Appunti Lezione 20/11/2025

Ionut Georgian Zbirciog - Christian Sfeir

20/11/2025

Indice

1 Autenticazione - AAA **2**
1.1 Fattori di Accesso 3
1.2 Password e Accesso Biometrico 3
1.2.1 Password vs Impronta Digitale 4
1.2.2 Falsi Positivi o Falsi Negativi? 4

2 Identità Federata **6**
2.1 Come Funziona? 7
2.2 Approfondimento sulla Struttura del Token 8
2.2.1 Struttura del Token (Esempio JWT) 8
2.2.2 Meccanismo di Verifica e Affidabilità 9

1 Autenticazione - AAA

L'**autenticazione** è il processo attraverso il quale un sistema informatico **verifica l'identità** dichiarata di un utente, un dispositivo, un'applicazione o qualsiasi altra entità che cerca di accedere a una risorsa (come un computer, una rete, un'applicazione o un database).

In sostanza, l'autenticazione risponde alla domanda: *"Sei davvero chi dici di essere?"*

Il framework di sicurezza **Tripla A (AAA)** è un pilastro fondamentale nella gestione delle reti e dei sistemi informatici. Il nome deriva dalle iniziali dei tre servizi essenziali che coordina per controllare l'accesso, applicare le policy e monitorare l'utilizzo delle risorse: **Autenticazione**, **Autorizzazione** e **Accounting** (Contabilità). Questi tre processi lavorano in sequenza e armonia per garantire un ambiente sicuro e tracciabile.

1. **Autenticazione (Authentication)**

   Questo è il primo passo e serve a **verificare l'identità** dell'entità che richiede l'accesso (che sia un utente, un dispositivo o un'applicazione). In termini semplici, risponde alla domanda cruciale: *"Sei davvero chi dici di essere?"*

   - Il suo **scopo** principale è stabilire se l'identità dichiarata è legittima.
   - I **meccanismi** di verifica si basano su uno o più fattori di prova, come qualcosa che l'utente *conosce* (es. password, PIN), qualcosa che *possiede* (es. token, smartphone per OTP) o qualcosa che *è* (es. dati biometrici).
   - Se la prova è valida, l'entità viene riconosciuta e può procedere al passo successivo.

2. **Autorizzazione (Authorization)**

   Una volta che un'entità è stata autenticata con successo, l'autorizzazione entra in gioco per definire **cosa può effettivamente fare**. Questo processo determina a quali risorse specifiche l'utente ha accesso e quali operazioni gli sono permesse. La domanda a cui risponde è: *"Cosa puoi fare qui?"*

   - Lo **scopo** è applicare le policy di accesso stabilite, spesso basate sul ruolo o sul livello di privilegio dell'utente.
   - Ad esempio, un *amministratore* sarà autorizzato a eseguire operazioni di lettura/scrittura sui file di configurazione, mentre un *utente standard* potrebbe essere limitato alla sola lettura su determinate aree.
   - È fondamentale ricordare che essere autenticati non implica automaticamente essere autorizzati a ogni azione; i permessi sono granulari.

3. **Accounting (Contabilità)**

   Questo è l'aspetto di tracciamento del framework. L'accounting è il processo di **registrazione e rendicontazione** di tutte le attività che un utente autenticato e autorizzato compie. Risponde alla domanda: *"Cosa hai fatto e per quanto tempo?"*

   - Lo **scopo** primario è la sicurezza (audit, analisi forense), ma anche la gestione delle risorse (fatturazione, pianificazione della capacità).

- Le **informazioni registrate** includono l'orario esatto di accesso e disconnessione, i servizi o le risorse utilizzate, la quantità di dati trasferiti e, in contesti critici, i comandi specifici eseguiti.
- Questo registro fornisce la **traccia forense** essenziale per ricostruire gli eventi in caso di violazioni o problemi operativi.

1.1 Fattori di Accesso

Il processo di autenticazione si basa tipicamente sulla presentazione e verifica di **fattori di prova** noti solo all'entità legittima. Questi fattori sono classificati in tre categorie principali:

1. **Qualcosa che conosci (Knowledge Factor)**
   Sono le informazioni segrete che solo l'utente conosce.
   - *Esempi:* Password, PIN, risposte a domande di sicurezza. Ovviamente in questo caso, il server non conosce comunque veramente la password in quanto è "hashata" all'interno del database.
2. **Qualcosa che possiedi (Possession Factor)**
   Sono gli oggetti fisici o digitali che solo l'utente dovrebbe avere.
   - *Esempi:* Chiavi di sicurezza hardware (token USB), smart card, smartphone che riceve un codice monouso (OTP).
3. **Qualcosa che sei (Inherence Factor)**
   Sono le caratteristiche biologiche uniche dell'utente (biometria).
   - *Esempi:* Impronta digitale, scansione della retina o dell'iride, riconoscimento facciale, riconoscimento vocale.

I sistemi di sicurezza più robusti utilizzano l'**Autenticazione a Fattore Multiplo (MFA)**, che richiede all'utente di fornire prove da **due o più fattori diversi** (ad esempio, una password *e* un codice OTP inviato al telefono) per aumentare significativamente la sicurezza.

1.2 Password e Accesso Biometrico

La distinzione principale tra l'uso di una password e un'impronta digitale (o il riconoscimento facciale) come metodo di accesso risiede nella loro natura di verifica: deterministica contro statistica. La verifica della password è un processo deterministico (o binario): o la sequenza di caratteri corrisponde esattamente a quella archiviata nel sistema, oppure l'accesso è negato. Non ci sono margini di errore. Al contrario, l'autenticazione tramite dati biometrici è intrinsecamente statistica e basata sulla probabilità. Il sistema non verifica una corrispondenza perfetta, ma calcola una percentuale di somiglianza tra la scansione attuale e il modello registrato. Per questo motivo, fattori esterni come l'umidità delle mani o l'uso di occhiali possono ridurre la percentuale di somiglianza al di sotto della soglia richiesta, portando al fallimento dell'accesso nonostante l'utente sia quello legittimo.

1.2.1 Password vs Impronta Digitale

La probabilità che due persone abbiano la stessa impronta digitale è considerata praticamente nulla. Tuttavia, nei sistemi di sicurezza, il punto critico non è la somiglianza fisica, ma il processo di **discretizzazione** o **estrazione delle feature**. L'impronta viene convertita in un modello matematico (punti di minuzia e schemi) per la memorizzazione e la verifica. Quando si confrontano due impronte, si verifica se i loro modelli discretizzati coincidono. La risposta a questa domanda è sì: a causa di un'acquisizione imperfetta (dita sporche, sensore difettoso) o della limitata risoluzione del sistema, è teoricamente possibile che due impronte distinte generino per errore lo stesso modello matematico discretizzato, causando un falso positivo nel sistema.

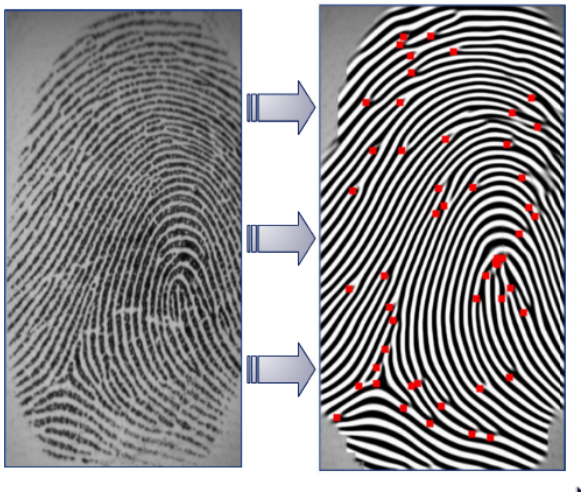

Figura 1: Discretizzazione dell'Impronta Digitale

1.2.2 Falsi Positivi o Falsi Negativi?

Supponiamo di dover progettare il sistema di accesso ad un'aula universitaria. Quando si progetta il sistema di accesso a un'aula universitaria, la scelta tra un **Falso Positivo (FP)** e un **Falso Negativo (FN)** dipende dalla tolleranza al rischio e dall'obiettivo primario.

Poiché l'obiettivo è massimizzare la partecipazione e l'esperienza positiva degli studenti, si tende a preferire i Falsi Positivi. Un FP si verifica quando si permette l'accesso a qualcuno che non dovrebbe essere lì (un non iscritto). Pur essendo un'imperfezione, in un contesto universitario questo errore è generalmente meno grave; il non iscritto può essere identificato in seguito. Al contrario, un Falso Negativo si verifica quando si blocca l'accesso a uno studente regolarmente iscritto. Questo errore causa un danno immediato e diretto all'utente legittimo, impedendogli di frequentare la lezione, un risultato che in un ambiente educativo è considerato più dannoso e inaccettabile. Di conseguenza, si imposta la soglia di sicurezza per minimizzare i blocchi ingiustificati (FN), accettando un aumento dei FP.

Supponiamo adesso di trovarci in un contesto di **sicurezza di frontiera** o controllo di accesso a uno stato, l'equilibrio tra Falsi Positivi (FP) e Falsi Negativi (FN) si inverte rispetto all'aula universitaria.

In questo scenario, si preferisce tollerare i Falsi Negativi (FN). Un FN si verifica quando un cittadino o un viaggiatore legittimo viene temporaneamente bloccato o sottoposto a un

controllo aggiuntivo, un inconveniente che può essere gestito con una verifica secondaria, dalla polizia di frontiera. Al contrario, il Falso Positivo (FP), ovvero ammettere un individuo non autorizzato, non idoneo o potenzialmente pericoloso, comporta un rischio per la sicurezza nazionale e la sovranità dello Stato . Pertanto, i sistemi di frontiera sono tarati per essere più restrittivi, minimizzando la possibilità di far entrare persone non idonee a scapito di un aumento dei ritardi per i viaggiatori legittimi.

2 Identità Federata

L'Identità Federata è un concetto architetturale di sicurezza che definisce un'alleanza o un rapporto di fiducia digitale tra due o più domini di sicurezza e organizzazioni indipendenti.

In pratica, si ha Identità Federata quando un'organizzazione, il Fornitore di Servizi (SP), accetta di delegare il compito dell'autenticazione di un utente a un'altra organizzazione autorevole, il Fornitore di Identità (IdP).

- **L'Utente (Principal)**
  È semplicemente l'individuo o l'entità che desidera accedere a una risorsa (l'utente finale).
- **Il Fornitore di Identità (Identity Provider - IdP)**
  L'identity Provider è un componente specifico o un servizio che fa parte dell'ecosistema **Identity Manager - IdM** e ha il solo compito di autenticare l'utente e rilasciare un'attestazione di identità (il token).
  - *Esempi comuni:* Servizi come Google o Facebook (per login social) oppure il sistema di login centralizzato di una grande azienda.
- **Il Fornitore di Servizi (Service Provider - SP)**
  Questa è l'applicazione o il sito web a cui l'utente vuole accedere (ad esempio, una piattaforma di e-learning o un'applicazione aziendale esterna). L'SP non chiede la password all'utente, ma si affida alla conferma di identità fornita dall'IdP.

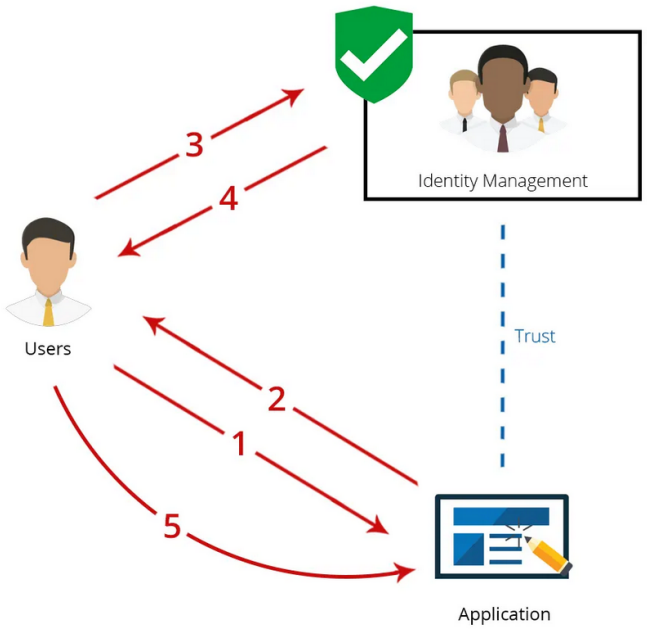

Figura 2: Schema Riassuntivo

2.1 Come Funziona?

Vediamo ora nel dettaglio come avviene il meccanismo di autenticazione.

Supponiamo che un utente, tramite il suo browser deve accedere ad una risorsa che si trova all'interno di un web-server (`www.example.it/resource`), e ottenere questa risorsa è necessario che sia autenticato.

1. **Richiesta di Accesso Iniziale:** L'Utente, non autenticato, tenta di accedere alla risorsa desiderata sul **Fornitore di Servizi (Web Server/SP)**.

   ```
   GET www.example.it/resource HTTP/1.1
   ```

2. **Reindirizzamento all'IdP (Redirect):** Il Web Server (SP) riconosce l'assenza di autenticazione e risponde, reindirizzando il browser dell'utente al **Fornitore di Identità (IdP)**.

   ```
   HTTP/1.1 302 Found
   location: https://www.idm.it/login?return=/resource
   ```

3. **Autenticazione e Generazione del Token:** Il browser si collega all'IdP per completare l'accesso.

   3.1. **Scambio del Form:** Il browser richiede e riceve la pagina di login dall'IdP.

   ```
   GET https://www.idm.it/login?return=/resource HTTP/1.1
   HTTP/1.1 200 OK
   ```

   3.2. **Invio Credenziali (con il Form):** L'Utente compila il form e lo invia all'IdP.

   ```
   <form action="/login" method="post">
       <input type="text" name="username" required>
       <input type="password" name="password" required>
       <input type="hidden" name="return" value="/resource">
       <input type="submit" value="Effettua Accesso">
   </form>
   ```

   3.3. **Ritorno con Token (Asserzione):** Una volta autenticato, l'IdP genera un **token di sicurezza cifrato** (o **Assertion**) che attesta l'identità dell'utente e lo usa per reindirizzare il browser al Web Server.

   ```
   HTTP/1.1 302 Found
   location: /resource?token=username+63xa3lgf3klas
   ```

4. **Invio del Token all'SP:** Il browser esegue il reindirizzamento e invia il token di verifica appena ricevuto al Web Server (SP), richiedendo nuovamente la risorsa, ma questa volta con la prova di identità.

   ```
   GET www.example.it/resource?token=username+63xa3lgf3klas HTTP/1.1
   ```

5. **Verifica Finale e Concessione Accesso:** Il Web Server (SP) decifra il token, verifica l'autenticità (fidandosi dell'IdP) e, dopo aver confermato l'identità, risponde con la risorsa richiesta.

A questo punto, per evitare di ripetere l'intero processo di autenticazione ogni volta che l'utente richiede una nuova pagina, il Web Server (SP) imposta un cookie di sessione.

1. **Cosa sono i Cookie?** I cookie sono piccoli file di testo che un sito web salva sul browser dell'utente. Essi servono come "promemoria" o etichette identificative.
2. **Come Funzionano?** Il cookie di sessione (spesso contenente un identificativo casuale e cifrato) viene inviato dal Web Server al browser. A partire da quel momento, per ogni richiesta successiva alla risorsa, il browser allega automaticamente questo cookie.
3. **Vantaggio:** Il Web Server non deve più fare il redirect all'IdP. Invece, legge il cookie, verifica che sia valido e non scaduto (e che corrisponda a una sessione attiva), e concede immediatamente l'accesso, mantenendo così la sessione di navigazione attiva e continua per l'utente. Questo consente il *Single Sign-On (SSO)* all'interno del dominio del Service Provider per tutta la durata della sessione.

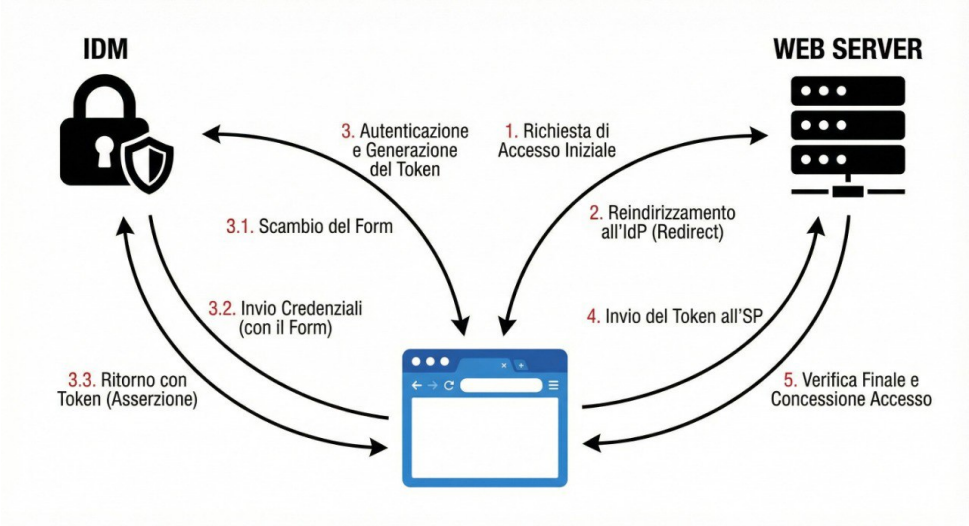

Figura 3: Schema Identità Federata

2.2 Approfondimento sulla Struttura del Token

Dopo aver ricevuto il token di sicurezza, il **Web Server (SP)** deve verificare la sua autenticità senza contattare nuovamente l'IdP. Questo è il cuore della Federazione dell'Identità.

Il token è un pacchetto di dati strutturato. Il meccanismo di verifica si basa sulla **crittografia** e, in particolare, sull'utilizzo di token strutturati come i **JWT (JSON Web Token)**.

2.2.1 Struttura del Token (Esempio JWT)

Lo standard JWT divide il token in tre parti distinte, separate da un punto (`.`):

$$Header.Payload.Signature$$

1. **Header (Intestazione)** Contiene i metadati sul token, specificando l'algoritmo utilizzato per la firma.
   - **Tipo di Token (`typ`):** Es. `JWT`.
   - **Algoritmo di Firma (`alg`):** Es. `RS256` (RSA con SHA-256) o `HS256` (HMAC con SHA-256).

2. **Payload (Corpo / Claim)** Contiene le informazioni effettive, chiamate **Claim**, che sono le asserzioni di identità rilasciate dall'IdP.
    - **Claim Registrati (Registered Claims - Metadati):**
        - **Emittente (`iss`):** Identifica l'IdP che ha emesso il token (`https://www.idm.it`).
        - **Soggetto (`sub`):** L'identificativo unico dell'utente.
        - **Destinatario (`aud`):** Il Servizio Provider (SP) per cui il token è valido.
        - **Durata/Scadenza (`exp`):** Il timestamp che definisce quando il token non è più valido.
        - **Data di Emissione (`iat`):** Il timestamp di creazione del token.
3. **Signature (Firma Digitale)** Questa componente garantisce l'integrità del token. Viene creata applicando l'algoritmo specificato nell'Header all'Header e al Payload codificati, utilizzando una **Chiave Segreta** nota solo all'IdP.
4. **Claim Pubblici/Privati:** Contengono dati aggiuntivi sull'utente e i suoi permessi (es. ruolo, nome, email).

2.2.2 Meccanismo di Verifica e Affidabilità

Il Web Server si fida del token grazie alla **firma digitale**, che è il meccanismo della "fiducia digitale" stabilita in anticipo .

- **Crittografia Simmetrica (Es. Firma HS256)**
    - **Meccanismo:** L'IdP e l'SP concordano **in anticipo** su un'unica **Chiave Segreta Condivisa**.
    - **Verifica:** L'SP utilizza la *stessa Chiave Segreta Condivisa* per ricalcolare la firma. Se la firma ricalcolata corrisponde a quella presente nel token, il token è autentico e non alterato.
- **BCrittografia Asimmetrica (Metodo Standard - Es. Firma RS256)** Questo è il metodo preferito perché non richiede la condivisione diretta di una chiave segreta.
    - **Chiavi:** L'IdP possiede una **Chiave Privata** segreta (per firmare) e condivide la **Chiave Pubblica** con l'SP.
    - **Verifica:** L'SP utilizza la *Chiave Pubblica* dell'IdP per decifrare e convalidare la firma. Solo se la firma è stata creata con la Chiave Privata corrispondente, la verifica ha successo, garantendo l'origine certa del token.

Una volta verificata la firma, l'SP legge il Payload per controllare le condizioni di validità, in particolare la **scadenza (`exp`)** e il **destinatario (`aud`)**. Solo se tutti i criteri sono soddisfatti, l'accesso viene concesso.