Servizi Sicuri

Return

Indice

Introduzione

I servizi sicuri possono essere implementati a livello di applicazione, trasporto o rete. I principali a livello di applicazione sono:

- **Kerberos**
- **PGP**
- **TSS** A livello di trasporto:
- **SSL**
- **TLS** A livello di rete:
- **IPSec**

Timestamping - Marcatura Temporale

Un documento ha validità leegale se è attestata in maniera univoca la data e l'ora in cui è stato firmato.

Esistono due sistemi di riferimento internazionali per il calcolo del tempo: **TAI (Tempo Atomico Internazionale)** e **UTC (Tempo Universale Coordinato)**. In ogni caso, per il timestamping si ricorre sempre a una terza parte fidata.

Una società che offre un servizio di timestamping deve garantire:

- Tempo della marca non deve essere falso
- Tramite una marchatura deve essere possibile individuare un documento, un istante e un autore.
- Ogni modifica deve essere rilevabile in modo da evitare che la marcatura possa essere riutilizzata.
- Deve essere possibile marcare documenti riservati mantendendo la riservatezza.
- Chiunque deve poter marcare e validare la marca di un documento.

I passaggi per il timestamping sono:

1. Un entità A produce un messaggio m che vuole firmare.
2. Concatena m con il suo identificativo e ne fa l'hash per richiedere la marca temporale.
3. A $H(m||A)$ viene concatenata la marcatura temporale T
4. Il messaggio viene firmato dall'entità di certificazione con la sua chiave S_{TSS} in modo che la marcatura sia non ripudiabile.
5. A concatena la marcatura al messaggio m, ne fa l'hash e lo firma con la sua chiave S_A.

Kerberos

Kerberos è un protocollo di autenticazione che offre funzionalità di **Single Sign-On (SSO)**, consentendo agli utenti di accedere a più servizi senza dover reinserire le credenziali ogni volta. Utilizza soltanto funzioni di crittografia simmetrica, quindi è più performante ma meno scalabile.

- Garantire un'autenticazione sicura su reti non affidabili (es. Internet).
- Evitare l'invio diretto di password in rete.
- Utilizzare ticket temporanei e cifrati per accedere ai servizi.

Componenti Principali

- **Client (C):** Utente che vuole autenticarsi e accedere a un servizio.
- **Authentication Server (AS):** Autentica l'utente e rilascia un primo ticket per il TGS.
- **Ticket Granting Server (TGS):** Fornisce i ticket specifici per accedere ai servizi richiesti.
- **Server di Servizio (V):** Offre il servizio finale (es. file server, mail, ecc.).

Fasi del Protocollo

1. Autenticazione iniziale con AS

- Il client invia: `IDC || ADC || IDTGS || T1` (identità del client, indirizzo, TGS richiesto, timestamp).
- AS risponde con:
  - **Chiave di sessione (KCT)** cifrata con la chiave segreta del client.
  - **TicketTGS** cifrato con la chiave del TGS, contenente: `IDC || ADC || IDTGS || T2 || ΔT2`

2. Richiesta di ticket al TGS

- Il client invia al TGS: `IDV || ticketTGS || autenticatoreC` (l'autenticatore è cifrato con KCT e contiene IDC, ADC e un nuovo timestamp T3).
- TGS restituisce:
  - **KCV**, chiave di sessione client–server.
  - **TicketV** cifrato con la chiave del server V, contenente: `KCV || IDC || ADC || IDV || T4 || ΔT4`

3. Accesso al servizio (V)

- Il client invia a V: `ticketV || autenticatoreC` (cifrato con KCV e contenente IDC, ADC, T5).
- Il server risponde con: `EKCV(T5 + 1)` (conferma che l'autenticazione è avvenuta con successo).

Meccanismi di Sicurezza

- **Crittografia simmetrica:** tutte le comunicazioni avvengono con chiavi condivise.
- **Ticket e autenticatori:** permettono di evitare replay attack.
- **Lifetime:** ogni ticket è valido solo per un periodo limitato.
- **Autenticazione mutua:** sia client che server possono verificare l'identità dell'altro.

Vantaggi

- Le password non viaggiano mai in rete.
- L'autenticazione è centralizzata e scalabile.
- Adatto per ambienti distribuiti (es. reti aziendali, sistemi universitari).
- Supporta Single Sign-On (SSO): l'utente accede una sola volta e può usare più servizi.

PGP - Pretty Good Privacy

PGP è un protocollo per fornire autenticazione e riservatezza in ambito mail e file storage. In particolare fornisce 4 servizi:

- **Autenticazione:** verifica l'identità del mittente.
- **Confidenzialità:** cifratura del messaggio.
- **Compressione:** riduce la dimensione del messaggio.
- **Compatibilità:** supporta diversi formati di chiavi e algoritmi.

Autenticazione

Questo servizio viene realizzato facendo ricorso a 4 meccanismi:

- **RNG**: per generare le chiavi da utilizzare
- **Hashing**: per calcolare il digest del messaggio in modo da poterlo firmare
- **Firma e Verifica**: per garantire l'integrità e l'autenticità del messaggio

Confidenzialità

Il mantenibento della riservatezza del contenuto di un messaggio è ottenuto tramite:

- **RNG**: per generare le chiavi di cifratura
- **Cifratura**: per cifrare il messaggio con la chiave pubblica del destinatario
- **Decifratura**: per decifrare il messaggio con la chiave privata del destinatario

In particolare Cifratura e Decifratura possono essere metodi simmetrici o assimmetrici. Solitamente si usa la cifratura simmetrica per cifrare il messaggio e quella assimmetrica per cifrare la chiave simmetrica e scambaiarla con il destinatario.

Compressione

Viene utilizzata per ridurre la dimensione dei dati, solitamente in combinazione con le funzioni per riservatezza e autenticazione.

Nel caso di autenticazione, la compressione viene effettuata dopo la firma e la decompressione prima della verifica della firma. Nel caso di riservatezza, la compressione viene effettuata prima della cifratura e la compressione dopo la decifratura.

Compatibilità

PGP offre trasparenza alla applicazione email. I testi cifrati vengono convertiti in ASCII utilizzando Radix-64 che si occupa della segmentazione dei messaggi.

Tramite PGP è possibile ottenere sia autenticazione che riservatezza contemporaneamente, in questo caso la sequenza di operazioni è:

1. Generazione di una chiave di sessione k monouso.
2. Si firma il messaggio m con la chiave privata del mittente S_A.
3. Si cifra il messaggio m firmato con la chiave di sessione k.
4. Si cifra la chiave di sessione k con la chiave pubblica del destinatario P_B.
5. Si invia al destinatario il messaggio cifrato e la chiave cifrata $E_K(S_A(m))||E_{P_B}(k)$.
6. Il destinatario decifra la chiave di sessione k con la sua chiave privata S_B. Decifra il messaggio con la chiave di sessione k ottenendo m firmato. Verifica la firma con la chiave pubblica del mittente P_A.

Nel caso si voglia mandare lo stesso messaggio a più destinatari, si cifra la chiave di sessione k con le chiavi pubbliche dei destinatari e si invia il messaggio cifrato con la chiave di sessione k cifrata per ogni destinatario.

In PGP ogni utente ha due portachiavi, pubblico e privato.

Portachiavi privato:

In cui l'utente ha le proprie chiavi private.

È strutturato come una tabella con le seguetni colonne:

- **Timestamp**: data di creazione della chiave, non uninversale.
- **Key ID**: identificativo della chiave, è un hash della chiave pubblica.
- **Public Key**: chiave pubblica dell'utente associata alla privata.
- **Encrypted Private Key**: chiave privata cifrata con la passphrase dell'utente.
- **User ID**: identificativo dell'utente, può essere un indirizzo email o un nome.

Portachiavi pubblico:

Contiene le chiavi pubbliche dell'utente e degli altri utenti con cui si scambiano messaggi cifrati.

Le colonne della tabella sono:

- **Timestamp**
- **Key ID**
- **Public Key**
- **Owner Trust**: fiducia iniziale che io ripopngo in una data chiave pubblica (0 = nessuna fiducia, 1 = fiducia parziale, 2 = fiducia completa).
- **User ID**
- **Key Legitimacy**: PGP calcola automaticamente sulla base di Owner Trust e Signature Trust
- **Signatures**: firme di altre chiavi pubbliche che attestano la legittimità della chiave pubblica.
- **Signature Trust**: fiducia che gli altri utenti ripongono nella chiave pubblica, calcolata in base alle firme ricevute.