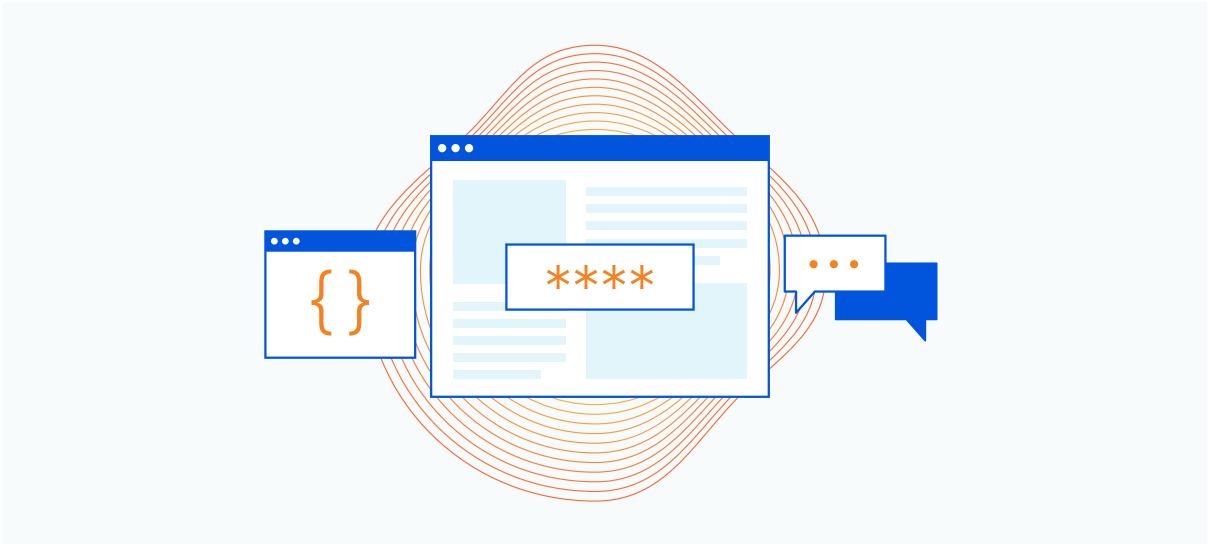
WHITE PAPER

CLOUDFLARE

Una guida alla sicurezza delle API

+44 20 3514 6970 | enterprise@cloudflare.com | www.cloudflare.com/it-it/

Le API fanno girare il mondo (delle app).



Sappiamo tutti ormai che le API, le interfacce di programmazione delle applicazioni, fanno girare il mondo. Più precisamente, consentono a diverse applicazioni moderne di comunicare tra loro. Le applicazioni mobili o Web possono accedere a un back-end in cui i dati vengono archiviati ed elaborati. Le API possono essere pubbliche consentendo alle applicazioni di diverse aziende di comunicare o private, il che è abbastanza comune, in cui le applicazioni interne si integrano per raggiungere gli obiettivi aziendali.

Il risultato sarà applicazioni, siti Web e app mobili più robusti e completi con funzionalità più ampie e dati più diversificati.

Ad esempio, invece di creare i propri servizi di pagamento da zero, le società di ride sharing possono aggiungerli tramite le API delle società di pagamento. Un altro esempio sono i siti di aggregazione di voli. Per mostrarci gli orari dei voli, i prezzi, le destinazioni e tutto ciò che dobbiamo sapere su un biglietto aereo si collegano tramite chiamate API ai database delle compagnie aeree per estrarre i dati appropriati e visualizzarli nella nostra pagina dei risultati di ricerca dell'aggregatore.

L'importanza delle API sta crescendo, con sempre più aziende che si descrivono come "API-first". In alcuni casi, il prodotto reale di un'azienda è un'API con un modello di business incentrato sul consumo. Ad esempio, se un'azienda che fornisce dati meteorologici ha fatto sì che la sua API diventasse il proprio prodotto, altre aziende che desiderano informazioni meteorologiche pagheranno a questa azienda una tariffa mensile per l'accesso all'API. In molti altri casi, visto che un'applicazione deve interagire con altre applicazioni, le API sono progettate insieme o anche prima del codice che fornisce le funzionalità effettive del prodotto, non sono legate alla fine dello sviluppo.

Le aziende dedicano tempo e sforzi per elaborare deliberatamente il loro approccio API per considerare come esporre i dati giusti, diventando un elemento fondamentale per i ricavi e i modelli di business.

Tuttavia, la creazione di API perfette è difficile, perché proprio come qualsiasi software, si verificheranno delle vulnerabilità che portano a problemi di sicurezza di cui parleremo in questo documento.

Basti dire che le API sono ovunque e acquisiranno slancio solo nei prossimi anni e devono essere protette. Per questo motivo, quando pensiamo alla sicurezza delle API dell'organizzazione. esamineremo gli attacchi alle API e gli aspetti della difesa in profondità.

Slancio delle API in numeri

Il 50%

del traffico che scorre attraverso Cloudflare è il traffico API

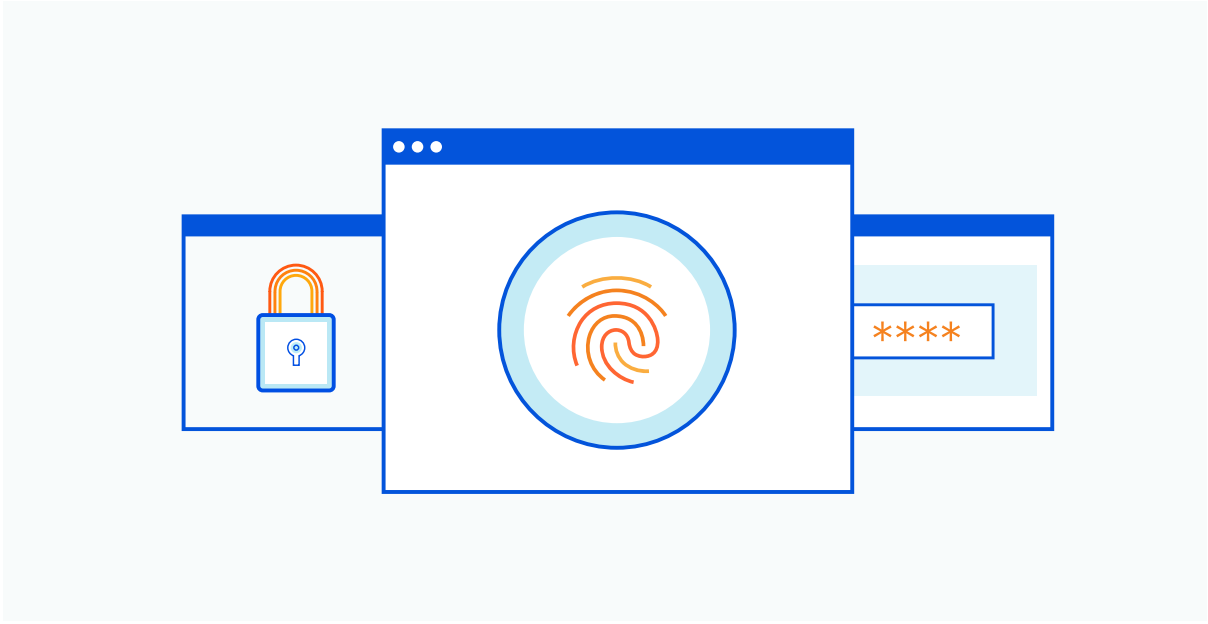
Il 61%

del traffico API aumentato anno dopo anno

Il Web programmabile[1] rileva che esistono più di 24.000 API conosciute e pubblicate. Si scopre tuttavia che la maggior parte delle API sono private e collegano tra loro le applicazioni interne. Le stime del numero di API private sono nell'ordine di milioni.

E quando diciamo che le API stanno guadagnando slancio, Cloudflare è un testimone in prima persona della loro crescita. Secondo i dati di Cloudflare Radar della prima metà del 2021, circa la metà del traffico sulla rete Cloudflare era correlato alle API. Inoltre, è aumentato del 61% dal 2020 al 2021.

Dato che espongono dati importanti, possiamo iniziare a vedere come rappresentano una nuova enorme superficie di attacco che dobbiamo proteggere. Come facciamo a saperlo? Ci sono stati molti attacchi importanti negli ultimi anni contro le API.



[1]https://www.programmableweb.com/apis/directory

Una superficie di attacco in espansione

Ora sappiamo che le API sono ovunque e sono fondamentali per il successo del business moderno. Utilizzano la logica dell'applicazione e possono condividere dati sensibili con altre applicazioni.

Si scopre tuttavia, senza sorpresa per nessuno, che gli autori di attacchi lo sanno e hanno tutte le intenzioni di sfruttare questa superficie di attacco in espansione nell'azienda.

Forse Gartner aveva ragione quando ha affermato[2] che entro il 2022 gli abusi delle API "sarebbero passati da un vettore di attacco raro a quello più frequente, con conseguenti violazioni dei dati per le applicazioni Web aziendali".

Resta da vedere se le API diventeranno il vettore di attacco più frequentemente preso di mira, ma è chiaro che continueranno a essere nel mirino degli aggressori.

Diciamo "continueranno" perché il mondo ha già assistito ad alcune violazioni di alto profilo per mano di una debole sicurezza delle API o dello sviluppo di API che non hanno tenuto conto della sicurezza.

T-Mobile è stata vittima di un attacco API nel 2017, quando 15 milioni di clienti che hanno acquistato nuovi dispositivi o hanno richiesto account T-Mobile hanno avuto le loro informazioni esposte. I dati includevano nomi, indirizzi, date di nascita, numeri di previdenza sociale, numeri di patente di guida e numeri di passaporto. Vice ha riferito che l'attacco è stato effettuato modificando il parametro del numero di telefono nella chiamata API. È possibile richiedere il numero di telefono di qualsiasi utente e l'API di T-Mobile invierà risposte inclusi i dati sensibili dell'account della persona il cui numero di telefono è stato richiesto.

Un'altra violazione relativa alle API ha colpito l'USPS quando un'API che supportava il monitoraggio dei pacchi in tempo reale è risultata priva di autorizzazione di base. Quando un utente ha effettuato l'accesso, può richiedere le informazioni sull'account di qualsiasi altro utente, tramite l'uso di parametri di ricerca con caratteri jolly che potrebbero estrarre tutti i record del set di dati. Ciò ha messo a rischio i dati di 60 milioni di titolari di conti USPS.

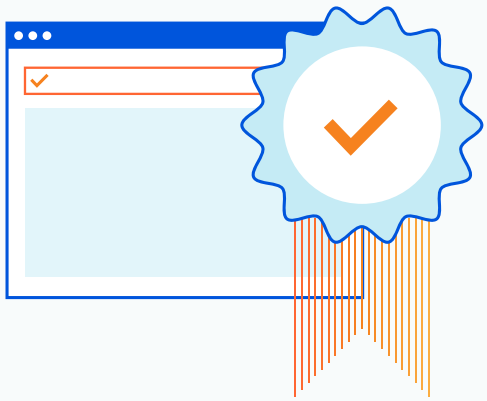
Nel 2019, JustDial, un importante motore di ricerca locale in India, ha effettivamente fatto trapelare tutti i dati dei clienti quando hanno lasciato esposte le versioni precedenti delle API che erano state sostituite da versioni più recenti. C'è di peggio: in effetti non c'era alcuna autenticazione in atto, il che significa che chiunque poteva chiamare le API ed estrarre i dati dal server di produzione. In altre parole, non erano necessarie tecniche avanzate per accedere ai dati degli utenti.

Anche Facebook, nonostante la sua leadership con GraphQL, ha subito numerose violazioni per mano delle sue API. Ad esempio, alla fine del 2019, Facebook ha raschiato un database, mettendo a rischio i nomi, il numero di telefono e gli ID utente di oltre 260 milioni di utenti.

C'è un lunghissimo elenco di organizzazioni che hanno avuto problemi a proteggere le API. E c'è un altrettanto lungo elenco di motivi per cui ciò si verifica. In primo luogo, le vulnerabilità dell'API sottostanti dovute a una progettazione non sicura aprono la porta agli attacchi. Inoltre, ad oggi, le organizzazioni non hanno avuto strumenti di sicurezza API first. Forse userebbero strumenti di sicurezza web come WAF o la limitazione della frequenza, ma quelli sono stati creati per proteggere le applicazioni, non le API. Questo può portare a sfide come falsi positivi e chiarisce la necessità di una sicurezza API progettata per il traffico che è in gran parte automatizzato.

[2]Fonte: Gartner: "API Security: What You Need to Do to Protect Your APIs", marzo 2021

Remix! Una nuova Top 10 delle API OWASP



Un aspetto positivo in tutto questo è che la OWASP Foundation, un'organizzazione da tempo focalizzata sul miglioramento della sicurezza delle applicazioni, è stata coinvolta. OWASP è nota per i suoi 10 principali rischi per la sicurezza delle applicazioni Web e ora ha pubblicato la Top 10 per la sicurezza delle API, un elenco dei principali rischi e vulnerabilità per la sicurezza delle API.

La verità è che ciò che da tempo ci preoccupa per la sicurezza delle applicazioni, vale anche per la creazione e la protezione delle API.

Per cominciare, qualsiasi organizzazione che sia API-first deve considerare la sicurezza in anticipo, mentre progetta le proprie API. Esaminiamo alcuni degli attacchi che abbiamo appena menzionato e il rischio per la sicurezza OWASP che hanno sfruttato.

Top 10 delle API OWASP

1. Autorizzazione interrotta a livello di oggetto
2. Autenticazione utente interrotta
3. Esposizione eccessiva dei dati
4. Mancanza di risorse e limitazione della frequenza
5. Autorizzazione interrotta a livello di funzione
6. Assegnazione di massa
7. Configurazione errata della sicurezza
8. Injection
9. Gestione non corretta delle risorse
10. Monitoraggio e registrazione insufficienti

Principali problemi della sicurezza delle API

1. Autenticazione e autorizzazione interrotte

Diamo un'occhiata più da vicino ad alcuni rischi chiave dell'API OWASP per gli attacchi sopra sfruttati, a partire dall'autenticazione e dall'autorizzazione.

JustDial ha ceduto all'autenticazione interrotta sui loro endpoint, che ha consentito a chiunque di effettuare chiamate. Quando viene implementata l'autenticazione, solo le chiamate API con il certificato TLS, le chiavi API, i token Web e così via corretti possono effettuare richieste, riducendo drasticamente il rischio per la sicurezza delle API.

Passando al numero uno nell'elenco OWASP, molti attacchi API sfruttano autorizzazioni deboli, interrotte o inesistenti, come abbiamo visto con USPS e T-Mobile. L'autorizzazione a livello di oggetto interrotta è comune e si verifica quando gli endpoint API vengono utilizzati sostituendo il numero ID di un oggetto a cui sono autorizzati ad accedere con l'ID di qualcosa a cui non sono autorizzati ad accedere. Spesso semplicemente modificando l'ID oggetto in una richiesta, ottengono l'accesso non autorizzato ai dati.

Il percorso dell'API e i parametri della query includono l'ID risorsa a cui si accede:

Chiamata autorizzata:

`GET api.greatsampleapis.com/v1/users/235` dove 235 è l'ID utente.

Le chiamate API manipolate possono ottenere l'accesso non autorizzato modificando 235 in 236, ovvero regolando l'identificatore dell'oggetto, in questo caso l'ID utente, per accedere ai dati per l'utente 236.

```
GET api.greatsampleapis.com/v1/users/236
```

Se non sono in atto controlli di autorizzazione, questa operazione può riuscire senza problemi. Gli sviluppatori dovrebbero modellare le minacce del proprio endpoint per assicurarsi che gli attacchi non possano regolare o modificare il valore dell'ID di un oggetto per ottenere l'accesso ad altri dati. L'uso di valori ID oggetto imprevedibili può anche aiutare in modo che non siano sequenziali e facili da indovinare.

2. Assegnazione di massa, esposizione dei dati e attacchi di iniezione

Un'altra classe di attacchi espone troppi dati nelle risposte o consente la modifica di oggetti interni con input.

Un'eccessiva esposizione dei dati si verifica quando un'API cerca di esporre ampiamente le proprietà dell'oggetto e restituisce troppi dati in una risposta, basandosi sui client che effettuano la richiesta di filtrare i dati.

Gli autori d'attacchi possono utilizzare dettagli aggiuntivi dalla risposta per creare un attacco ancora più potente o un'e-mail di phishing. Ad esempio, se una risposta restituisce tutti i dati seguenti, può essere utilizzata per e-mail di phishing molto convincenti:

```
{
“Id”: 213,
“FirstName”:”Sanjay”,
“LastName”:”Smythe”,
“EmailAddress”:”ssmythe@hacketyhack.com”,
“Occupation”: “Assistant to the Deputy Associate Vice Sub-undersecretary”,
“DOB”:”1986-05-21”,
“Bank”:”Easygo Financial”,
“AccountNumber”: 1362886306,
“PetName”: “Aloysius”,
 }
```

Gli attacchi di assegnazione di massa consentono alle chiamate API di alterare o sfruttare i valori interni quando le API espongono oggetti e variabili interni.

OWASP lo spiega così:

> "I framework software a volte consentono agli sviluppatori di associare automaticamente i parametri di richiesta HTTP a variabili o oggetti del codice del programma per semplificare l'utilizzo di quel framework per gli sviluppatori... A volte gli aggressori possono utilizzare questa metodologia per creare nuovi parametri che lo sviluppatore non ha mai voluto che a sua volta crea o sovrascrive nuova variabile o oggetti nel codice del programma che non era previsto."

Cosa devono fare gli sviluppatori? Dovrebbero comprendere i potenziali rischi quando fanno appello all'assegnazione di massa in fase di sviluppo ed evitare di esporre oggetti interni o variabili che possono essere utilizzati. Dovrebbero essere considerate anche le proprietà della lista consentita che possono essere aggiornate dai clienti.

Le applicazioni Web sono state a lungo soggette ad attacchi di iniezione e le API non sono così diverse. Dato quanto siano noti gli attacchi di iniezione, non ci soffermeremo su di essi, ma basti dire che gli input dovrebbero essere convalidati e sanificati prima di essere trasmessi. Dovrebbero essere compiuti sforzi per utilizzare la prevenzione della fuga di dati sulle risposte API e limitare il numero di record che possono essere restituiti per prevenire un incidente di divulgazione di massa.

3. Abuso di risorse e API shadow/rogue

Altri attacchi possono abusare delle API, quindi consumano quantità spropositate di risorse di calcolo, venendo sopraffatti, soccombendo a un attacco di tipo DoS. La porta è lasciata aperta a questi attacchi. Se non vengono posti limiti ad aspetti come il numero di richieste per client/risorsa, i record restituiti in un'unica risposta o la dimensione del payload della richiesta.

Come abbiamo visto negli attacchi JustDial, le API di produzione possono essere dimenticate, diventando così ombre o canaglia, dal momento che sono probabilmente non protette e possono essere sfruttate. Come nel resto della sicurezza, dobbiamo avere visibilità sul nostro patrimonio IT o sulla superficie di attacco per poi applicare i controlli di sicurezza appropriati. La visibilità nell'intera proprietà dell'endpoint API non è diversa.

Durante lo sviluppo delle API, i team dovrebbero disporre di un processo raffinato per tenere traccia delle versioni delle API per capire quali API sono in produzione e quali sono deprecate.

Considerazioni sulla protezione delle API

Abbiamo spiegato cosa sono le API e perché sono importanti e gli attacchi generali che prendono di mira le API. Ora esaminiamo come Cloudflare ha costruito la sicurezza delle API per proteggere le API dagli attacchi più comuni. Un'efficace sicurezza delle API deve tenere conto di tutto, dalla visibilità, ai modelli di sicurezza positivi, all'arresto degli abusi, alla protezione dei dati.

Cloudflare API Shield

Fondamento di visibilità

Visibilità

Come con altri aspetti della sicurezza, per applicare una protezione è necessario che si verifichi qualcosa. Le API non sono diverse, in particolare quando le aziende hanno centinaia o addirittura migliaia di endpoint API.

L'individuazione e la visibilità delle API è un aspetto fondamentale per il governo delle API, in modo che le organizzazioni abbiano sempre un'istantanea chiara della proprietà degli endpoint API per evitare che le API shadow o rogue diventino un problema.

Come abbiamo visto con JustDial, se le organizzazioni perdono traccia delle API, ciò può portare a violazioni dei dati.

Difesa delle API in profondità

Protezioni L7 delle API

Abbiamo implementato da tempo Web Application Firewall per proteggere le applicazioni dagli attacchi DDoS di livello 7. La protezione API dovrebbe iniziare con molti di questi controlli affidabili come la limitazione della frequenza e la protezione da attacchi DDoS per scongiurare attacchi di negazione dei servizi e tentativi di accesso con forza bruta e abusi generali effettuati da indirizzi IP specifici. Ciò applicherà i limiti di utilizzo dell'API e garantirà la disponibilità combattendo l'API 4 OWASP, Mancanza di risorse e limitazione della frequenza.

Le regole WAF dovrebbero essere utilizzate anche per identificare e bloccare gli attacchi comuni che prendono di mira le API.

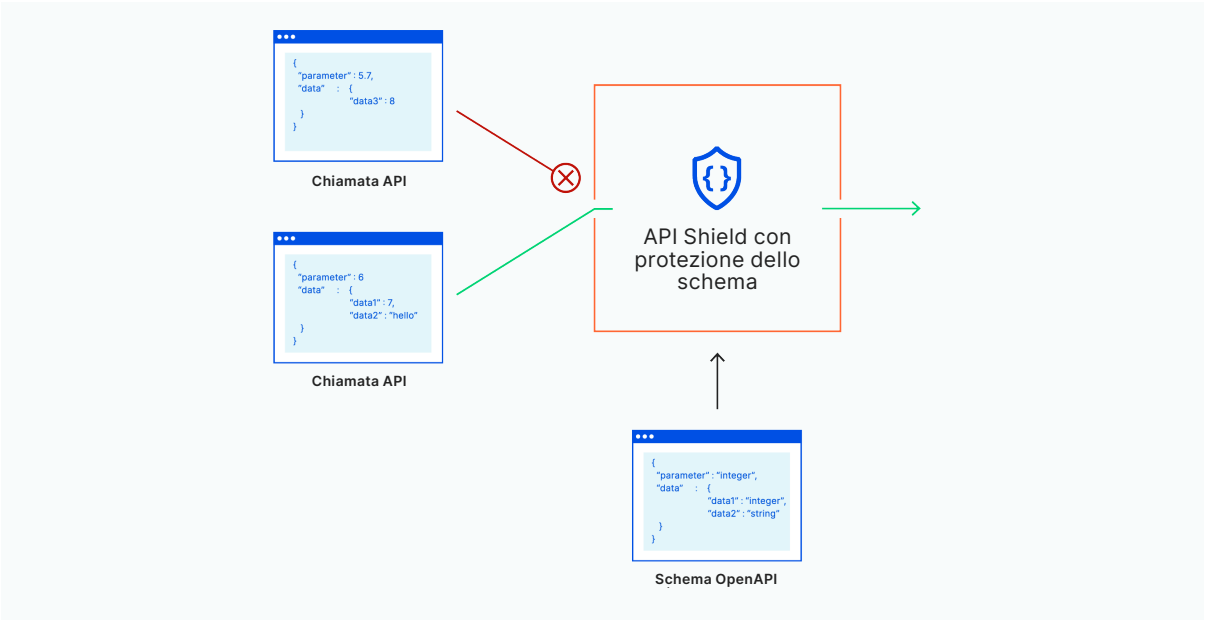
Autenticazione e autorizzazione

Autenticazione MTL

Abbiamo visto negli attacchi alle API che abbiamo evidenziato che la mancanza di autenticazione può essere devastante. L'autenticazione deve essere integrata fin dall'inizio e dovrebbe essere rafforzata con TLS reciproco per applicare l'identità basata su certificato per casi d'uso come dispositivi mobili o IoT. Questo approccio è un modello di lista consentita più positivo che consente la connessione solo alle richieste provenienti da client legittimi
con certificati validi.

Verifiche delle credenziali esposte

Le API non sono immuni dagli attacchi di credential stuffing che passano in rassegna i tentativi di accesso utilizzando credenziali rubate. Queste credenziali dell'account potrebbero essere compromesse da violazioni di terze parti fuori dal controllo di un'organizzazione. Come parte dei controlli di autenticazione, la sicurezza dell'API dovrebbe essere in grado di scansionare le credenziali di autenticazione all'accesso, rispetto a un database di credenziali trapelate. Se le credenziali sembrano essere compromesse, la sicurezza dell'API dovrebbe attivare misure di sicurezza aggiuntive come la reimpostazione della password o l'autenticazione a più fattori e, naturalmente, bloccare il tentativo.



La convalida dello schema valuta ogni richiesta rispetto a una registrazione dello schema API o alle richieste di blocco che non lo rispettano.

Protezione positiva delle API

Convalida dello schema delle API

Gli sviluppatori fanno di tutto per creare uno schema API, che è la documentazione, o regole di base, per come si aspettano che gli altri interagiscano con l'API. Questo può stabilire cose come i metodi di richiesta e le operazioni su ciascun endpoint (GET /users, POST /users) o parametri di input e output per ogni operazione. OpenAPI v3, comunemente noto anche come standard Swagger, è lo schema più noto per la definizione delle API.

Una sicurezza API affidabile dovrebbe utilizzare un modello positivo, zero trust, che si applica allo schema.

Con uno schema in atto, le richieste dovrebbero essere convalidate automaticamente rispetto ad esso. Tutte le operazioni API sono bloccate, ad eccezione di quelle che sono state convalidate come conformi allo schema.