

Bonus: Esplorazione dei CVE per macOS

Preparazione Ambiente e Introduzione

Utilizzeremo l'intelligenza artificiale in modo da reperire le CVE ovvero le vulnerabilità comuni del target, individuato nel sistema operativo MAC OS.

Prompt fornito

sono uno studente di cybersecurity sto studiando le CSE
dei vari sistemi operativi in particolare di mac os per
proteggermi da quest'ultime, potresti aiutarmi a fare un
elenco di questi e spiegarmi alcune dettagliatamente
inclusi i dettagli delle vulnerabilità e le soluzioni
consigliate

Risposta

◆ Ciao! È un piacere aiutarti. Come studente di cybersecurity, approcciarsi a macOS è molto interessante perché, nonostante la fama di sistema "impermeabile", Apple deve gestire una superficie di attacco complessa che va dai driver del kernel alle applicazioni sandboxate.

Nello slang tecnico e accademico, quando parliamo di **CSE** (Common Security Exploits) o vulnerabilità note, ci riferiamo spesso alle falliche catalogate come **CVE** (Common Vulnerabilities and Exposures). Di seguito ho preparato un elenco delle principali categorie di attacco su macOS aggiornato al 2025/2026, con i relativi ragionamenti tecnici.

1. Bypass di SIP (System Integrity Protection)

Cos'è: SIP (chiamato anche "rootless") è una tecnologia che impedisce anche all'utente `root` di modificare file di sistema cruciali (come `/System` o `/usr/bin`).

- **Vulnerabilità** (es. **CVE-2024-44243**): Recentemente sono state scoperte falliche in daemon di sistema come `storagekitd`. Se un attaccante ottiene privilegi di root, può sfruttare un errore nella logica di gestione del file system per bypassare SIP senza riavviare in modalità Recovery.
- **Dettagli:** L'attaccante può installare `rootkit` o driver del kernel malevoli che diventano invisibili al sistema operativo.
- **Soluzione:** - Mantenere sempre aggiornato macOS (queste falliche vengono patchate rapidamente).
 - Monitorare l'integrità del sistema tramite il comando `csrutil status` (anche se un exploit avanzato potrebbe falsificare l'output).

2. Exploits nel framework ImageIO (Memory Corruption)

Cos'è: ImageIO è il componente che macOS usa per leggere e renderizzare le immagini (JPEG, PNG, ecc.).

- **Vulnerabilità** (es. **CVE-2025-43300**): Si tratta di un errore di **Out-of-Bounds Write**. Elaborando un'immagine malevola appositamente creata, un utente può causare una corruzione della memoria che porta all'esecuzione di codice arbitrario (RCE).
- **Dettagli:** Poiché molte app (Safari, Mail, Messaggi) usano ImageIO per generare anteprime, l'attacco può essere "zero-click" (basta che l'immagine venga visualizzata).
- **Soluzione:** - Non aprire file multimediali da fonti non attendibili.
 - Utilizzare strumenti di analisi dinamica per monitorare crash insoliti di processi legati alle immagini.

Dunque notiamo come la prima tra le vulnerabilità è una falla legata al bypass del SIP. Questo sistema operativo è noto per la sua sicurezza proprio perché non c'è possibilità di agire su file di sistema nemmeno da parte dell'utente root. La falla trovata consiste nello sfruttare un errore di logica di gestione del file system (una volta che l'attaccante accede ai permessi root). La soluzione è ovviamente tenere il sistema operativo sempre aggiornato in quanto vista la criticità del problema vengono rilasciate molto frequentemente delle

patch per rimediare alla vulnerabilità.

3. Bypass di TCC (Transparency, Consent, and Control)

Cos'è: TCC è il database che gestisce i permessi delle app (accesso alla fotocamera, microfono, cartella Documenti).

- Vulnerabilità:** Esistono tecniche di **Clickjacking** dove un'app malevola sovrappone una finestra invisibile sopra la richiesta di permesso di macOS, inducendo l'utente a cliccare "Consenti" senza saperlo. Altre falle (come CVE-2025-43286 nel componente `SharedFileList`) permettono alle app di uscire dalla propria "sandbox" e accedere ai dati protetti.
- Dettagli:** L'attaccante mira a rubare i database `TCC.db` per leggere cronologie chat o file sensibili.
- Soluzione:** - Verificare periodicamente in *Impostazioni di Sistema > Privacy e Sicurezza* quali app hanno accesso ai dati.
 - Utilizzare software come *LuLu* o *Little Snitch* per monitorare connessioni di rete in uscita non autorizzate da app "insospettabili".

4. Vulnerabilità WebKit (Safari)

Cos'è: WebKit è il motore di rendering di Safari e di quasi tutte le visualizzazioni web su macOS.

- Vulnerabilità (es. CVE-2025-43529):** Un errore di **Use-After-Free** nel motore JavaScript. Un sito web compromesso può mandare in crash il processo e iniettare codice nel sistema.
- Dettagli:** Spesso queste falle vengono concatenate (chained) con un bypass della sandbox per ottenere il controllo totale del Mac partendo da una semplice navigazione web.
- Soluzione:** - Attivare la **Modalità di isolamento (Lockdown Mode)** se sei un bersaglio ad alto rischio (riduce drasticamente la superficie d'attacco di WebKit).
 - Usare browser con architetture multi-processo isolate e tenerli costantemente aggiornati.

Tabella Riassuntiva per lo Studio

Categoria	Vettore di Attacco	Conseguenza	Difesa Principale
Kernel/SIP	Driver corrotti / Exploit Daemon	Controllo totale (Rootkit)	Aggiornamenti OS
Media (ImageIO)	Immagini malevole (Mail/Web)	Esecuzione Codice (RCE)	Patching rapido
Sandbox/TCC	App di terze parti	Furto Dati Privati	Controllo Permessi / LuLu
Browser (WebKit)	Siti web compromessi	Memory Corruption	Lockdown Mode / Safari Updates

 Esporta in Fogli



Conclusione

Possiamo concludere che macOS si basa molto sulla firma del codice e alla notarizzazione. Gli exploit moderni infatti puntano molto ad aggirare questi controlli o a sfruttare bug logici presenti in processi che Apple considera fidati.

Seguono dunque le varie spiegazioni di vari exploit recenti a cui l'intelligenza artificiale trova soluzione.