Raport consolidat eveniment cibernetic

Vulnerabilități Critice în Node.js

Node.js este o platformă runtime extrem de populară pentru dezvoltarea aplicațiilor backend. Este utilizată de companii mari și mici pentru a dezvolta API-uri, aplicații web și microservicii datorită vitezei și scalabilității sale. Această popularitate sporită o face o țintă atractivă pentru atacatorii cibernetici. Modulele Node.js descărcate din registrul NPM sunt frecvent expuse la vulnerabilități datorită naturii lor open-source și utilizării pe scară largă.

Informații suplimentare

Versiunea afectată		Vulnerabilitățile afectează versiunile *Node.js mai vechi de 16.20.1 și 18.16.1*, precum și modulele abandonate sau fără suport activ.
Vulnerabilități	**CVE-2025-23087**	❖ **Tip:** Escaladarea Privilegiilor ❖ **Descriere:** Această vulnerabilitate permite atacatorilor să ocolească mecanismele de securitate și să obțină acces privilegiat la resursele interne. ❖ **Impact:** Atacatorii pot accesa date sensibile și pot compromite funcționalitățile critice ale sistemului. ❖ **Metoda de exploatare:** Exploatarea unui bug în gestionarea sesiunilor pentru a obține acces administrativ.
	CVE-2025-23088	❖ **Tip**: Remote Code Execution (RCE) ❖ **Descriere**: Permite atacatorilor să execute cod arbitrar în aplicațiile vulnerabile folosind inputuri nefiltrate. ❖ **Impact:** Execuția de cod malițios poate duce la compromiterea completă a sistemului. ❖ **Metoda de exploatare:** Trimiterea unui payload special creat prin endpoint-uri API expuse.
	CVE-2025-23089	❖ **Tip:** Escaladarea Privilegiilor ❖ **Descriere:** Exploatarea mecanismului de autentificare pentru a obține acces la nivel de root. ❖ **Impact:** Atacatorii pot controla întregul sistem și pot accesa resurse protejate.

		❖ **Metoda de exploatare:** Trimiterea unor cereri modificate pentru a manipula tokenurile de autentificare.
	CVE-2025-23083	❖ **Tip**: Denial of Service (DoS) ❖ **Descriere:** Exploatarea unei vulnerabilități în gestionarea worker-ilor duce la blocarea serverului. ❖ **Impact:** Serverul devine indisponibil, afectând serviciile. ❖ **Metoda de exploatare**: Trimiterea unui volum mare de cereri care declanșează condiții de blocaj.
	CVE-2025-23084	❖ **Tip**: Ocolirea Restricțiilor de Acces ❖ **Descriere**: Permite atacatorilor să acceseze directoare protejate folosind căi neautorizate pe sistemele Windows. ❖ **Impact:** Expune fișiere sensibile și configurări critice. ❖ **Metoda de exploatare:** Utilizarea unui nume de unitate manipulat pentru a accesa directoare neprotejate.
	CVE-2025-23085	❖ **Tip:** Denial of Service (DoS) ❖ **Descriere:** Problema de scurgere de memorie afectează disponibilitatea resurselor sistemului. ❖ **Impact:** Poate cauza blocaje și degradarea performanței serverului. ❖ **Metoda de exploatare:** Închiderea necorespunzătoare a conexiunilor socket pentru a induce scurgeri de memorie.

Execuția Codului Arbitrar RCE		Atacatorii pot exploata modulele Node.js pentru a executa cod malițios, obținând control complet asupra sistemului vulnerabil.
		Exemplu de cod RCE**:**
		„const http = require('http');
		const vm = require('vm');
		http.createServer((req, res) => {
		const userInput = req.url.slice(1); // Exemplu de intrare nesecurizată
		vm.runInNewContext(userInput); // Execuţie directă a intrării utilizatorului
		res.end('Cod executat!');
		}).listen(8080);”

Escaladarea privilegiilor		Vulnerabilități în funcțiile interne ale Node.js permit atacatorilor să obțină acces la resurse sau date care ar trebui să fie protejate de mecanisme de securitate. Cum ar fi: ❖ **Date Confidențiale:** ✓ Datele utilizatorilor, precum acreditive sau informații personale. ✓ Chei API utilizate pentru accesarea altor servicii integrate. ❖ **Sistemele de Fișiere:** ✓ Fișierele de configurare (e.g., .env, config.json). ✓ Codul sursă al aplicației sau fișiere de șabloane. ❖ **Resurse Externe:** ✓ Acces la API-uri terțe, care pot fi exploatate pentru atacuri suplimentare.

Modulele de exploatare	Anumite module pot fi exploatate pentru a suprasolicita serverele prin trimiterea unui volum mare de cereri special concepute, cauzând indisponibilitatea serviciilor.
	❖ ***vm:*** Permite execuția de cod arbitrar într-un sandbox, dar poate fi exploatat dacă inputurile utilizatorilor nu sunt validate.
	❖ ***child_process***: Poate fi utilizat pentru a executa comenzi de sistem, fiind extrem de periculos dacă este manipulat incorect.
	
	Exemplu:
	„const { exec } = require('child_process');
	const userCommand = 'rm -rf /'; // Exemplu de input maliţios
	exec(userCommand, (error, stdout, stderr) => {
	console.log(stdout);
	});”
	
	❖ ***http:*** Dacă sunt acceptate cereri nefiltrate, atacatorii pot trimite payload-uri periculoase pentru a exploata vulnerabilități în alte module conectate.
	❖ ***Module NPM Nesecurizate:*** Module abandonate sau cu vulnerabilități cunoscute pot fi exploatate pentru a compromite aplicațiile.

Impact asupra sistemelor	Exploatarea vulnerabilităților identificate poate conduce la: ❖ ***Pierderea confidențialității:*** Datele utilizatorilor pot fi compromise. ❖ ***Încetinirea sau blocarea aplicațiilor:*** Atacurile de tip DoS pot paraliza aplicațiile. ❖ ***Compromiterea infrastructurii:*** Atacatorii pot folosi sistemele compromise pentru activități malițioase, inclusiv instalarea de malware sau furtul de informații.
Măsuri de mitigare	❖ ***Actualizarea Node.js:*** Este esențial să utilizați cea mai recentă versiune disponibilă, care conține patch-uri pentru vulnerabilități. ❖ ***Scanarea Dependențelor:*** Folosiți instrumente precum npm audit sau Snyk pentru a identifica și remedia vulnerabilitățile din pachetele utilizate. ❖ ***Securizarea Aplicațiilor:*** ✓ Utilizați module precum helmet.js pentru a proteja aplicațiile web. ✓ Implementați politici stricte de Content Security Policy (CSP). ❖ ***Izolarea Aplicațiilor:*** Folosiți containere Docker și sandbox-uri pentru a limita impactul exploatărilor. ❖ ***Monitorizarea Sistemelor:*** Implementați soluții de monitorizare a activității aplicațiilor pentru a detecta atacuri sau comportamente anormale.

Resurse externe

https://cyble.com/blog/critical-vulnerabilities-in-node-js-expose-systems/