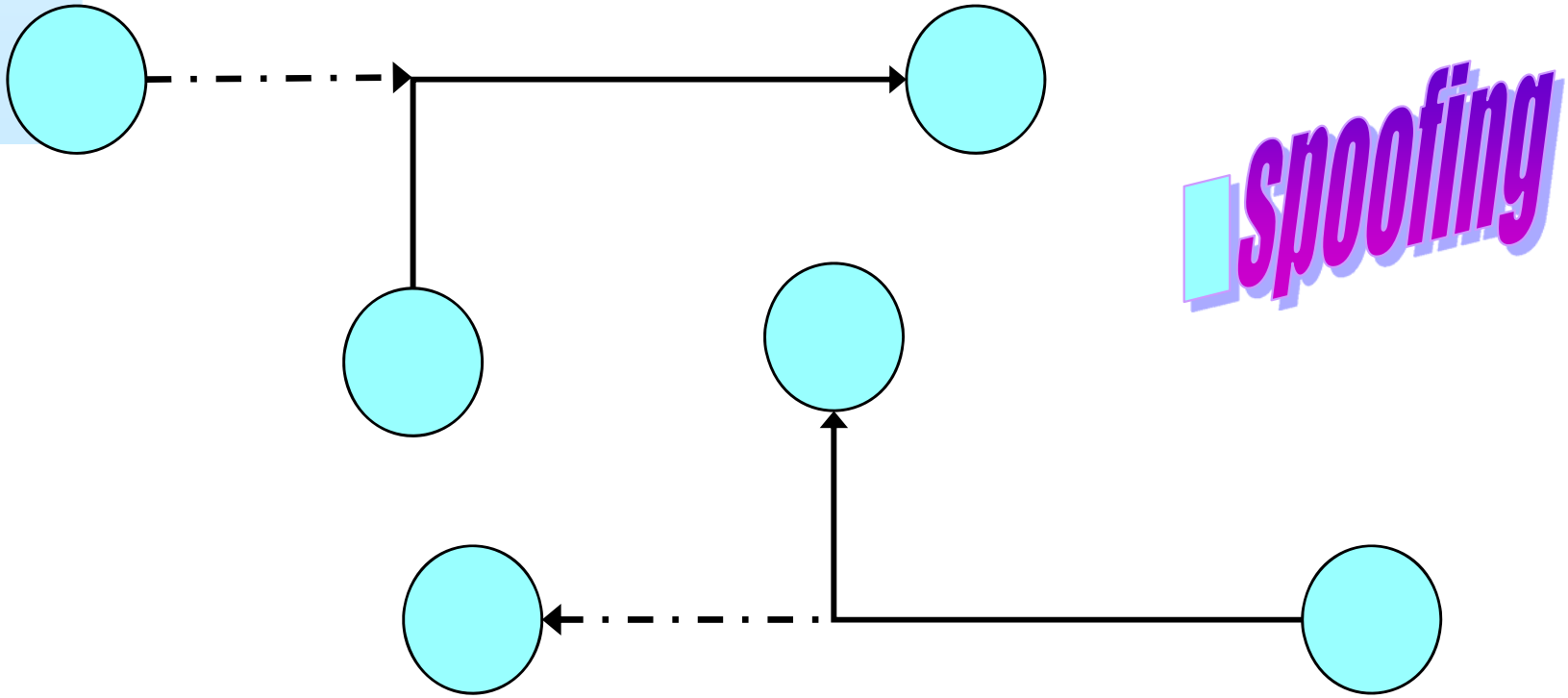
Ottenere copie dei messaggi
senza avere l'autorizzazione



Sniffing

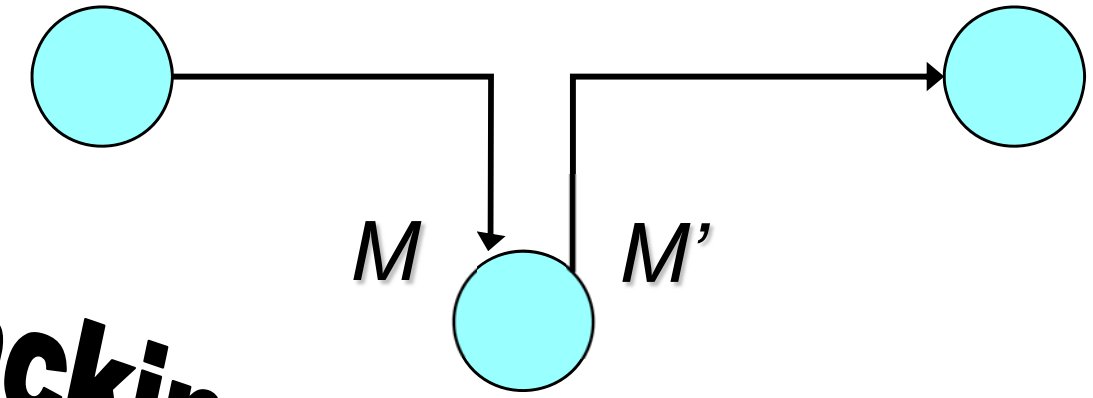
Masquerading

Inviare o ricevere messaggi usando l'identita di altri principals senza la loro autorizzazione



Message tampering

Intercettare messaggi ed alterarne il contenuto prima di ritrasmetterli alla destinazione prevista

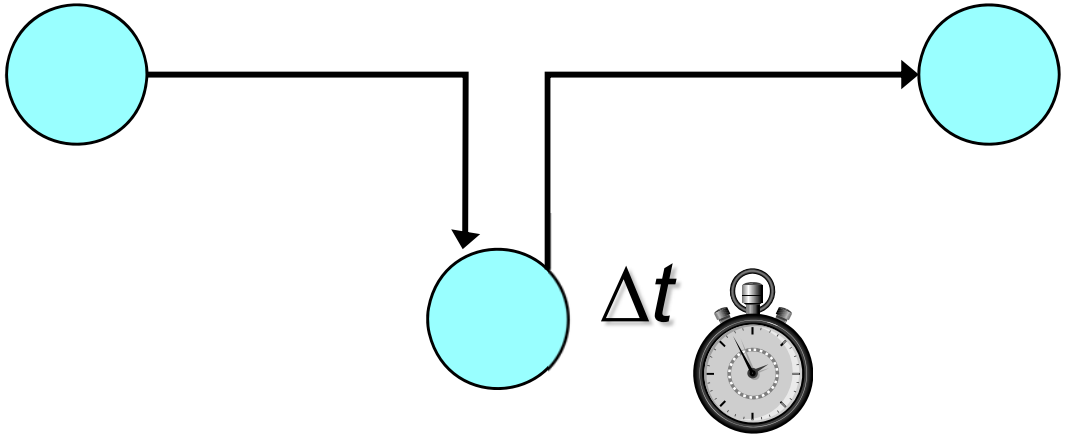


Hijacking

Man-in-the-middle

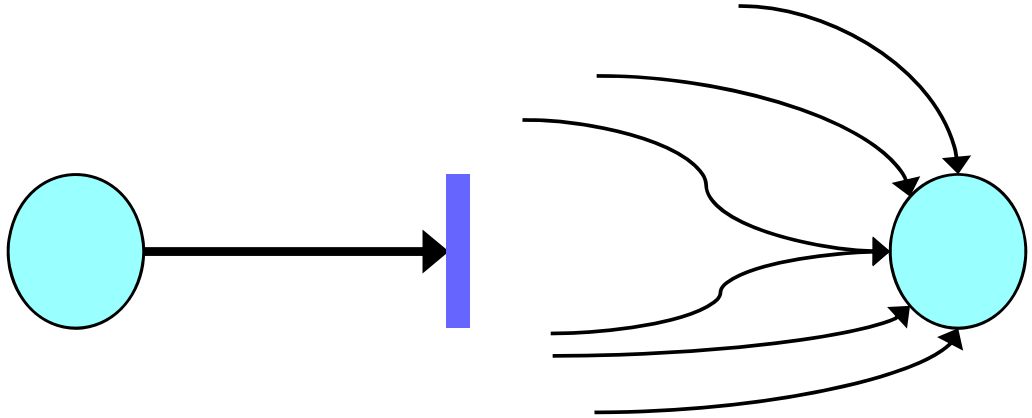
Replaying

Memorizzare messaggi intercettati e inviarli in ritardo rispetto alla loro reale origine



Denial of service

Saturare un canale di comunicazione o altre risorse con messaggi ripetuti in modo da negarne l'accesso da parte degli autorizzati



Pericoli potenziali ed effettivi

- Tutti questi sono pericoli soltanto in teoria, ma gli attacchi che possono andare a buon fine dipendono dal sistema di sicurezza
- Attacchi con successo contano sul fatto di individuare imperfezioni del sistema di sicurezza (*loopholes*)
- Negli attuali sistemi in uso sono comuni e ben evidenti
- sono stati individuati 42 punti deboli che mettono in serio pericolo chi usa largamente sistemi e componenti di Internet
- Quando venivano gettate le basi di Internet la sicurezza non era certamente una priorità

Minacce dal *mobile code*

- Si definisce *mobile code* un programma che viene caricato da un server remoto e viene eseguito in locale
- In questo contesto le risorse all'interno del sistema locale possono subire un attacco dal mobile code
- JVM da ad ogni applicazione mobile code un suo ambiente predefinito ed un security manager determina quali risorse sono disponibili
  - Le classi scaricate in memoria diversa dalle classi locali
  - Il bytecode viene controllato prima di essere eseguito
- Molti browser impediscono alle applet JAVA di accedere ai file locali, alle stampanti o alle socket del sistema

Information leakage

- Il problema della riservatezza delle informazioni non riguarda soltanto il contenuto dei messaggi scambiati

- Anche *osservare* che in un canale sorgente-destinzione il flusso di dati è rilevante può essere un informazione importante

Transazioni elettroniche sicure 1

- E-mail
  - Anche il protocollo dedicato allo scambio di posta non prevedeva originariamente un supporto ala sicurezza, ma la crittografia è adesso comune a molti applicativi
- Aquisto di beni e servizi
  - E-commerce prevede il selezionamento dei beni da acquistare ed il pagamento attraverso il Web

Transazioni elettroniche sicure 2

- Transazioni Bancarie
  - Le banche elettroniche offrono virtualemente i tipici servizi presenti allo sportello di una banca comune (estratto conto, bonifico bancario, domiciliazione utenze)

- Micro-transazioni
  - Altri serivizi posso essere forniti dal Web tipo supporto alla *comunicazione vocale* o alla *videoconferenza*, pagabili a tempo tipicamente con importi bassi da non giustificare la sicurezza prevista per altre transazioni

Requisiti di sicurezza per le transazioni

- Autenticare il venditore al compratore, cosicché egli sia sicuro di essere in contatto con il server del venditore che gli interessa
- Tenere nascoste (ed inalterate) informazioni importanti di pagamento in modo che non cadano in mani sbagliate (i.e. carta di credito)
- Se i beni sono fruibili tramite download assicurare che il contenuto sia consegnato al compratore senza alterazioni e senza accesso da parte di altri

- In aggiunta a questi requisiti può essere necessario autenticare l'identità del client per fornirgli i diritti previsti all'accesso (e-banking)

Progetto di sicurezza

- Progettare un sistema senza punti deboli è simile a scrivere un programma senza *bugs*
- La validazione formale è l'unica possibilità di garantire completa sicurezza
- La prova di validità è articolata in due fasi:
  - Si crea una lista di minacce possibili al sistema
  - Si mostra come ognuna di essi è gestita con successo dal sistema
- La dimostrazione può avere un aspetto informale anche se si predilige un approccio di tipo formale

Progettare un sistema di sicurezza è un esercizio nel bilanciare i costi in relazione alle minacce

Considerazioni worst-case

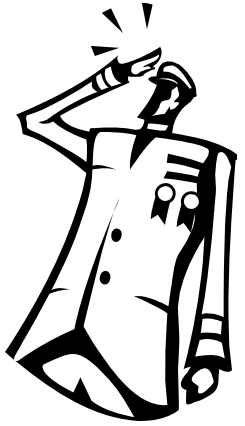
- Le interfacce dei processi server sono necessariamente aperte
- Gli indirizzi degli host possono subire *spoofing*
- Una chiave segreta generata è sicura al momento della sua generazione, ma la sua segretezza diminuisce con il tempo
- Gli algoritmi sono disponibili ai responsabili di sicurezza come agli attackers
- Gli attacckers dispongono spesso di grandi risorse di calcolo
- La base di fiducia (hardware e software) è spesso la causa delle debolezze

Tecnologie per la sicurezza

- Utilizzo della crittografia
- Certificati
- Controllo di accesso
- Credenziali
- Firewall

Cenni storici sulla crittografia

- La crittografia fornisce le basi per la maggior parte dei sistemi di sicurezza informatica
- Ha origine in campo militare a causa del bisogno di comunicazione sicure
- Intercettare e decriptare i messaggi è stato il compito principale di alcuni tra i più autorevoli matematici di quel tempo
- Recentemente la crittografia è uscita dal contesto militare che ne curava l' uso e lo sviluppo
- *Applied Criptography (1996)* è stata una vera propria pietra miliare per chi voleva farsi una cultura nel campo
- Da quel momento nasce una comunità al di fuori dell' ambito militare che produce un grande sviluppo delle tecniche crittografiche

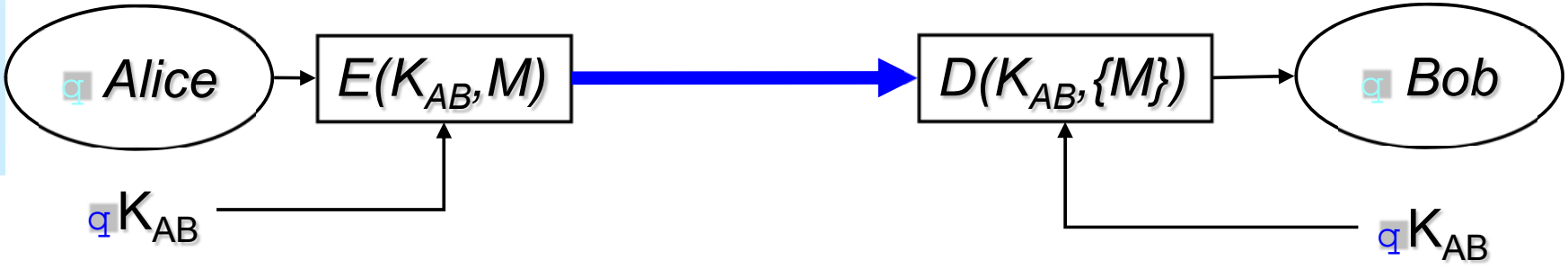


Utilizzo della crittografia

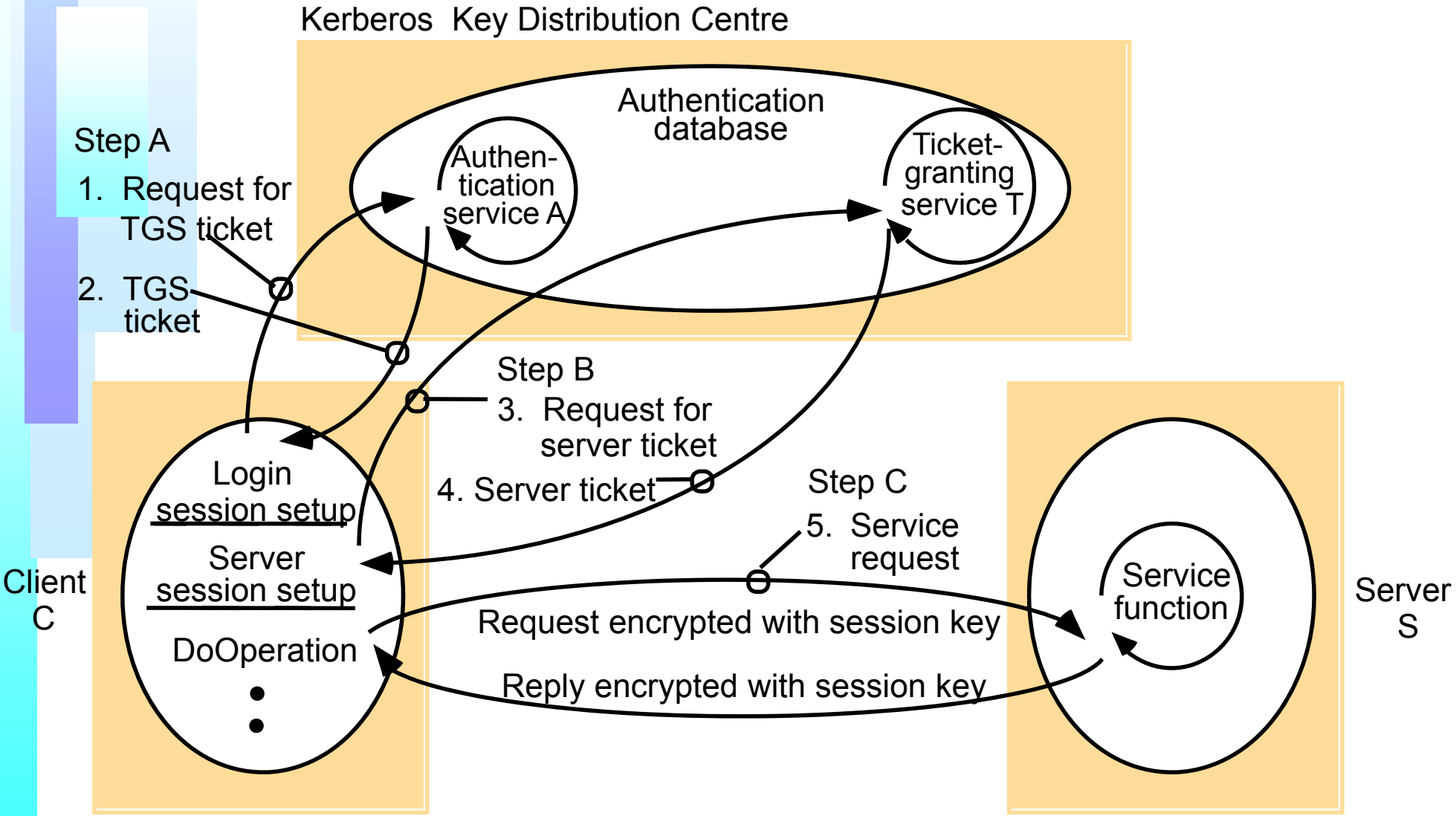
- Encryption è il processo che codifica un messaggio in modo da nasconderne il contenuto
- Si basano sull'uso di parametri segreti chiamati *chiavi*
- Si dividono in due classi fondamentali
  - Chiavi segrete condivise *(secret-key)*
  - Coppie di chiavi pubblica/privata *(public-key)*
- Segretezza e integrità
- Autenticazione
- Firma digitale

Scenario 1: secret communication

- Alice vuole inviare alcune informazioni segretamente a Bob
- Alice e Bob conoscono entrambi la chiave segreta K_{AB}
- La comunicazione è segreta finchè K_{AB} non è compromessa

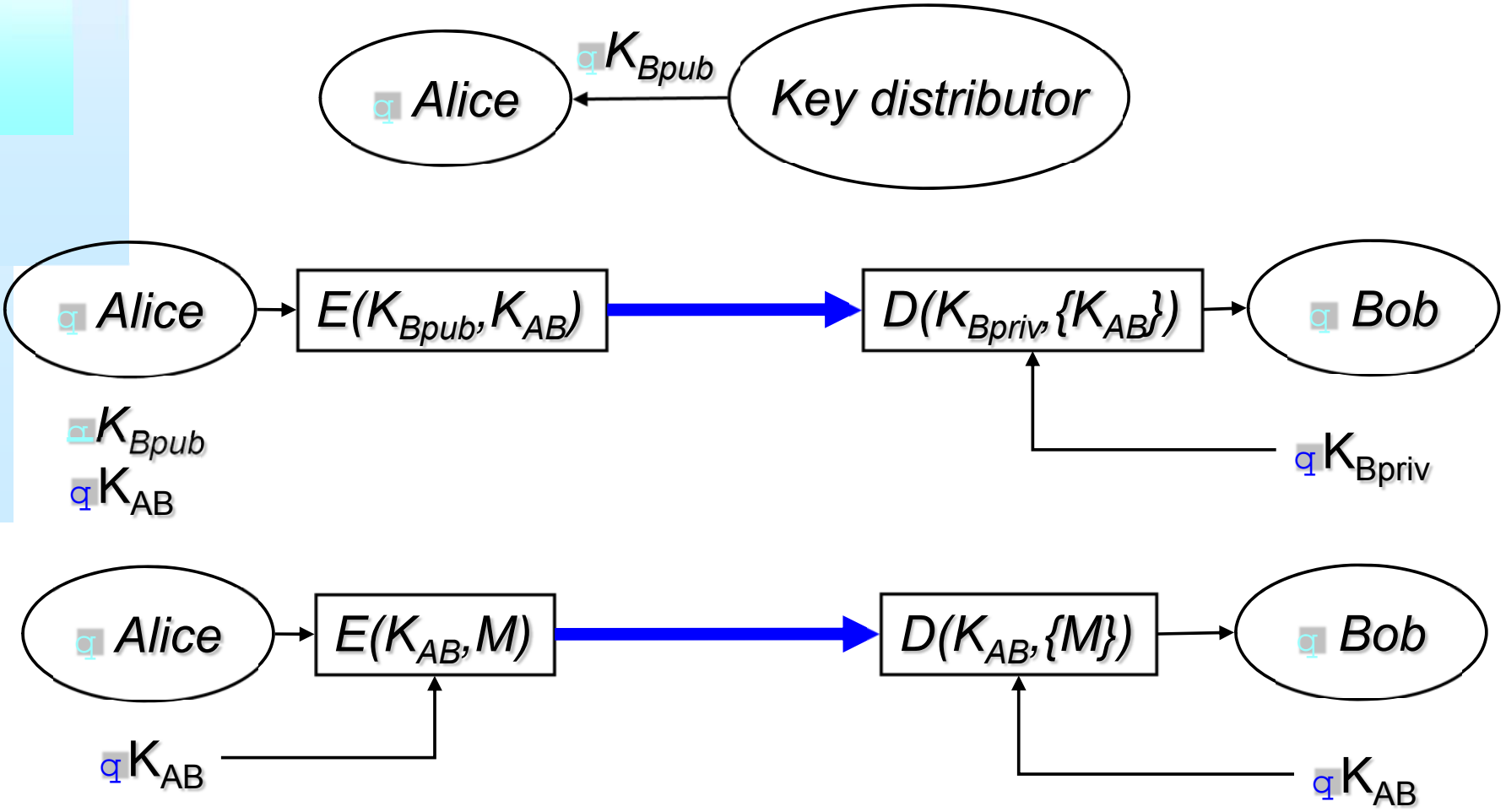


Scenario 2: authenticated via server

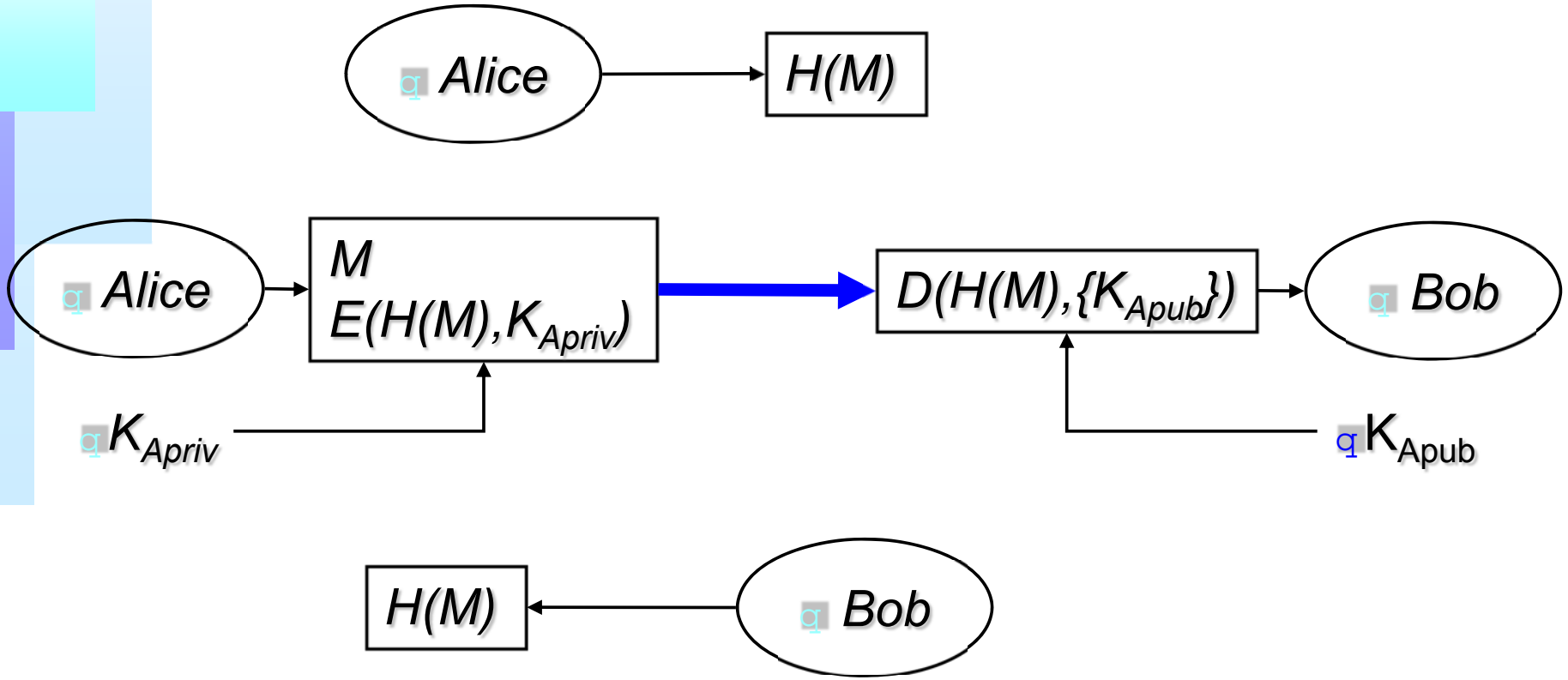


Scenario 3: authenticated with public-key

- Richiesta chiave pubblica
- Determinazione chiave di sessione
- Cumunicazione sicura



Scenario 4: digital signature



Algortimi di crittografia

- Un messaggio si dice criptato quando il mittente applica alcune regole per trasformare il testo originale (*plaintext*) in un altro testo (*ciphertext*)

$$E(K_1, M) = \{M\}_K$$

- Il ricevente deve conoscere la trasformazione inversa per ritrasformare il *ciphertext* nel messaggio originale

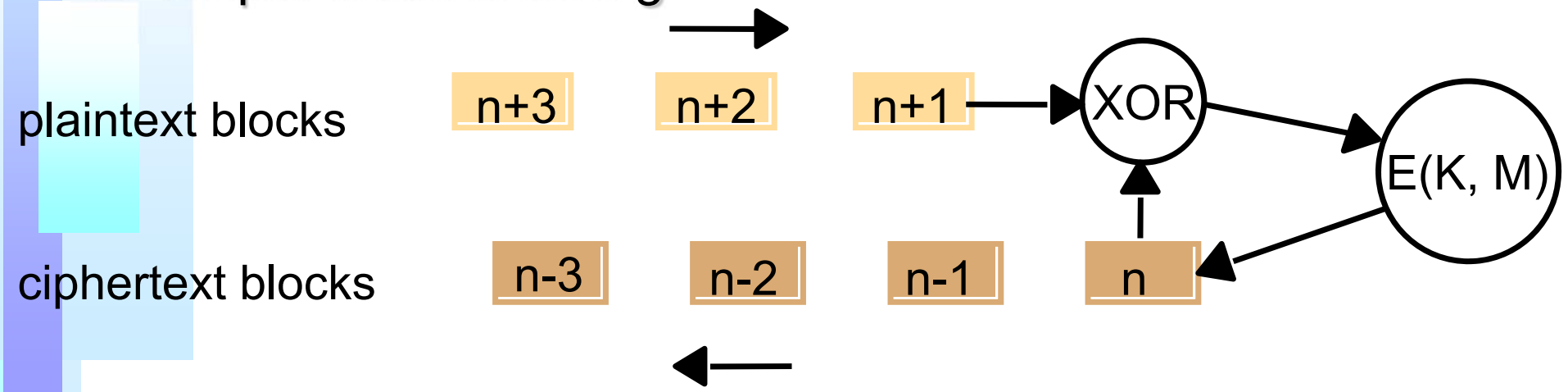
$$D(K_2, E(K_1, M)) = M$$

$K_1 = K_2$ ⇒ **simmetrico**

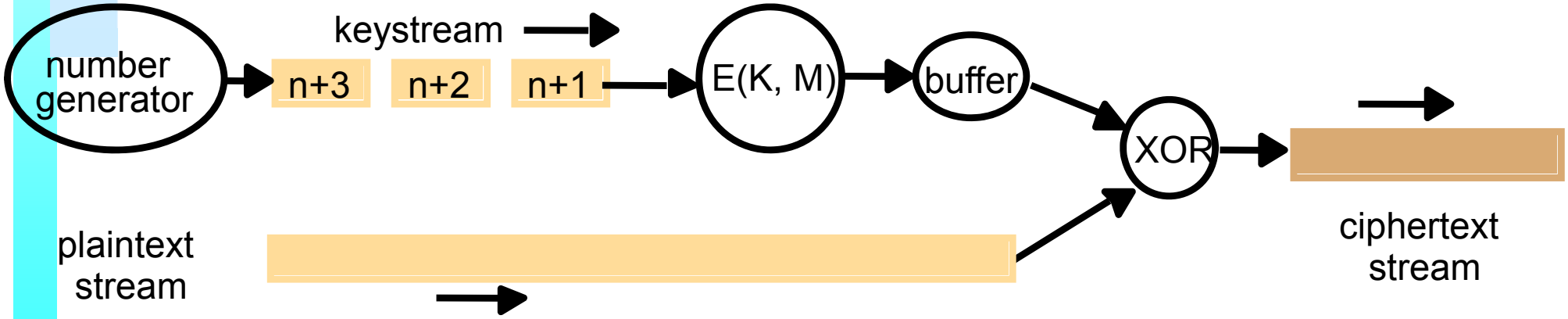
$K_1 != K_2$ ⇒ **asimmetrico**

Algoritmi di critografia 2

- Chiper block chaining



- Stream cipher



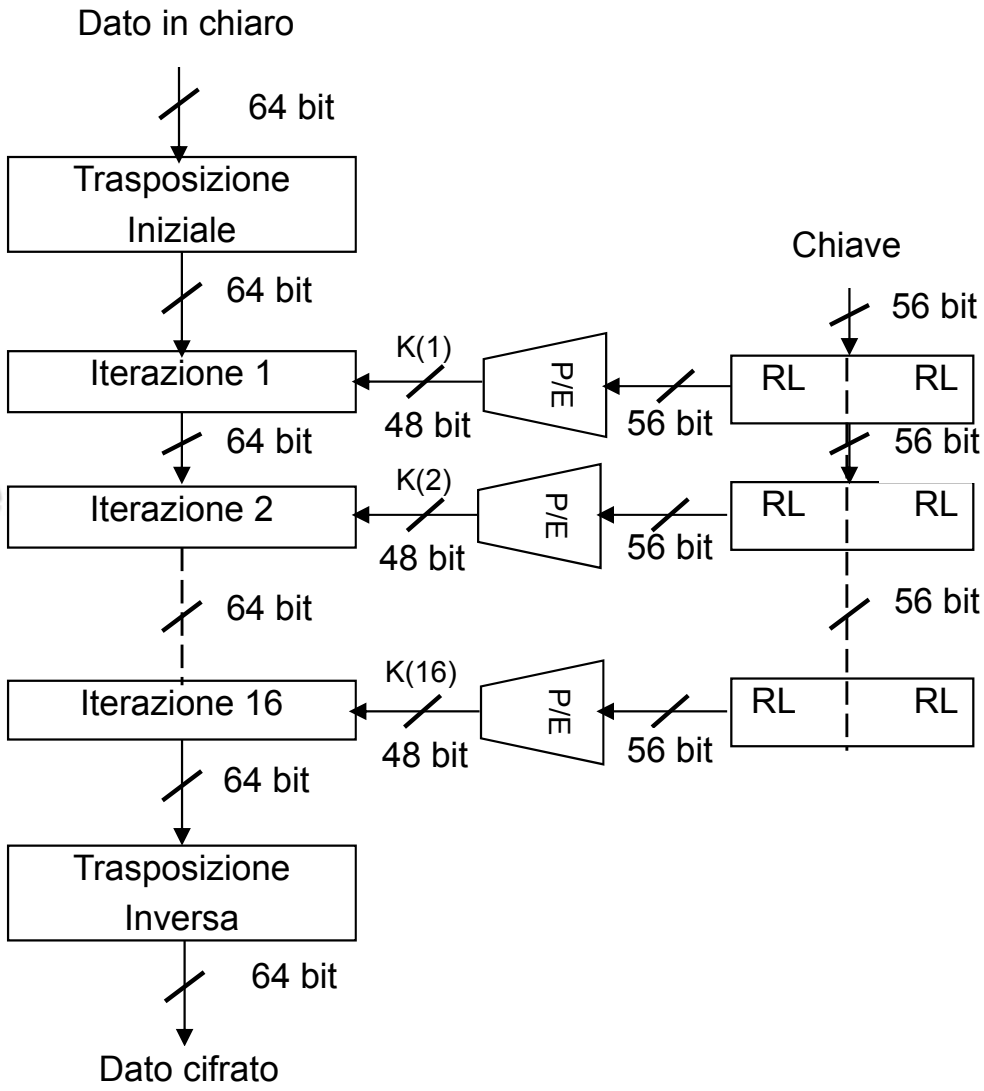
Algorotmi simmetrici

$$F_K([M]) = E(K,M)$$

- Proprietà della funzione di encryption
  - $F_K([M])$ tipicamente facile da calcolare
  - $F_K^{-1}([M])$ "impossibile" da calcolare
- Tali funzioni si definiscono *one-way*
  - *elevamento a potenza*
  - *modulo*
- È tale proprietà che protegge contro gli attacchi che cercano di determinare M dato $\{M\}_K$
- L'unico modo per scardinare algoritmi ben progettati è dato M e $\{M\}_K$ determinare K mediante *brute-force*
- *Combinazioni provate da tale attacco:*
  - *Media* 2^{N-1}; *Massimo* 2^N *dove N = #bit di K*

DES

- Blocchi da 64 bit
- Chiave di 56 bit
- 16 stadi dipendenti dalla chiave detti *rounds*
- Algoritmo apparentemente complesso, ma facilemente codificabile visto che si basa su operazioni di SHIFT e XOR
- Inizialmente implementato hardware ed incorporato nei dispositivi di comunicazione



Algoritmi asimmetrici

- Con coppie di chiavi pubblica/privata le funzioni *one-way* vengono sfruttate in altro modo
- Esistono funzioni dette *trap-door:*
  - one-way con *secret-exit*
  - impossibile calcolare l'inversa senza conoscere un parametro segreto
- Le chiavi K_d, K_e sono derivate da una *radice* comune
- Le due chiavi vengono ottenute con one-way functions
- Chi possiede K_e può criptare M ma soltanto chi possiede K_d ha il segreto per la trap-door (decription)

$$D(K_d, E(K_e, M)) = M$$

RSA

- Sviluppato nel 1977 da **R**ivest, **S**hamir, **A**dleman
- Algoritmo a chiave pubblica più diffuso
- Si basa sul prodotto di due numeri primi sufficentemente grandi (10^{100})
- La funzione prodotto è semplice ad calcolare (anche con numeri grandi)
- Tale traformazione non è invertibile se non si conoscono i fattori e la scomposizione in fattori (grandi numeri) è praticamente impossibile
- La fatorizzazione di un numero dell'ordine 10^{200} richiede più di 4 miliardi di anni con i migliori algoritmi in circolazione (10^6 istr/sec) *1978*
- Attualmente è garantita sicurezza per ~20 anni con chiavi lunghe almeno 768 bits (230 decimali)
- In alcuni casi sono usate chsivi da 1024 o 2048 bit

RSA: find a key pair

To find a key pair *e*, *d*:

1. Choose two large prime numbers, *P* and *Q* (>10^{100}):

   $N = P \times Q \qquad Z = (P-1) \times (Q-1)$

2. For *d* choose any number that is relatively prime with *Z*

   (that is, such that *d* has no common factors with *Z*)

   $P = 13, Q = 17 \Rightarrow N = 221, Z = 192 \qquad d = 5$

3. To find *e* solve the equation: $e \times d = 1 \bmod Z$

   (*e x d* is the smallest element divisible by *d* in *Z*+1, 2*Z*+1,3*Z*+1…)

   $e \times d = 1 \bmod 192 = 1, 193, 385$!!!

   $e = 385/5 \qquad e = 77$

RSA: encryption

To encrypt with RSA method:

- the plaintext is divided into equal blocks of k bits where $2^k < N$

  (that is, such that the numerical value of a block is always less than N;
  in practical applications, k is usually in the range 512 to 1024)

  k = 7, since $2^7 = 128$

2. The function for encrypting a block of plaintext M is:

$$E'(e,N,M) = M^e \bmod N$$

for a message M, the ciphertext is

$$c = M^{77} \bmod 221$$

RSA: decryption

To decrypt a block of encrypted text *C:*

- This function has to be computed:

$$D'(d,N,c) = c^d \bmod N$$

- Rivest, Shamir and Adelman proved that

E' and D' are mutual inverses

(that is, $E'(D'(x)) = D'(E'(x)) = x$)

for all values of x in the range $0 \le x \le N$

- K_e = <e,N> and K_d = <d,N>
- $E(K_e, M) = \{M\}_K$ and $D(K_d, \{M\}_K) = M$

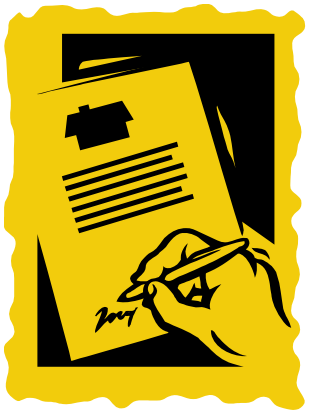
Utilizzo ibrido degli algoritmi

- La crittografia a chiave pubblica è particolarmente idonea alle transazioni commerciali sulla rete perché non necessita di una distribuzione di chiavi segrete
- Il costo computazionale è 100 - 1000 volte quello degli algoritmi a chiave segreta
- La soluzione adottata è nella maggior parte dei sistemi di sicurezza prevede l'uso di entrambi i metodi

1. Autenticazione reciproca mediante chiave pubblica
2. Scambio di una chiave segreta comune (*session-key*)
3. Comunicazione ad alta prestazione usando la chiave di sessione in modo simmetrico

Firma elettronica

- Come la firma “del mondo reale”, la firma elettronica ha valore se posta su un particolare documento
- Una firma “di pugno” certifica che:
  - Il firmatario ha deliberatamente sottoscritto quel documento che non è stato alterato da nessun’altro
  - Il firmatario è sicuramente la persona incaricata a firmare tale documento
  - Il firmatario non potrà in un secondo momento negare di aver sottoscritto il contenuto del documento
- Un segreto che il principal firmatario possiede è considerato come la sua “calligrafia”



Firma elettronica e documenti

- I documenti informatici sono più facili da generare, copiare da alterare
- Non basta legare al documento un'identità in qualunque forma (testo, immagine) forgery

È necessario legare indissolubilmente

l'intero documento

con l'identità del firmatario

- I documenti elettronici sono più resistenti alla falsificazione di quelli cartacei, perdono però il significato del termine *originale*

Firma elettronica e chavi

- Il documento M firmato consiste nella sequenza:

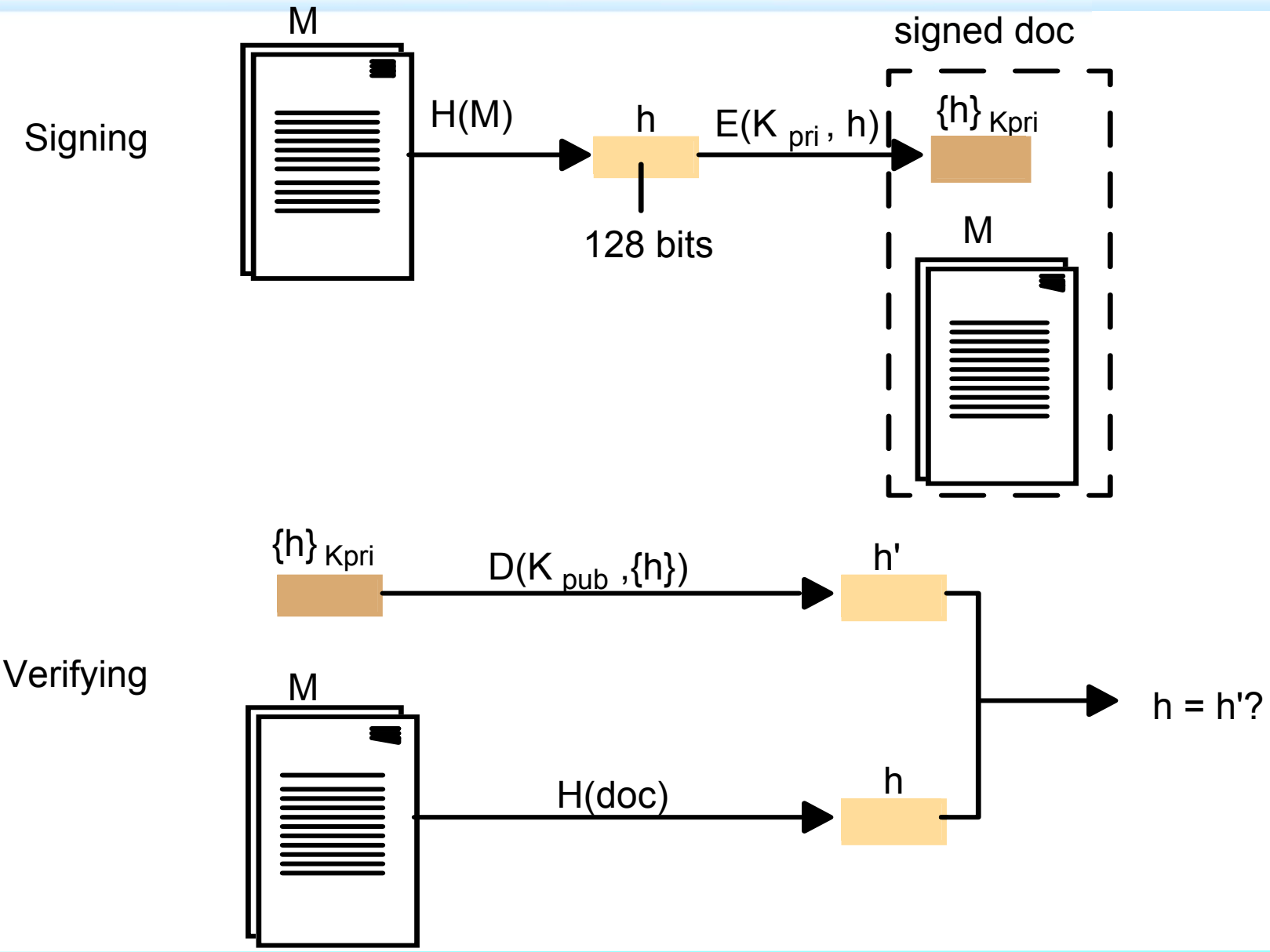
$$M, A, [M]_{K_A}$$

- Se per criptare viene utilizzata una chiave segreta il ricevente può essere sicuro dell'identità del firmatario
- Nella crittografia asimmetrica (preferibile per questo tipo di sicurezza) si utilizza la chiave privata (poiché è segreta) in modo che chi possiede la chiave pubblica possa autenticare il mittente

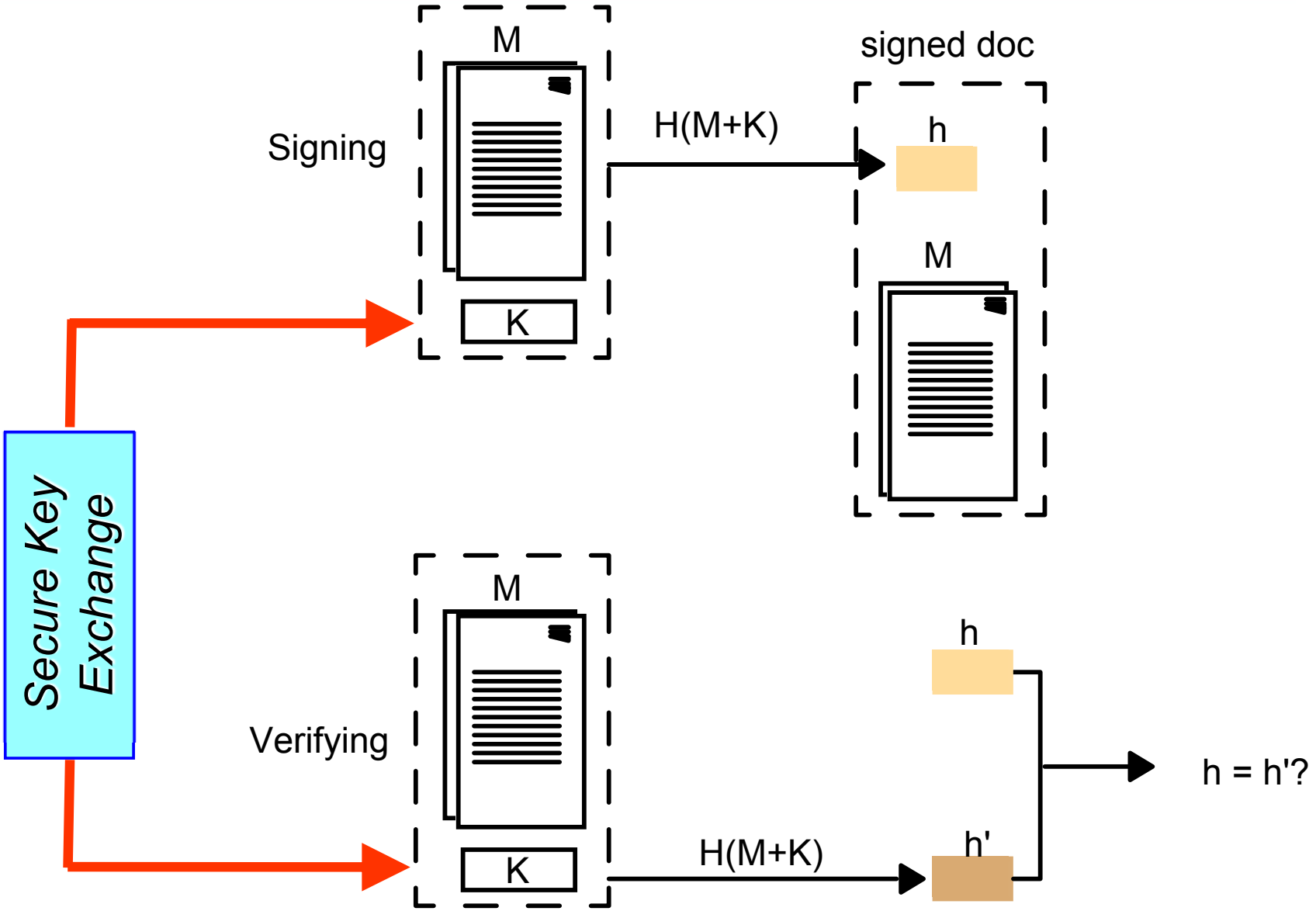
Funzione Digest

- *Digest:* funzione H(M) che produce un'impronta del messaggio e deve possedere le seguenti proprietà:
  - accetta un messaggio di dimensione variabile
  - produce un digest di lunghezza fissa
  - è veloce da calcolare
  - è difficilmente invertibile
  - è estremamente improbabile che messaggi diversi generino lo stesso digest
- Se M e M' diversi danno lo stesso risultato H(M) è possibile che avendo firmato M il ricevente dichiari che è stato firmato M'

Firma elettronica: chiave pubblica



Firma elettronica: chiave segreta (costo minore)



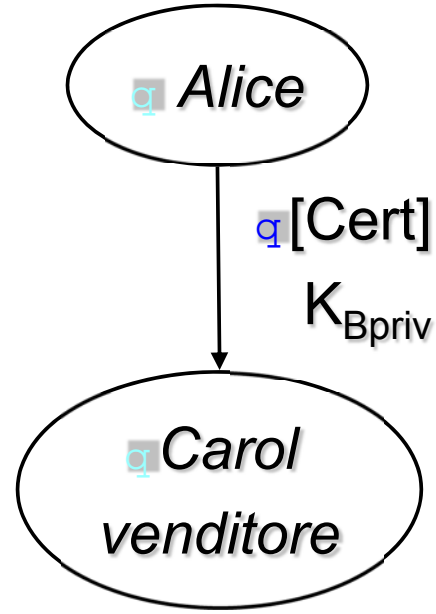
Certificati

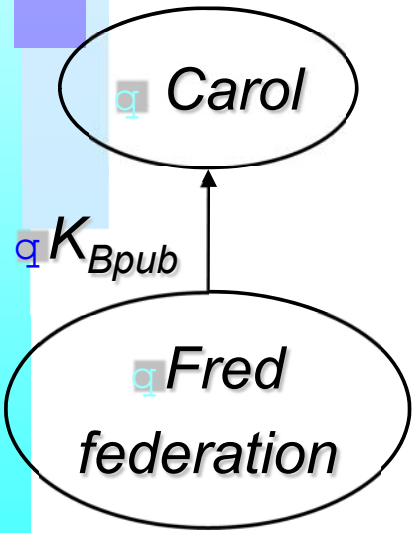
- Documento (breve) che attesta dati di un principal
- Firmati elettronicamente dall'ente emettitore: la Certification Authority (CA)
- Verificati mediante la chiave pubblica della CA
- Scadenza temporale e sono revocabili sia dall'utente che dall'emettitore

1. *Certificate type*	Account number
2. *Name*	Alice
3. *Account*	6262626
4. *Certifying authority*	Bob's Bank
5. *Signature*	$\{Digest(field\ 2 + field\ 3)\}_{K_{Bpriv}}$

Catena di CA

1. *Certificate type*:	Account number
2. *Name*:	Alice
3. *Account*	6262626
4. *Certifying authority*	Bob's Bank
5. *Signature*	{*Digest*()} K_{Bpriv}





1. *Certificate type*:	Public key
2. *Name*:	Bob's Bank
3. *Public key*:	K_{Bpub}
4. *Certifying authority*:	Fred – The Bankers Federation
5. *Signature*:	{*Digest*(*field 2* + *field 3*)} K_{Fpriv}

Certificati X.509

- Formato:

Subject	Distinguished Name, Public Key
Issuer	Distinguished Name, Signature
Period of validity	Not Before Date, Not After Date
Administrative information	Version, Serial Number
Extended Information	.

- La CA fornisce un certificato a seguito di prove inconfutabili di identità (Verisign, CREN)
  - Ottenere il certificato di *issuer public-key* da una fonte affidabile
  - Validare la firma usando tale chiave

- Approccio *Simple Public Key Infrastructure*
  - "Bob crede che la chiave pubblica di Alice è K_{Apub}"
  - "Carol si fida di Bob a riguardo della chiave di Alice"

  => "Carol crede che la chiave pubblica di Alice è K_{Apub}"

Controllo di accesso

- Il controllo di accesso entra in gioco a seguito di un autenticazione
- Al momento in cui è noto chi fa la richiesta è possibile controllare se egli è autorizzato a inoltrare tale richiesta
- Una richiesta è vista come una terna

  <op, principal, resource>
- Esempio di controllo
  - Alice può effettuare 1 prelievi di banconote al giorno
  - Bob ne può effettuare 3
- Viene associato al principal un dominio di protezione
- Questo concetto può essere implementato come
  - Capabilities
  - Access Control List (NT, UNIX…)

Credenziali

- Per credenziali si intende un dato evidente fornito da un principal per accedere ad una risorsa
- Un certificato è l'esempio più semplice di credenziale, ma il concetto può essere generalizzato
- Non è conveniente richiedere all'utente di interagire ogni volta con il sistema per autenticarsi
- La credenziale fornita *parla per* il principal
- Esempi
  - Il certificato di chiave pubblica
  - Ogni processo autenticato con la chiave segreta dell'utente
- Credenziali più avanzate
  - In ambienti cooperativi può essere richiesto un accesso simultaneo di due principal

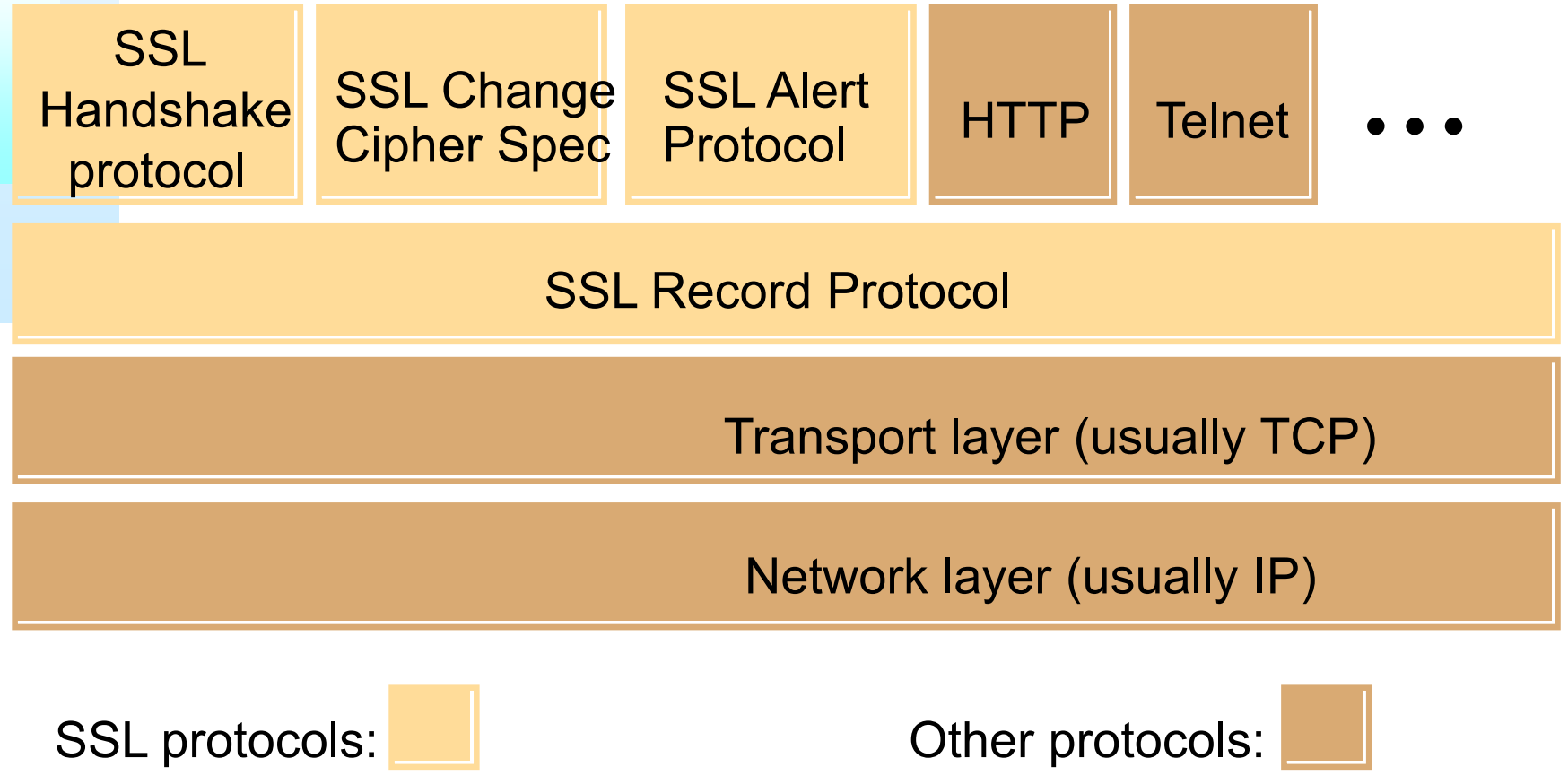
Firewall

- Proteggono le reti Intranet che prevedono una connessione ad Internet
- Operano da filtro al traffico entrante ed uscente dalla rete
- Sono efficaci se associati a protocolli sicuri come Ipsec (VPN), HTTPS che vengono gestiti da un proxy server

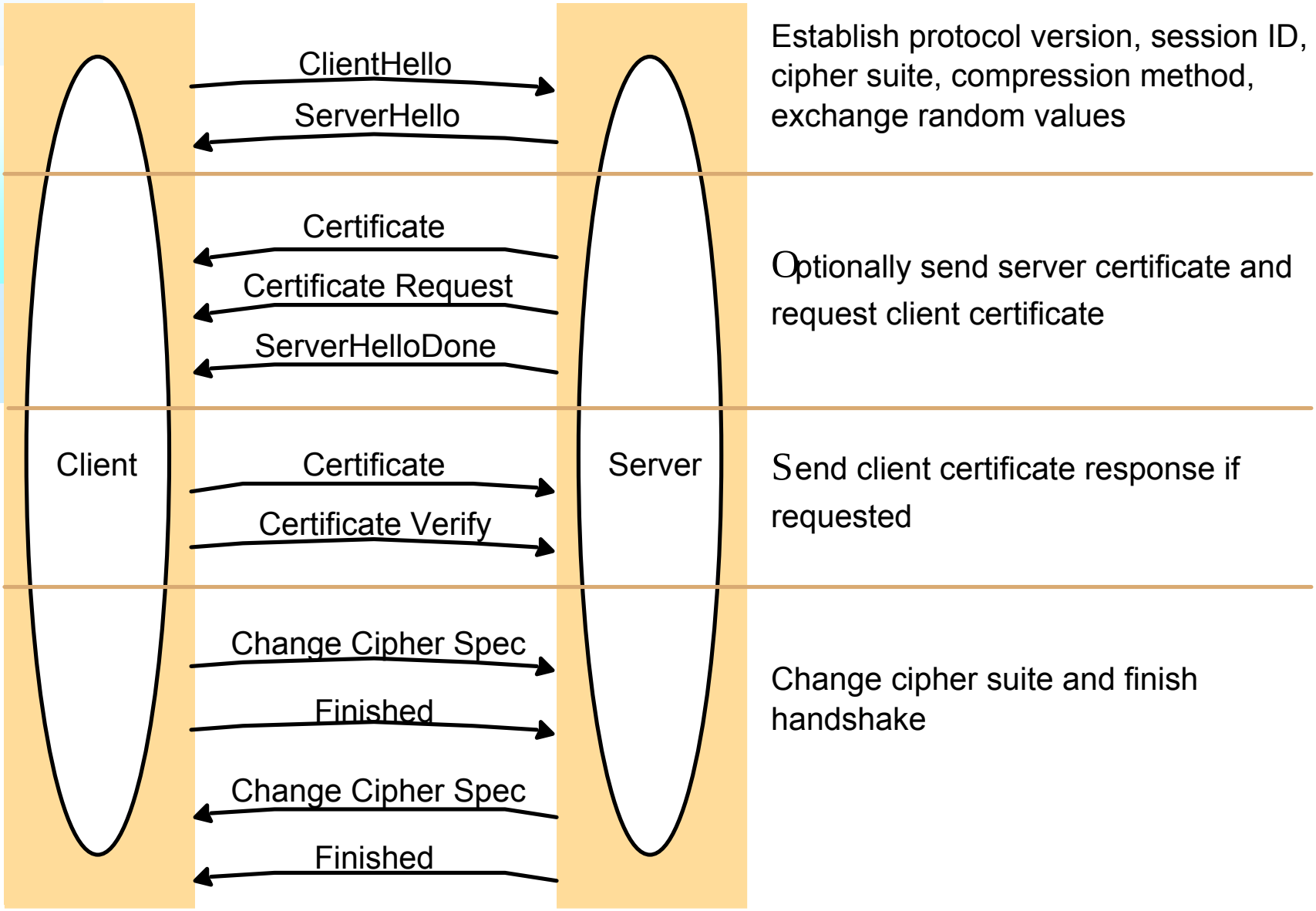
Costi (computazionali) e sicurezza

	Key size/hash size (bits)	*Extrapolated speed (kbytes/sec.)*	*PRB optimized (kbytes/s)*
TEA	128	700	-
DES	56	350	7746
Triple-DES	112	120	2842
IDEA	128	700	4469
RSA	512	7	-
RSA	2048	1	-
MD5	128	1740	62425
SHA	160	750	25162

SSL: Secure Socket Layer



SSL: Secure Socket Layer 2



SSL: negoziazione

Component	*Description*	*Example*
Key exchange method	the method to be used for exchange of a session key	RSA with public-key certificates
Cipher for data transfer	the block or stream cipher to be used for data	IDEA
Message digest function	for creating message authentication codes (MACs)	SHA

SSL: funzionamento

Application data

Fragment/combine

Record protocol units

Compress

Compressed units

Hash

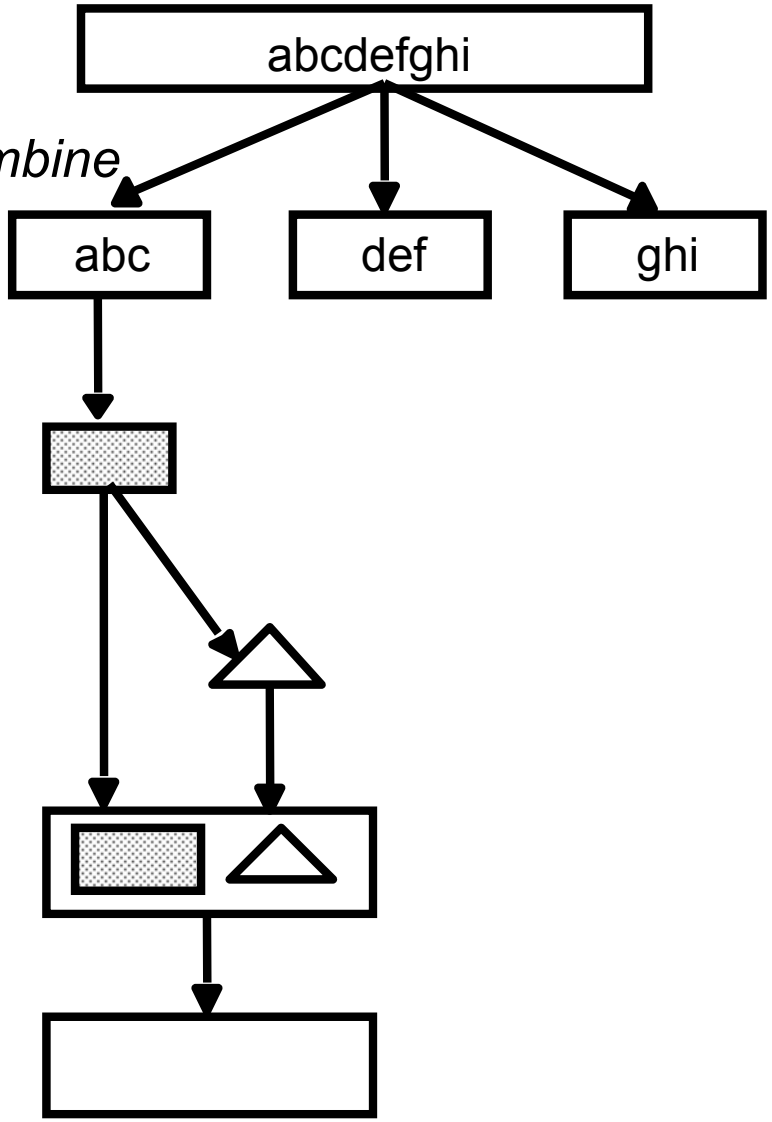
MAC

Encrypt

Encrypted

Transmit

TCP packet



riferimenti

- Si veda il libro del Corso di Sistemi Distribuiti
- Coulouris, Dollimore and Kindberg
  Edition 4, Addison-Wesley 2006