Gestione e protezione di identità e accessi

Una panoramica delle tecniche disponibili per preservare la sicurezza degli accessi ai servizi online

DI UGO LOPEZ*

La pandemia ha avuto diverse conseguenze sulla vita delle persone, una su tutte il forte impulso all'utilizzo di nuove tecnologie. In particolare, la necessità di ricorrere all'*e-learning* e allo *smart working* ha fortemente contribuito all'adozione delle tecnologie in *cloud*, con particolare riguardo alla modalità **SaaS** (*Software as a Service*): in ambienti a bassissimo impatto tecnologico, ricorrere a risorse gestite principalmente da *cloud provider* e persino gratuite per scuole, università e associazioni no-profit era l'unica scelta possibile. In **Figura 1** vengono esemplificati i servizi gestiti da un *cloud provider* in caso di architettura **IaaS** (*Infrastructure as a Service*), **PaaS** (*Platform as a Service*) e **Saas**: come è facile notare, l'architettura SaaS (utilizzata da notissimi servizi come Google Workspace e Microsoft 365) è quella in cui tutti i *layer* sono gestiti dal *provider*, lasciando all'utente il solo utilizzo delle applicazioni.

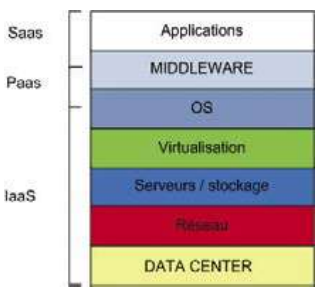

Figura 1. Servizi gestiti dal Cloud provider
(Fonte: Wikimedia Commons)

IL PROBLEMA DELLA SINGLE SIGN ON

Per quanto riguarda la sicurezza degli accessi, il primo problema che si incontra è quello del *Single Sign On* (**SSO**): tutti noi usiamo le credenziali di accesso alle nostre applicazioni web come Facebook, Gmail, Outlook etc. per accedere ad altre applicazioni: questo approccio, se, da un lato, porta a centralizzare la sicurezza degli accessi in pochi "accentratori", dall'altro comporta che la compromissione del nostro *account* Facebook: solo per fare un esempio, compromette tutti i servizi a cui accediamo autenticandoci in SSO proprio con lo stesso Facebook, oltre a dare accesso a Meta ad alcuni dati degli altri servizi che ne sfruttano l'autenticazione.
Il SSO può essere effettuato con 3 differenti approcci:

- approccio centralizzato;
- approccio federativo;
- approccio cooperativo.

Nel primo caso, vi è un DB unico per molti servizi: la compromissione del DB, quindi, comporta la compromissione di tutti i servizi che si appoggiano al DB. Nel secondo caso, l'accesso è gestito da uno dei servizi coinvolti: anche in questo caso, la compromissione del servizio garante per l'accesso, comporta anche la compromissione degli altri servizi coinvolti. L'ultimo caso, molto simile al secondo (ma con un gestore per ciascun servizio), comporta analoga insicurezza.

Figura 2. YubiKey 5 NFC

LA FUNZIONE MATEMATICA DI HASH

Un primo livello di protezione è dato dall'applicazione della **funzione matematica di hash** alle *password*: in pratica, quando inseriamo la *password*, per esempio, nel nostro computer, la stessa viene confrontata con la *password* che abbiamo originariamente impostato e, se coincidenti, veniamo autenticati. Ovviamente, questo comporta per il sistema di dover mantenere in memoria la *password* impostata, con conseguenti rischi legati alla sicurezza. Per ovviare a questo, la *password* viene memorizzata dopo essere stata cifrata con una funzione di hash, la quale ha due principali caratteristiche:

- non è invertibile;
- sebbene operi una forte riduzione, da dominio infinito a codominio a dimensione fissa, ha una probabilità di output non univoco assai bassa (tanto più bassa quanto è più robusto l'algoritmo).

Inoltre, la *password* scelta dall'utente viene prima "salata", ovvero si applica la tecnica di "*salt*" che consiste nell'aggiungere una stringa pseudorandomica in testa o in coda alla *password*, e poi cifrata con *hash*, sebbene questa tecnica, in caso di compromissione del servizio, renda ancora più facile la decifrazione delle *password*, di per sé già non complessissima e ottenibile attraverso collaudate tecniche quali l'attacco a forza bruta, l'attacco a dizionario e altri.
Un ulteriore livello di protezione, oggi assai diffuso, è quello dell'autenticazione multifattore (**MFA**, *MultiFactor Authentication*): alla classica autenticazione con nome utente e *password* vengono aggiunti ulteriori fattori di autenticazione come una stringa numerica generata da un *token hardware*, inviata a mezzo SMS/email, dettata via telefono, etc. In questa maniera, la compromissione della *password* non è più sufficiente a violare un *account* che, però, può comunque essere violato con alcuni tipi di attacco come per esempio il *phishing*, *smishing* *vishing*, etc.

IL RISCHIO CONNESSO AL RIPRISTINO AUTONOMO DELLA PASSWORD

In un contesto già poco sicuro, inoltre, si aggiungono altri fattori di rischio. Il primo è la possibilità che alcuni sistemi offrono agli utenti di ripristinare autonomamente le proprie *password* (**SSPR**, *Self Service Password Reset*): questo metodo, specie in alcune modalità come le domande di sicurezza, consentono a un attaccante sprovvisto di *password* di ricavarla. Proprio per questa ragione, non è mai prudente partecipare ai classici giochi *online* in cui vengono fatte domande personali (per esempio, data di nascita in cambio di oroscopo), spesso veri e propri strumenti di ingegneria sociale per raccogliere informazioni personali sugli utenti nella fase preparatoria all'attacco. Anche in modalità più sicura, però, ci sono dei problemi di sicurezza: il secondo fattore di sicurezza inviato via SMS, ad esempio, può essere soggetto ad attacco di *SIM swapping*.
Altri servizi, invece, memorizzano le *password* dei propri utenti in chiaro: è facile accorgersi della circostanza quando, una volta persa la *password*, tentate di ripristinarla ma, anziché riceverne una provvisoria da cambiare al primo accesso, ricevete proprio la *password* che avevate dimenticato.

TECNICHE DI PROTEZIONE ACCOUNT

In realtà esistono anche alcune tecniche, specie in ambito *enterprise*, per la protezione degli *account*:

- l'accesso contestuale (Just in Time Access), ovvero fornire privilegi amministrativi a un utente solo per il tempo strettamente necessario a compiere una determinata operazione che li richieda;
- integrazione di algoritmi di machine learning che valutino la sicurezza degli accessi e degli account sulla base di una serie di parametri come accessi quasi simultanei da IP geograficamente lontanissimi, accessi da IP anonimi, accessi da account compromessi, etc;
- suddivisione di azioni amministrative tra più utenti, in maniera tale da non dare piena visibilità di un'operazione a un unico amministratore;
- analisi periodica dei log e impostazione di avvisi immediati in caso di azioni pericolose.

In termini più generali, il problema fondamentale è quello di un eventuale *data breach* o *data leak* che comprometterebbe, in un sol colpo, tanti più utenti quanti più *account* sono memorizzati nel sistema. Sebbene finora esistano dei servizi che ci consentono di sapere se e quali tra i nostri *account* sono stati violati, il numero di attacchi che colpiscono nel segno aumenta di giorno in giorno, complice l'autenticazione centralizzata verso la quale tutte le organizzazioni si sono ormai orientate per l'evidente riduzione del carico amministrativo.

L'AUTENTICAZIONE PASSWORDLESS O FRICTIONLESS

Una soluzione drasticamente più efficace di quelle finora esaminate è quella dell'autenticazione *passwordless* (o *frictionless*): sistemi quali **Windows Hello** e **Windows Hello for Business** o *token* con il supporto per protocolli quali il **fido2** o similari (**Figura 2**) hanno un indiscutibile vantaggio rispetto alle altre forme di autenticazione: attraverso una "relazione di fiducia" che viene stabilita tra il sistema informatico e il *device* dell'utente, le credenziali di autenticazione vengono conservate sul *device* dell'utente e utilizzate per la sola fase di autenticazione, in maniera tale da non essere tra i dati esfiltrati in caso di attacco al sistema a cui si accede. Oltre al vantaggio di avere un'autenticazione centralizzata senza centralizzazione delle credenziali e a poter accedere in modalità *"anywhere/anytime"*, questo approccio presenta ulteriori numerosi vantaggi, quale quello di poter cifrare il proprio dispositivo con robusti algoritmi crittografici, grazie anche ai notevoli miglioramenti delle periferiche negli ultimi anni, come per esempio i **chip TPM**, in maniera tale da rendere difficilmente accessibile anche la singola macchina su cui sono memorizzate le credenziali.
In questo contesto, anche il pin numerico a 4/6 cifre, apparentemente insicuro, offre invece un maggiore livello di sicurezza, sia perché facile da ricordare, sia perché specifico del *device* (e, ancora, non conservato centralmente). L'autenticazione biometrica e il "pin corto", inoltre, evitano all'utente di dover ricordare (o scrivere) complesse *password* o, viceversa, di utilizzare *password* facilmente decifrabili, a tutto svantaggio di sicurezza.
In ultimo, i sistemi MDM spesso consentono il controllo remoto della periferica che, in caso di smarrimento o furto, può essere completamente ed efficacemente "ripulita", con la totale protezione di dati (spesso presenti comunque solo nei *cloud storage*) e delle credenziali di accesso.
In conclusione, è facile notare come, anche analizzando solo un aspetto della gestione dei sistemi *cloud* SaaS (**IAM**, *Identity and Access Management*), ci si stia rapidamente muovendo verso nuove forme di protezione dei dati e, soprattutto, delle identità degli utenti.

*COMPONENTE GDL CYBERSECURITY