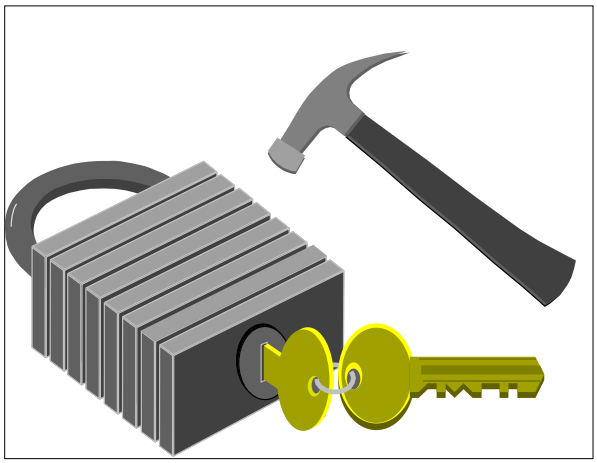
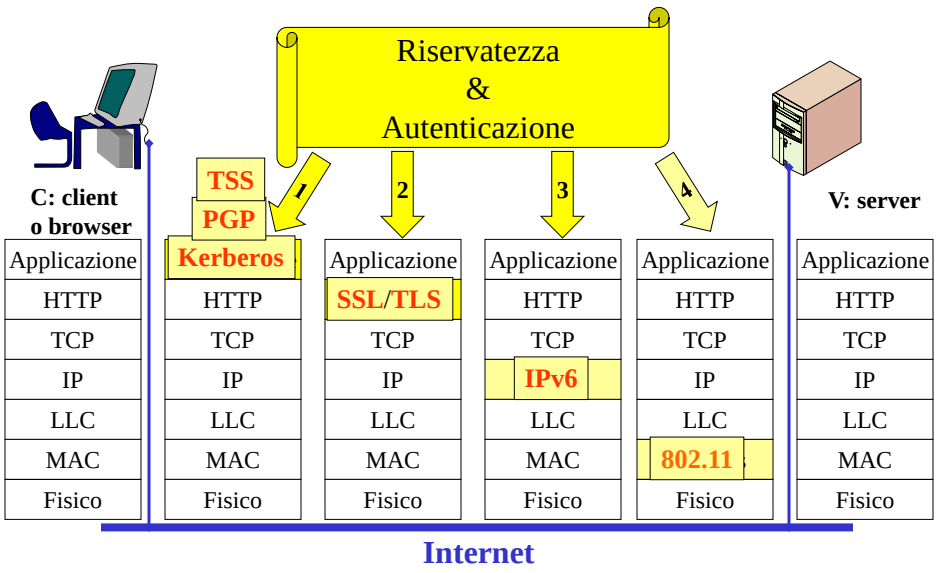
Servizi sicuri



Applicazioni distribuite

Sicurezza a vari livelli

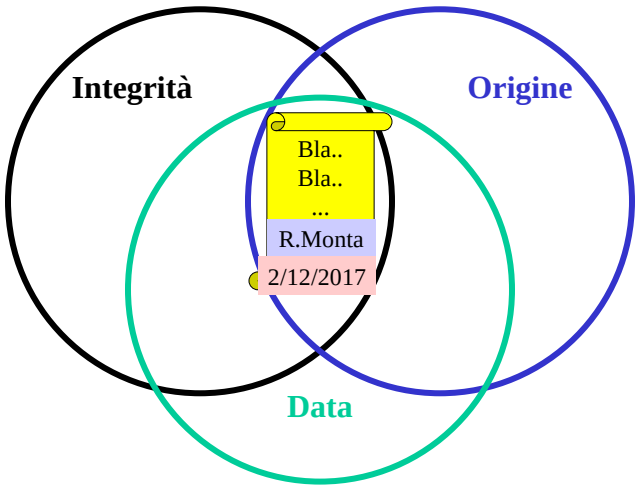


Servizi per le applicazioni

- **TSS**
- **Kerberos**
- **PGP**

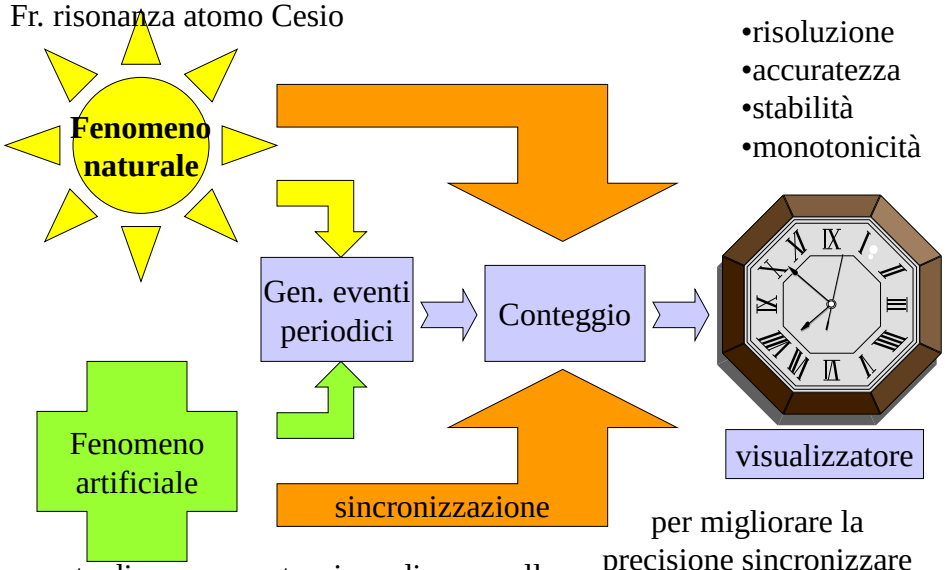
Marche temporali sicure

Il documento informatico sicuro



Misura del tempo

Sole in verticale.
Fr. risonanza atomo Cesio

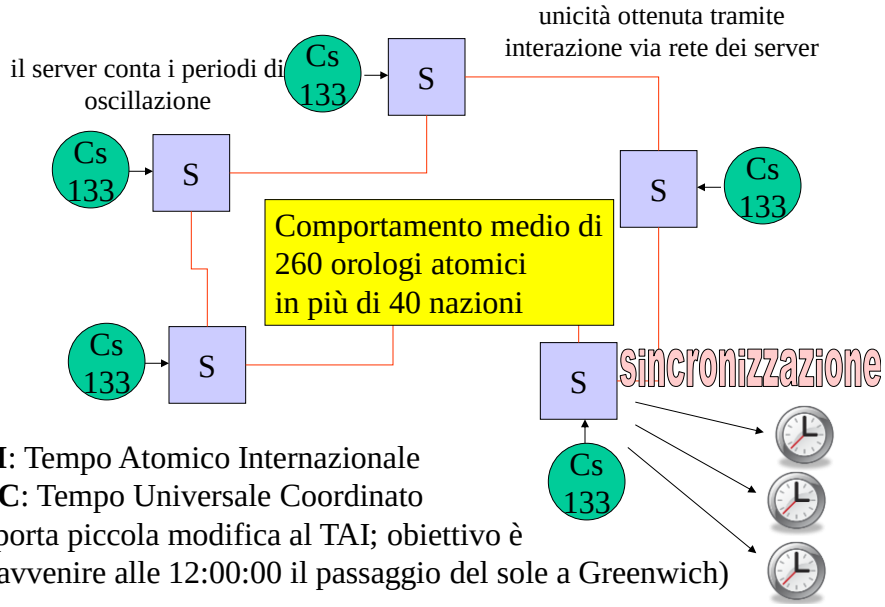


- risoluzione
- accuratezza
- stabilità
- monotonicità

moto di un peso, estensione di una molla, frequenza di un oscillatore elettronico

per migliorare la precisione sincronizzare con un gen. esterno

Riferimenti internazionali: TAI e UTC



TAI: Tempo Atomico Internazionale
UTC: Tempo Universale Coordinato
(apporta piccola modifica al TAI; obiettivo è far avvenire alle 12:00:00 il passaggio del sole a Greenwich)

Marche temporali

•Messaggistica elettronica	nessun controllo
•File system, Database	monotonicità (per ordinamento eventi)
•Applicazioni distribuite	sincronismo (per coordinamento)
•Certificato di chiave pubblica	anno/mese/giorno di inizio/fine
•Revoca e CRL	data/ora con grande precisione

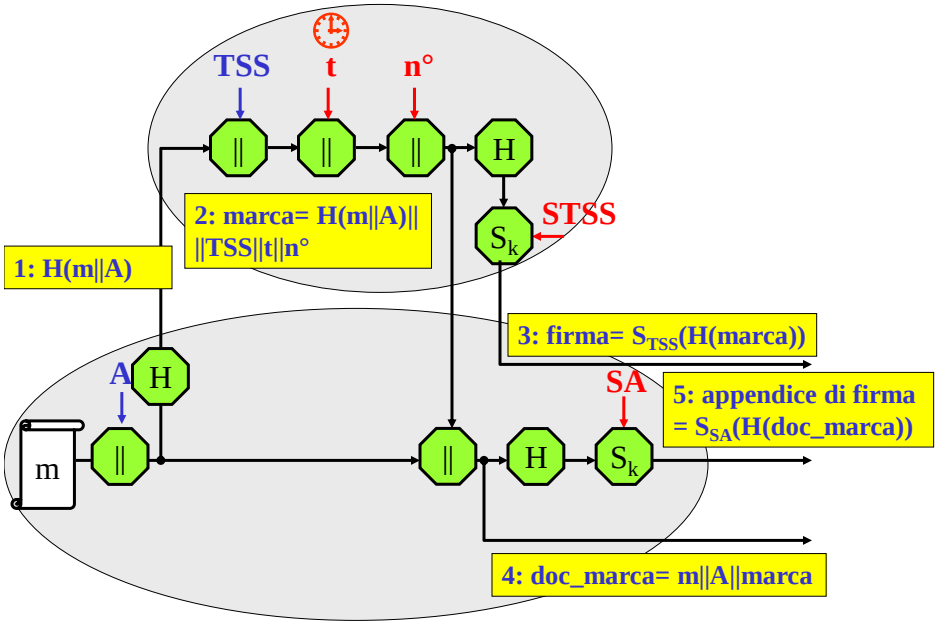
Firma digitale & marca temporale
- collocazione temporale del documento
- validità della chiave di firma

TSS: Time Stamp Service

Marcatura temporale di un documento

- Il tempo della marca non deve essere falso — **Ente fidato**
- Tramite la marca deve essere possibile individuare in modo sicuro un documento, un istante ed un autore — **Firma digitale**
- La modifica anche di un solo bit di una marca deve poter essere rilevata
- Deve essere possibile farsi marcare documenti mantenendone riservato il contenuto — **Hash sicura**
- Chiunque deve poter sia farsi marcare i suoi documenti, sia verificare la marcatura dei documenti di chiunque altro — **Servizio pubblico**; **Chiave di verifica sicura**

Marcatura e firma di un documento



Problemi

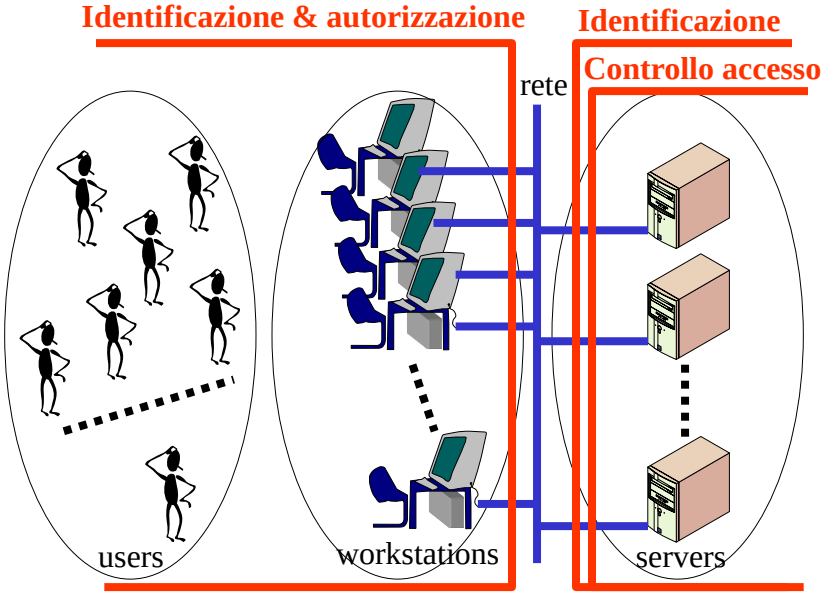
- PKIX: PKI&TSS
- efficienza
- vita della chiave/vita dei documenti
- fidatezza

- Nuovi standard?
- notarizzazione
- tempo relativo

KERBEROS

In Greek mythology, a many headed dog, the guardian of the entrance of Hades

Single sign-on: una sola psw per accedere a molti servizi



KERBEROS

- Provides a centralized authentication server to authenticate users to servers and servers to users.
- Relies on conventional encryption, making no use of public-key encryption
- Two versions: version 4 and 5 (RFC 1510)

Three threats exist:

–User pretend to be another user.

–User alter the network address of a workstation.

–User eavesdrop on exchanges and use a replay attack.

Semplice dialogo di autenticazione

1. C -> AS: $ID_c||P_c||ID_v$
2. AS ->C: Ticket
3. C ->V: $ID_c||$ Ticket

Ticket = $E_{kv}[ID_c,AD_c,||ID_V]$

AD_c per evitare intercettazione del ticket al passo 2 e riutilizzo al passo 3 (al passo 3 l'indirizzo del client deve essere uguale a quello originario)

Problemi:

Trasmissione della password

Ad ogni connessione a un server (anche lo stesso) l'utente deve rifare il protocollo

Dialogo di autenticazione più sicuro

Per ogni sessione di login:

- C -> AS: $ID_c||ID_{tgs}$
- AS ->C: $E_{kc}[Ticket_{tgs}]$

Si evita invio della pwd!
E_{kc} deriva dalla pwd

$$Ticket_{tgs} = E_{ktgs}[ID_c||AD_c,||ID_{tgs}\ ||TS_1||Lifetime_1]$$

La cifratura impedisce la costruzione di informazioni falsificate

Perchè TS_1 e $Lifetime_1$? Non si vuole che Eve catturi e riutilizzi il ticket (ad es. aspetta che il logout dell'utente, falsifica indirizzo di rete; timestamp e lifetime limitano possibilità di riutilizzo)

Dialogo di autenticazione più sicuro

Per ogni sessione di login:

- C -> AS: $ID_c||ID_{tgs}$
- AS ->C: $E_{kc}[Ticket_{tgs}]$

Si evita invio della pwd!
E_{kc} deriva dalla pwd

$$Ticket_{tgs} = E_{ktgs}[ID_c||AD_c,||ID_{tgs}\ ||TS_1||Lifetime_1]$$

La cifratura impedisce la costruzione di informazioni falsificate

Dialogo di autenticazione più sicuro

Per ogni tipo di servizio:
1. C ->TGS: IDc|| ID_V ||$Ticket_{tgs}$
2. TGS ->C: $Ticket_v$

$Ticket_v = E_{kv}[ID_c||AD_c,||ID_v ||TS_2||Lifetime_2]$

Per ogni sessione di servizio:
C ->V: ID_c|| $Ticket_v$

Dialogo di autenticazione più sicuro

Per ogni tipo di servizio:
1. C ->TGS: IDc|| ID_V ||$Ticket_{tgs}$
2. TGS ->C: $Ticket_v$

$Ticket_v = E_{kv}[ID_c||AD_c,||ID_v ||TS_2||Lifetime_2]$

Per ogni sessione di servizio:
C ->V: ID_c|| $Ticket_v$

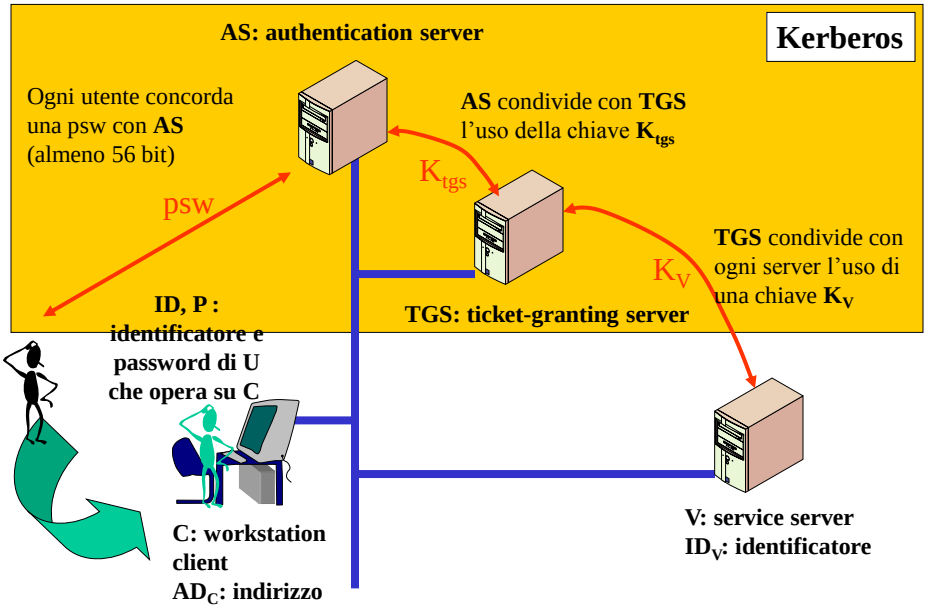
Problemi:

•Durata del ticket (se troppo breve, l'utente deve re-inserire la pwd, se troppo lungo problema di intercettazione e riutilizzo)

•E se servisse anche l'autenticazione da parte dei server?

•Occorre che il TGS dimostri che la persona che utilizza il ticket è quella persona per cui è stato emesso

AS, TGS, V, C



Il dialogo tra C, AS, TGS e V

1: All'inizio della sessione di lavoro sulla stazione **C**, l'utente dichiara la sua identità ad **AS**

2: **AS** fornisce il permesso d'accesso a **TGS** **e lo sfida ad usarlo** — ticket TGS, ΔT

3: **C** risponde alla sfida, richiedendo anche l'accesso al server **V**

4: **TGS** fornisce a **C** il permesso d'accesso a **V** **e lo sfida ad usarlo** — ticket V, ΔT

5: **C** si qualifica a **V**

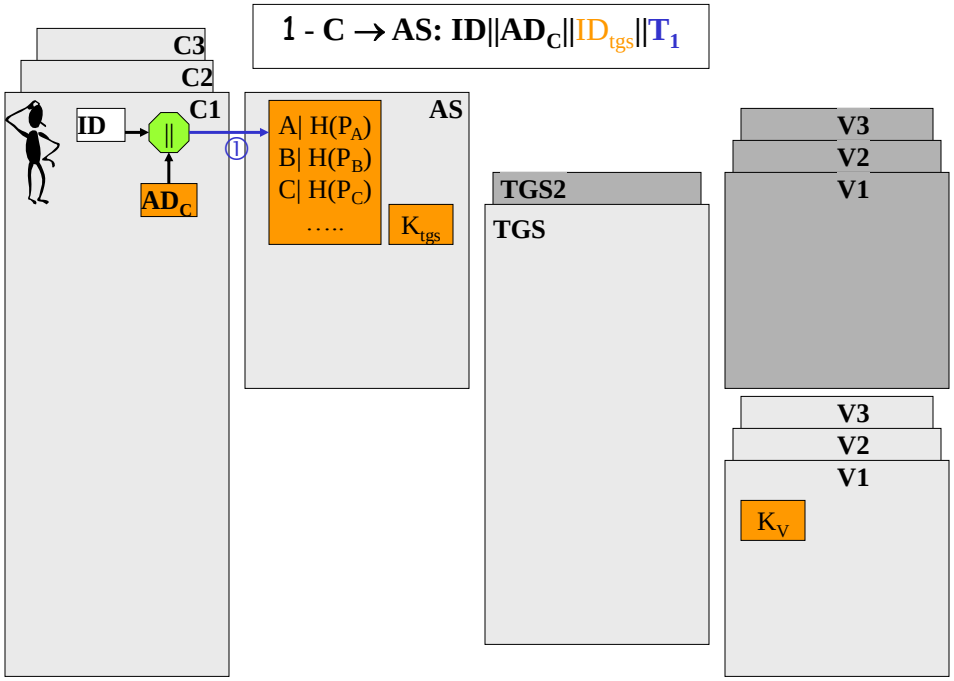
6: **V** si fa identificare da **C**

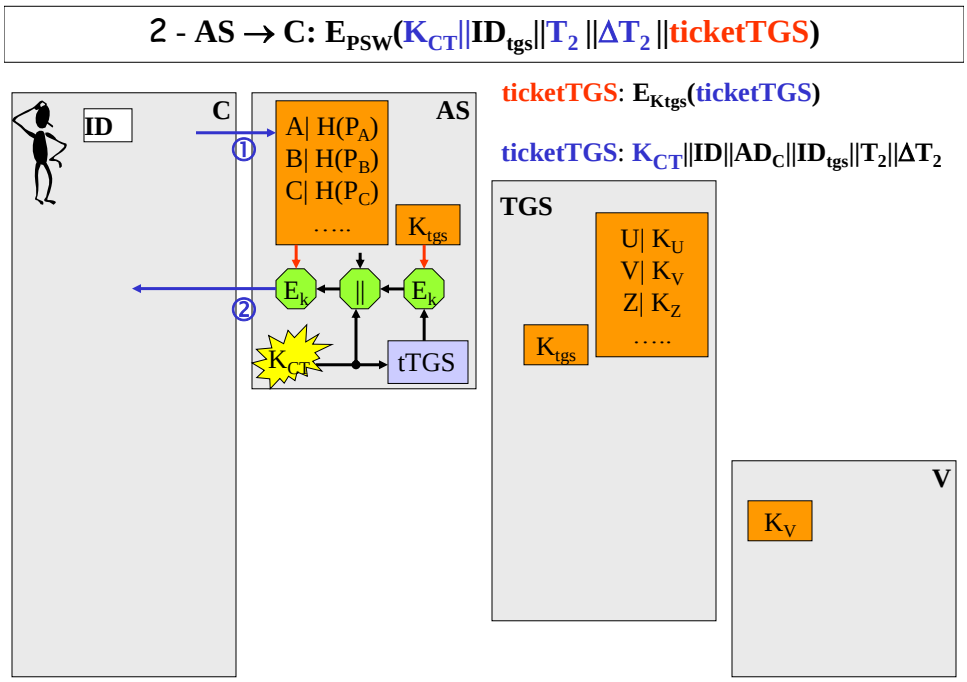
Svolgimento del servizio

Successiva esigenza di servizio da V: **goto 5**

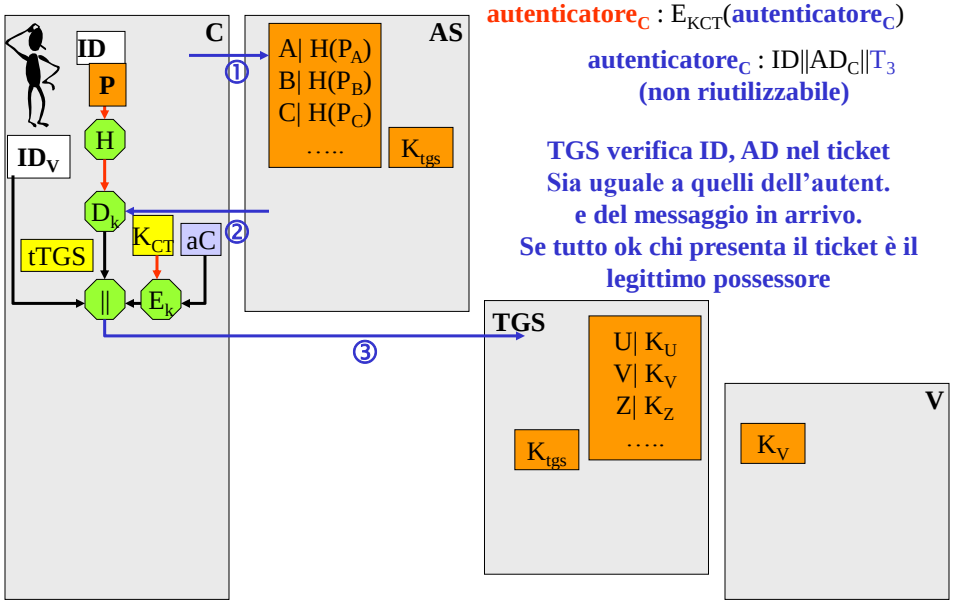
Accesso ad un nuovo server: **goto 3**

N: Fine della sessione o tempo scaduto


1 - C → AS: ID||AD_C||ID_tgs||T_1
C3
C2
C1
ID
||
AD_C
①
AS
A| H(P_A)
B| H(P_B)
C| H(P_C)
…..
K_tgs
TGS2
TGS
V3
V2
V1
V3
V2
V1
K_V


2 - AS → C: E_PSW(K_CT||ID_tgs||T_2 ||ΔT_2 ||ticketTGS)
ticketTGS: E_Ktgs(ticketTGS)
ticketTGS: K_CT||ID||AD_C||ID_tgs||T_2||ΔT_2
ID
C
①
AS
A| H(P_A)
B| H(P_B)
C| H(P_C)
…..
K_tgs
E_k
||
E_k
②
K_CT
tTGS
TGS
U| K_U
V| K_V
Z| K_Z
…..
K_tgs
V
K_V

3 - C → TGS: ID_V||ticketTGS||$autenticatore_C$

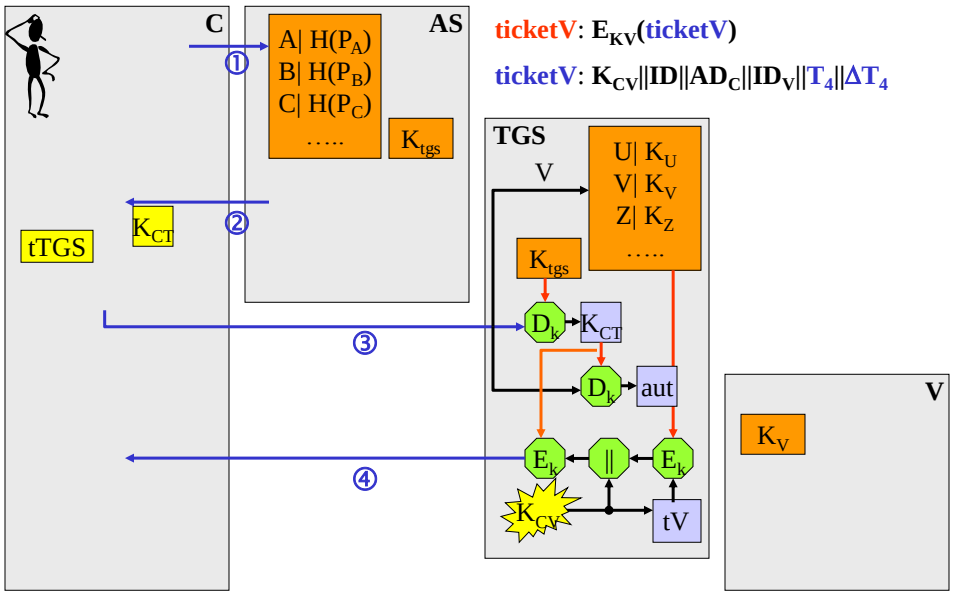


$autenticatore_C$: E_{KCT}(**$autenticatore_C$**)

$autenticatore_C$: $ID||AD_C||T_3$
(non riutilizzabile)

TGS verifica ID, AD nel ticket
Sia uguale a quelli dell'autent.
e del messaggio in arrivo.
Se tutto ok chi presenta il ticket è il legittimo possessore

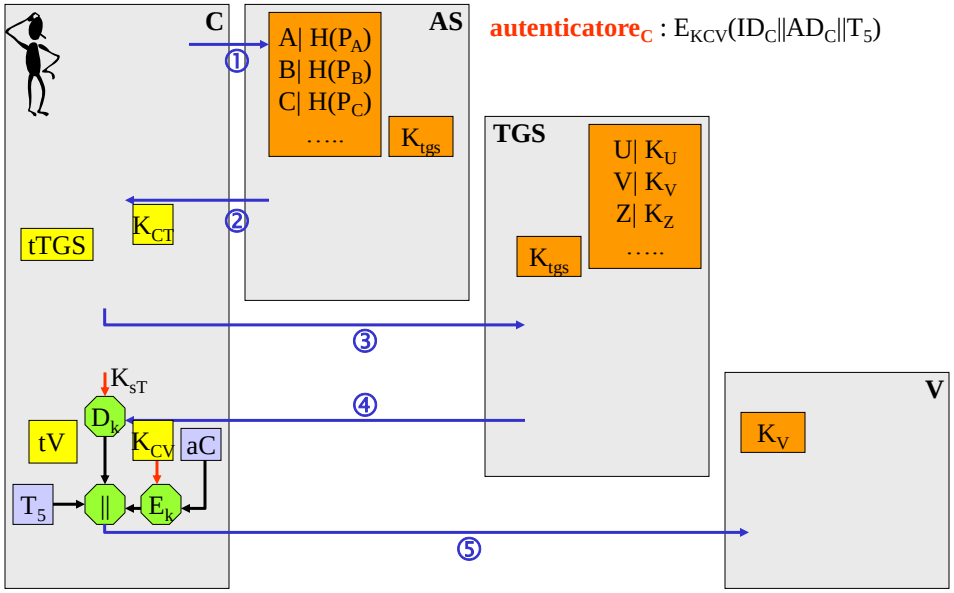
4 - TGS → C : $E_{KCT}(K_{CV}||ID_V||T_4||$ ticketV)



ticketV: E_{KV}(ticketV)

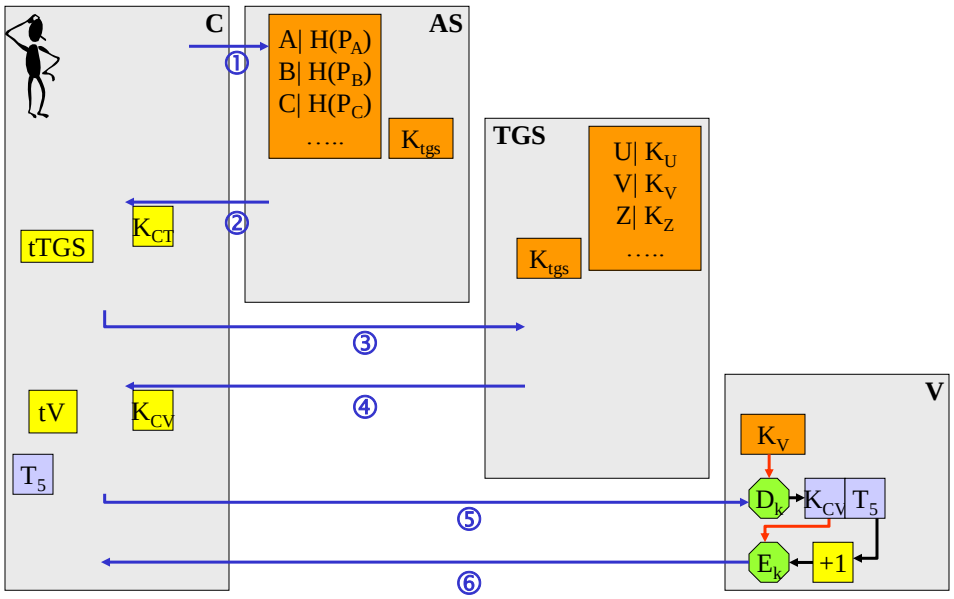
ticketV: $K_{CV}||ID||AD_C||ID_V||T_4||\Delta T_4$

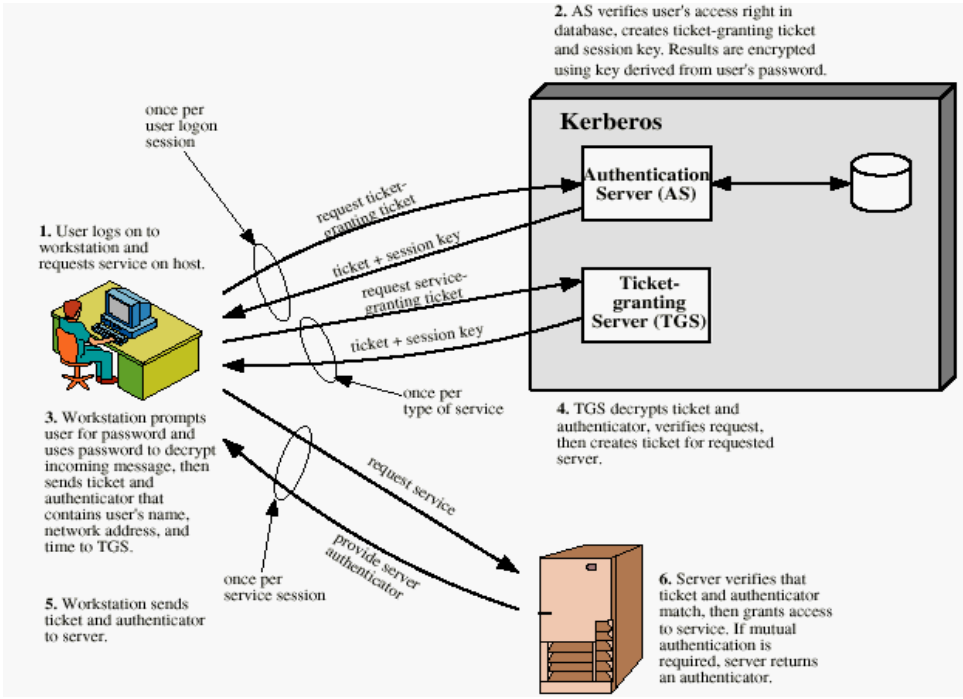
5 - C → V: ticketV||autenticatore$_C$



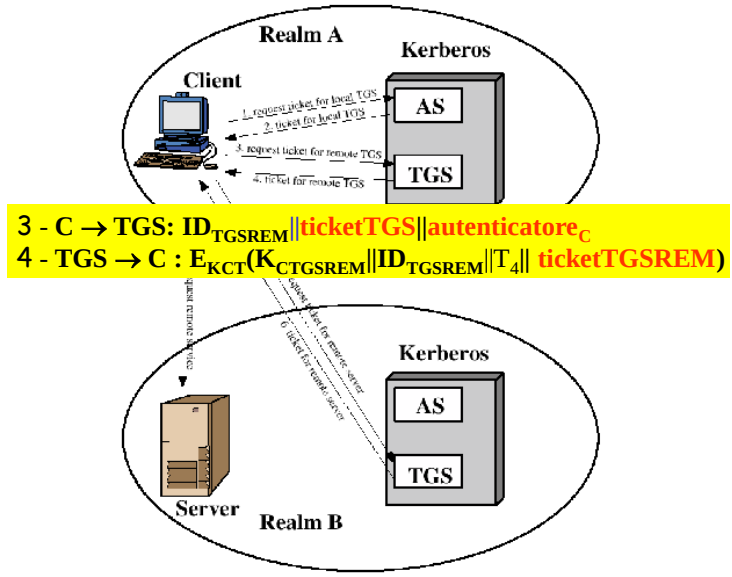
autenticatore$_C$: $E_{KCV}(ID_C||AD_C||T_5)$

6 - V → C : $E_{KCV}(T_5+1)$





Request for Service in Another Realm



3 - C → TGS: ID_{TGSREM}||ticketTGS||$autenticatore_C$
4 - TGS → C : $E_{KCT}(K_{CTGSREM}||ID_{TGSREM}||T_4||$ ticketTGSREM)

Vuoi imparare a progettare un servizio sicuro?

..leggere

Bryant W. "*Designing an Authentication System: A Dialogue in Four Scenes*"

http://www.cert.org/annual_rpts/cert_rpt_98.html

Kerberos v4

1. Dipendenza dal DES
2. Dipendenza dal protocollo IP per gli indirizzi
3. Durata del ticket (max 8 bit con unità di 5 minuti => 2^8x5 -> 1280 minuti
4. Autenticazione tra realm differenti (N^2)

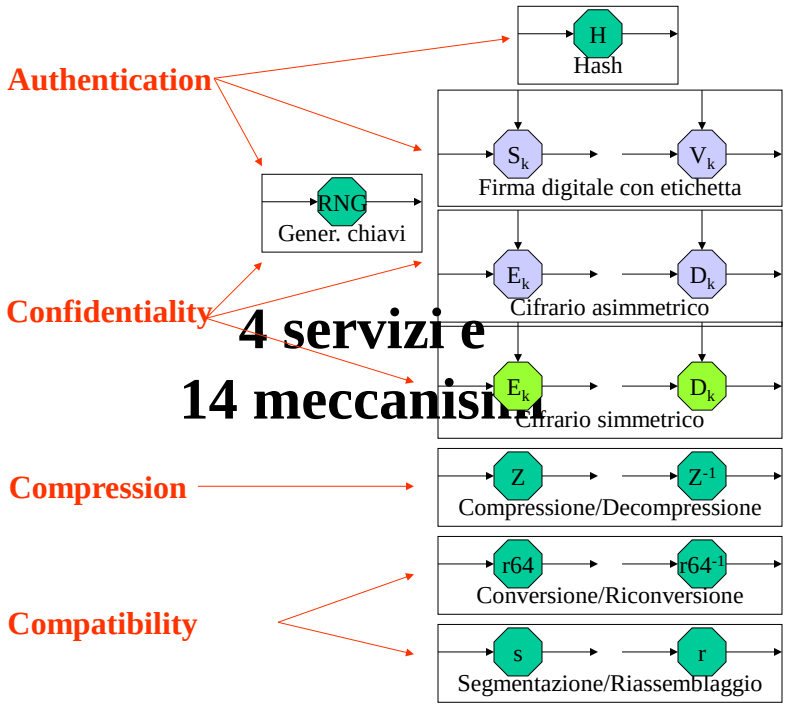
Pretty Good Privacy

- Philip R. Zimmerman is the creator of PGP (1993).

PGP provides
a **confidentiality** and **authentication** service
that can be used for
electronic mail and **file storage** applications.

Confidentiality: the act of keeping something private and secret from all but those who are authorized to see it.

Authentication: to prove genuine by corroboration of the identity of an entity.

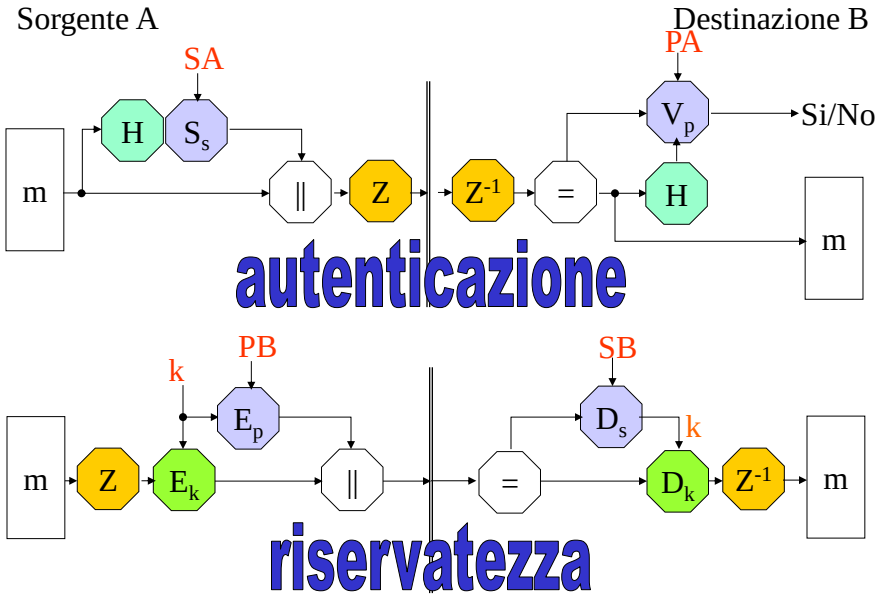


4 servizi e
14 meccanismi

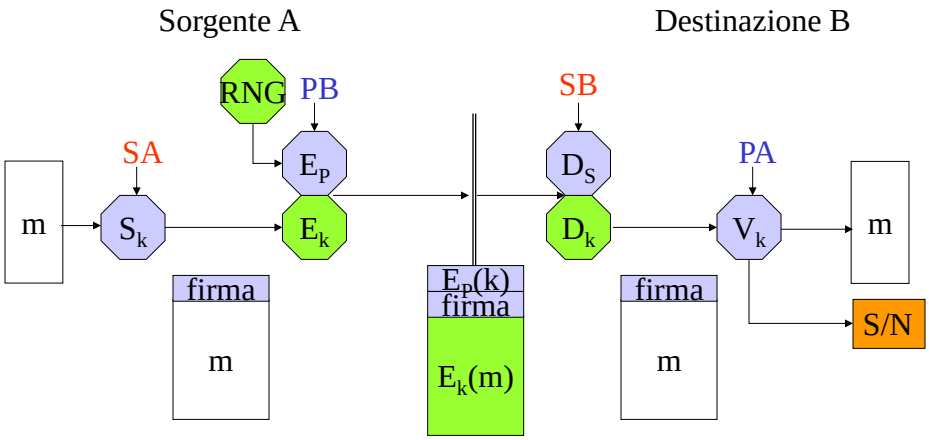
Summary of PGP Services

Function	Algorithm Used
Digital Signature	DSS/SHA or RSA/SHA
Message Encryption	CAST or IDEA or three-key triple DES with Diffie-Hellman or RSA
Compression	ZIP
E-mail Compatibility	Radix-64 conversion
Segmentation	

PGP: autenticazione o riservatezza



PGP: autenticazione & riservatezza

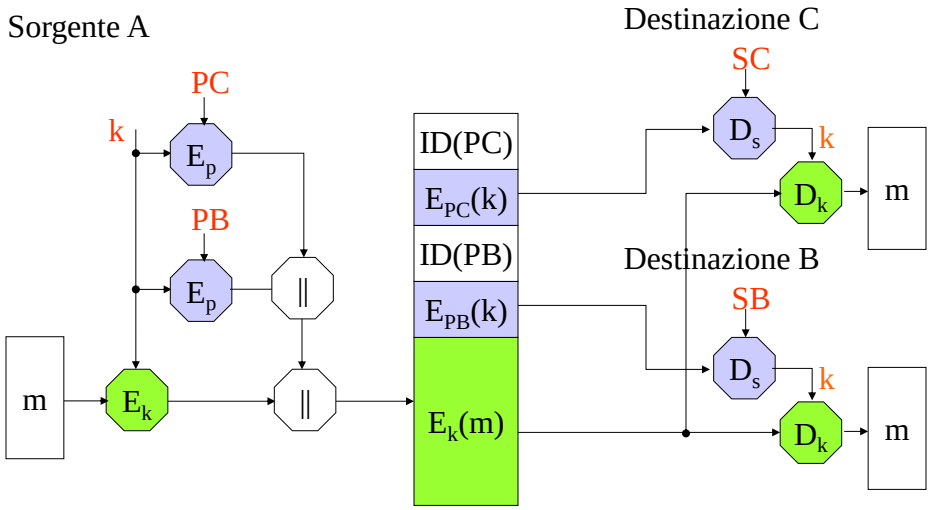


K chiave simmetrica
monouso

PGP: note

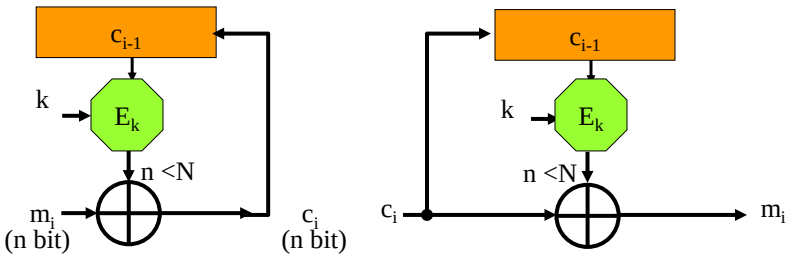
1. Firma con appendice; firme allegate o separate dal messaggio (vantaggi ad es. firme multiple, verifica di virus nel messaggio se file eseguibile)
2. Compressione per motivi di risparmio di trasmissione e memorizzazione.
    1. Nel caso della firma: compressione dopo la firma =>se cosi' non fosse ogni volta che verifico un messaggio dovrei ricomprimerlo oppure memorizzare m, compressione di m e firma; inoltre l'alg. di compressione non è deterministico, quindi quale è stato usato?
    2. Nel caso della cifratura: compressione prima della cifratura, elimino ridondanze eventuali nel messaggio

Messaggio riservato per più destinatari



Modalità di cifratura del PGP

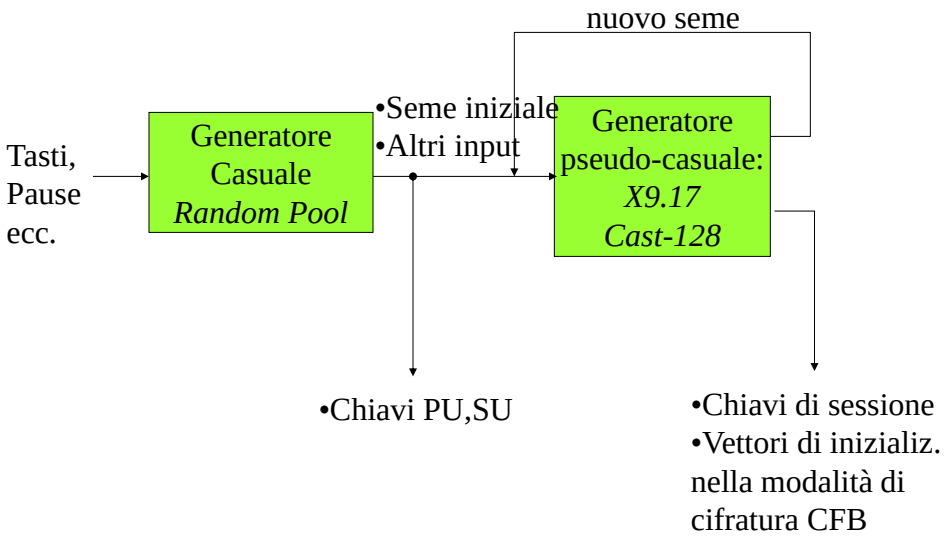
CFB (Chipher Feedback)



$\mathbf{c_i = m_i \oplus E_k(c_{i-1})}$ $\mathbf{m_i = c_i \oplus E_k(c_{i-1})}$

N = 64/128 bit
n = 64 bit

RNG e PRNG

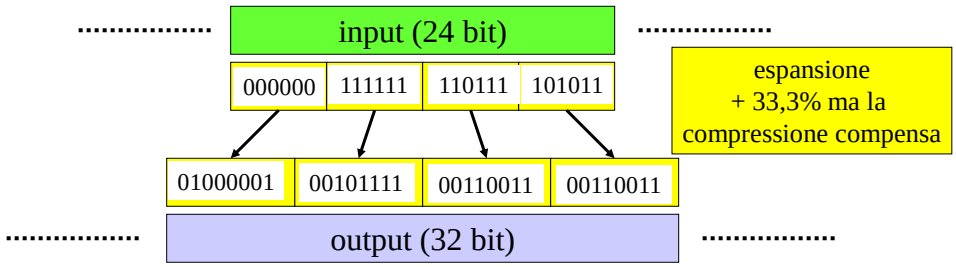


radix-64

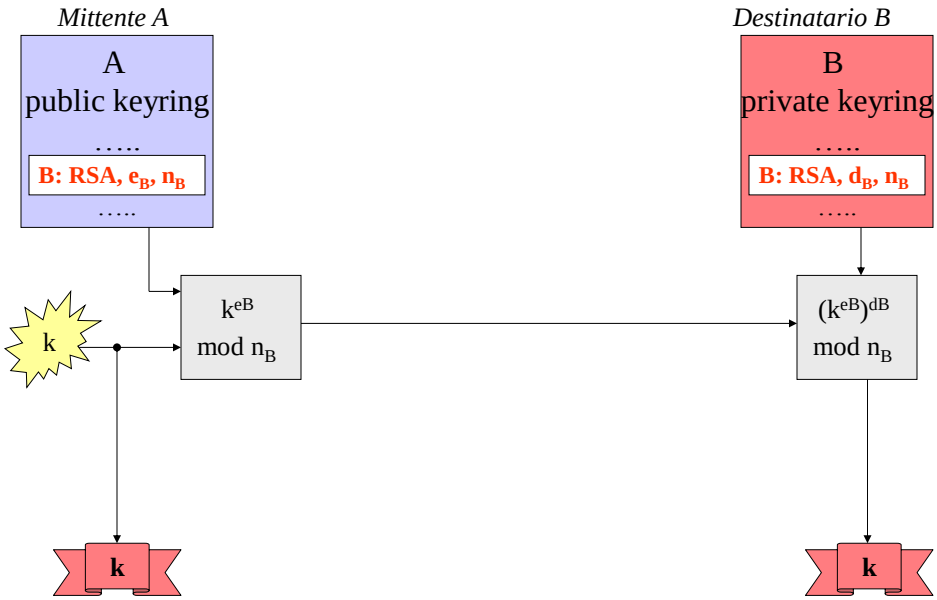
	000	001	010	011	100	101	110	111
000	A	B	C	D	E	F	G	H
001	I	J	K	L	M	N	O	P
010	Q	R	S	T	U	V	W	X
011	Y	Z	a	b	c	d	e	f
100	g	h	i	j	k	l	m	n
101	o	p	q	r	s	t	u	v
110	w	x	y	z	0	1	2	3
111	4	5	6	7	8	9	+	/

Alcuni sist. di posta elettr. permettono di usare solo blocchi di testo ASCII => conversione radix-64 (flusso di 8 bit binari in flusso di caratteri stampabili)

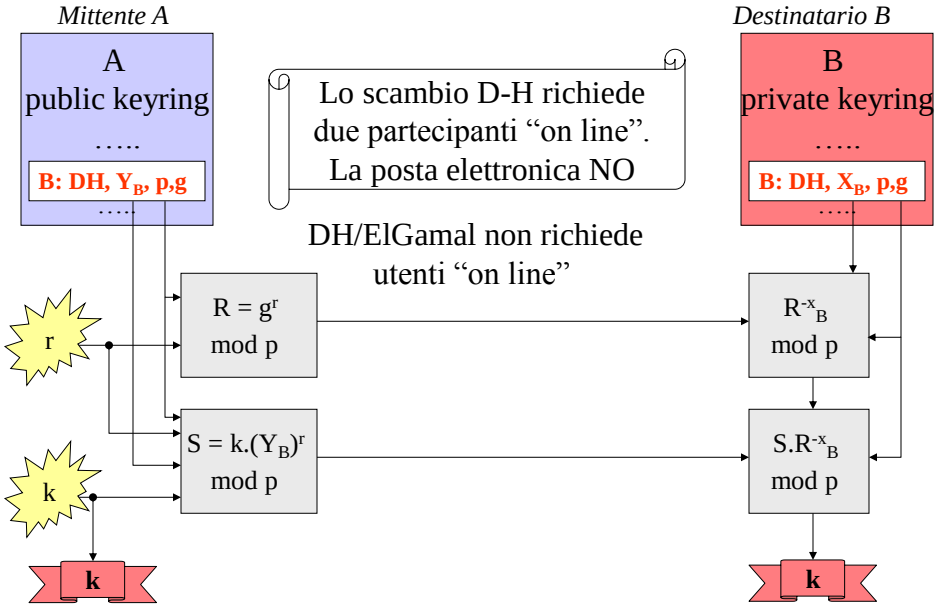
Blocchi di tre ottetti binari trasformati in 4 caratteri



Chiave di messaggio con RSA



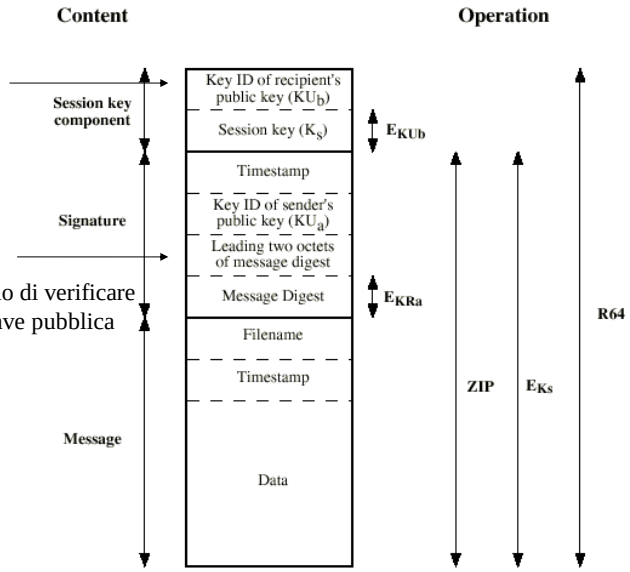
Chiave di messaggio con DH/ElGamal



Format of PGP Message

PGP pevede che ogni utente possa avere più coppie di chiavi. KeyID per risparmiare spazio Come creare un KeyID? (64 bit meno sign. KUbmod264)

Per permettere al destinatario di verificare se è stata utilizzata la chiave pubblica corretta



Struttura dei portachiavi

Media pesata: 1/X per ogni 'completam. Fidata' e 1/Y per ogni 'normalm. Fidato' (X e Y configurabili); se una firma è trusted KeyLeg è complete

codice utente

chiave pubblica in un certificato firmato da un intermediario

- Sconosciuto
- Completam. Fidato
- Normalm. Fidato
- Non fidato
- Ultimate trust

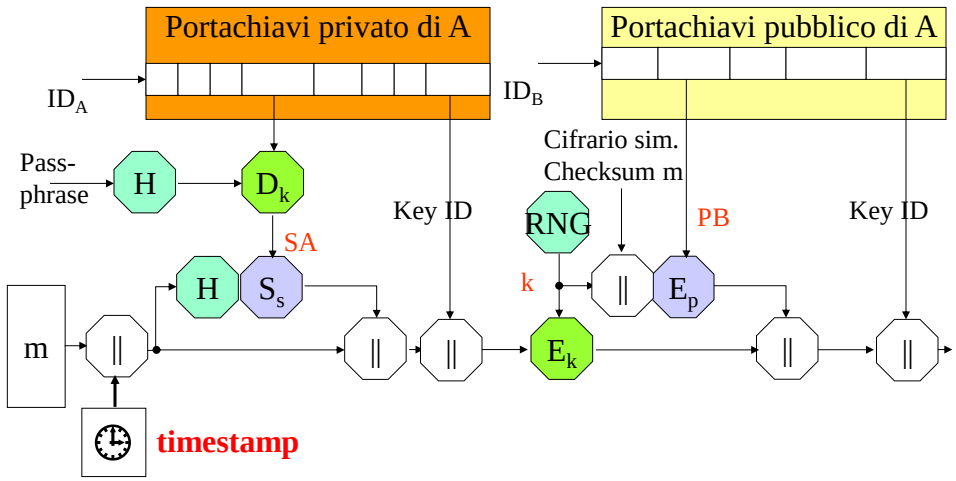
Private Key Ring

Timestamp	Key ID*	Public Key	Encrypted Private Key	User ID*
.	.	.	.	.
.	.	.	.	.
.	.	.	.	.
T_i	KU_i mod 2^{64}	KU_i	$E_{H(P_i)}[KR_i]$	User i
.	.	.	.	.
.	.	.	.	.
.	.	.	.	.

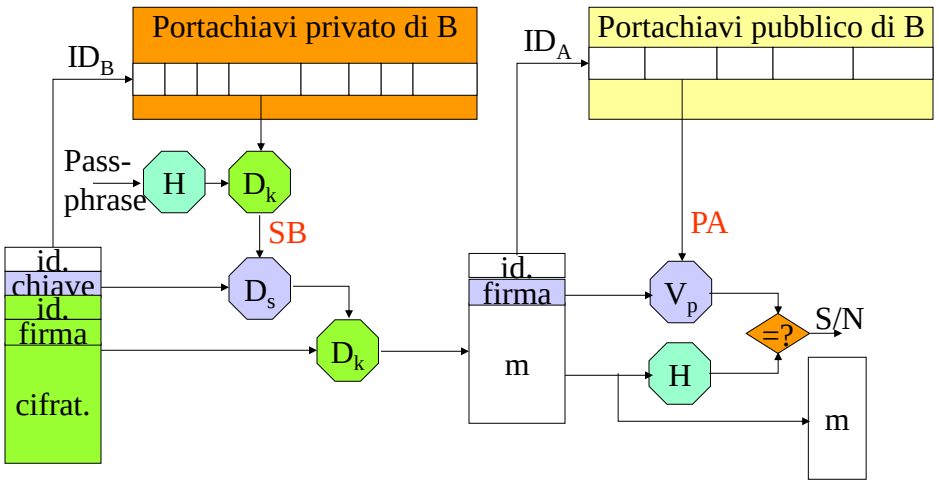
Public Key Ring

Timestamp	Key ID*	Public Key	Owner Trust	User ID*	Key Legitimacy	Signature(s)	Signature Trust(s)
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
T_i	KU_i mod 2^{64}	KU_i	trust_flag$_i$	User i	trust_flag$_i$		
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.

Mittente: portachiavi privato e pubblico



Destinatario: portachiavi privato e pubblico



La “vista” del portachiavi pubblico