PRUEBA DE CONCEPTO (POC)

A06 - COMPONENTES VULNERABLES Y DESACTUALIZADOS DEL OWASP TOP TEN

Braulio Alberto Perez Cano
Israel Alejandro Cel Alcocer

¿Qué es el riesgo A06? Componentes vulnerables y desactualizados

Este riesgo del OWASP Top 10 se refiere a cuando una aplicación usa piezas de software (librerías, módulos, frameworks) que:

- Están viejas
- Tienen errores o fallas conocidas
- No se han actualizado

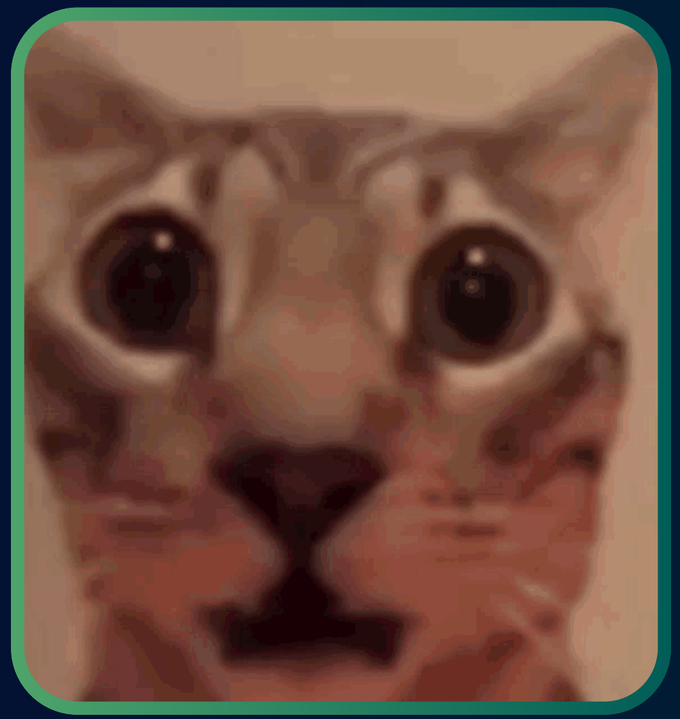
¿Qué son esas "piezas de software"?

Cuando los desarrolladores hacen una app, no programan todo desde cero. Usan "piezas listas" hechas por otros (por ejemplo: express, lodash, jsonwebtoken, etc.).

Esas piezas se llaman:

- Librerías
- Dependencias
- Componentes de terceros

Y son como los ladrillos con los que se construye la app.



¿Dónde está el peligro?

Si usas una versión antigua de un componente que tiene una vulnerabilidad conocida, un atacante puede buscar esa falla y explotarla para:

- **Robar datos**
- **Tomar control de cuentas**
- **Acceder al sistema sin permiso**

juice-shop

Public

Sponsor | Watch 165 | Fork 12.2k | Star 11k

master | 4 Branches | 215 Tags | Go to file t | Add file | Code

bkimminich Add AppSec Barcelona 2025 demo sessions ✓ 62816da · 5 days ago 20,474 Commits

.dependabot	Explicitly ignore JWT library version	5 years ago
.github	Upgrade action to prevent file upload failures for release bui...	2 weeks ago
.gitlab	Change liveness (or readiness) probe to port 3000	4 years ago
.well-known	chore: fix CSAF sorting, language to lower, and version	11 months ago
.zap	Ignore all Sec-Fetch-* header warnings	2 years ago
config	Add new "Memory Bomb" challenge	last month
data	Merge pull request #2507 from juice-shop/l10n_develop	2 weeks ago
encryptionkeys	use a 16byte IV	8 years ago
frontend	Bump to v17.2.0	2 weeks ago
ftp	Fix password of support team to match the actually defined ...	3 years ago
i18n	Prevent accidental i18n overwrites during runtime	6 years ago
lib	Add "Memory Bomb" to coupled file upload challenges	3 weeks ago

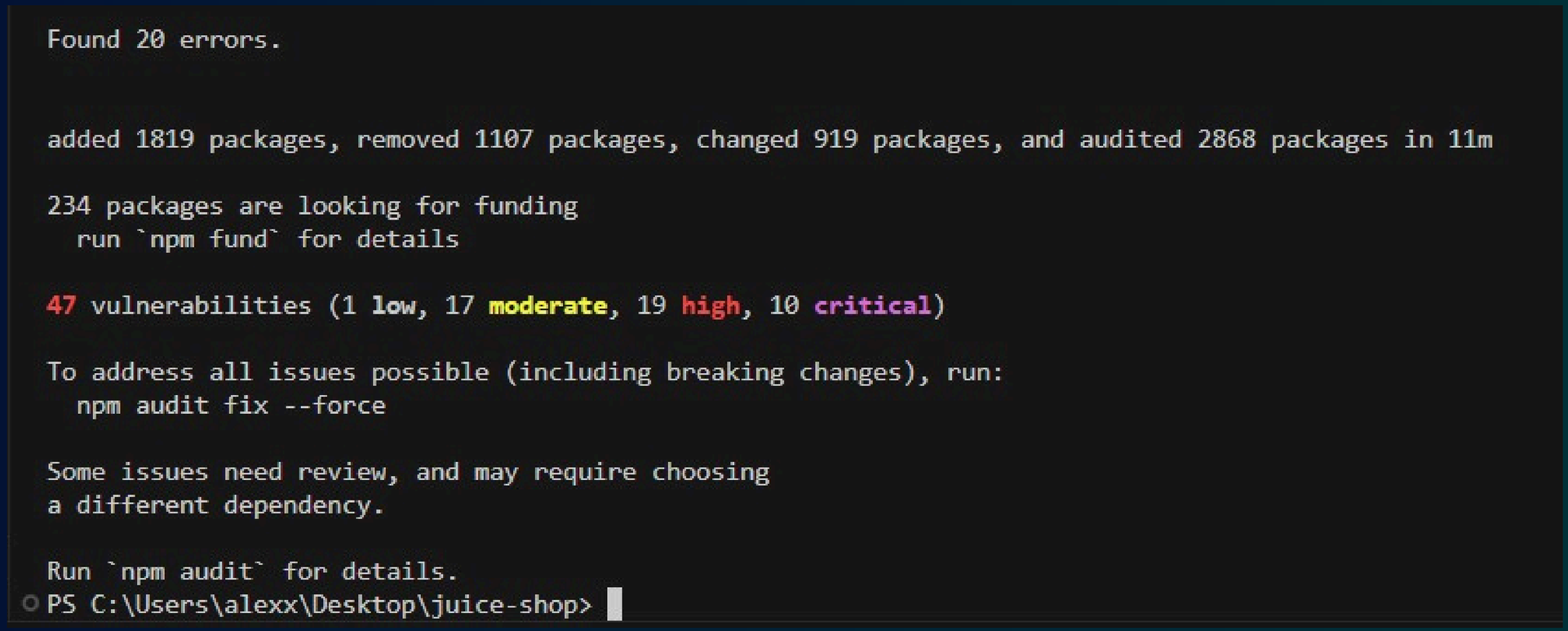
About

OWASP Juice Shop: Probably the most modern and sophisticated insecure web application

owasp-juice.shop

javascript security hacking owasp application-security pentesting ctf vulnerable appsec hacktoberfest owasp-top-10 owasp-top-ten 24pullrequests vulnapp

Readme
MIT license
Code of conduct
Security policy
Activity
Custom properties
11k stars
165 watching
12.2k forks

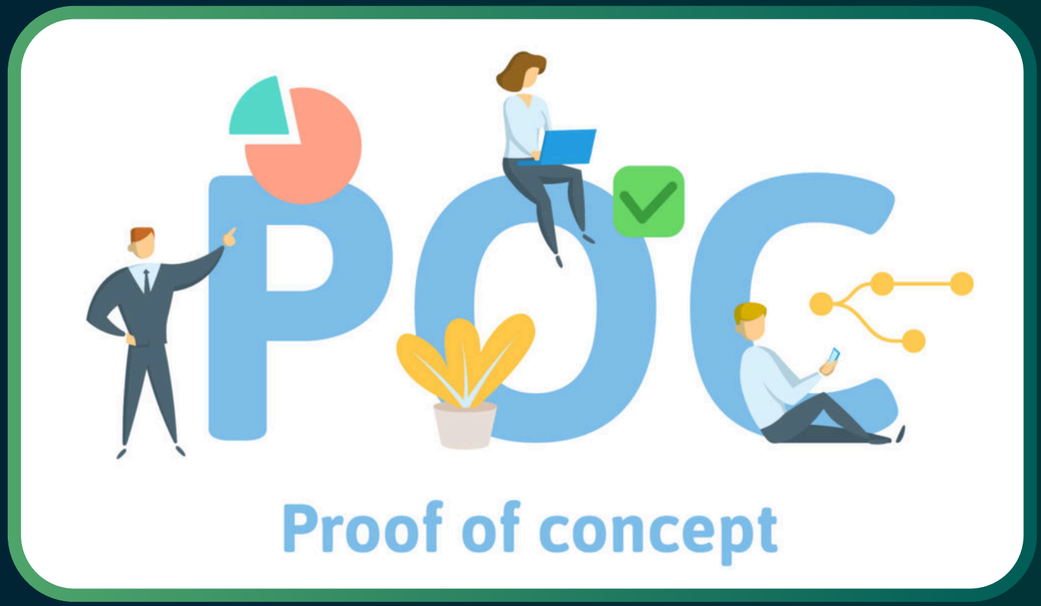


Found 20 errors:
Esto indica problemas en el proyecto, como conflictos de dependencias o configuraciones incorrectas que podrían comprometer su seguridad y funcionalidad.

47 vulnerabilities (1 low, 17 moderate, 19 high, 10 critical)":
Las vulnerabilidades críticas y altas son especialmente peligrosas porque un atacante podría explotarlas para comprometer la aplicación, robar datos o tomar control del sistema.

Poc
(Proof con concept)

Una PoC (Proof of Concept en ciberseguridad no necesita ser un ataque real:
solo necesita demostrar que existe una vulnerabilidad que podría ser explotada.



El PoC nos demuestra que

- Se instalo una aplicación (**Juice Shop**).
- Se usaron herramientas de análisis (npm audit) para inspeccionar sus componentes (librerías).
- Se encontraron que muchas de esas librerías están desactualizadas y tienen vulnerabilidades conocidas:

¡10 críticas!

  - Con nombres y CVEs asociados (npm audit los muestra).
  - Eso demuestra que la aplicación es vulnerable, incluso sin tocar el código.

Resultado final:

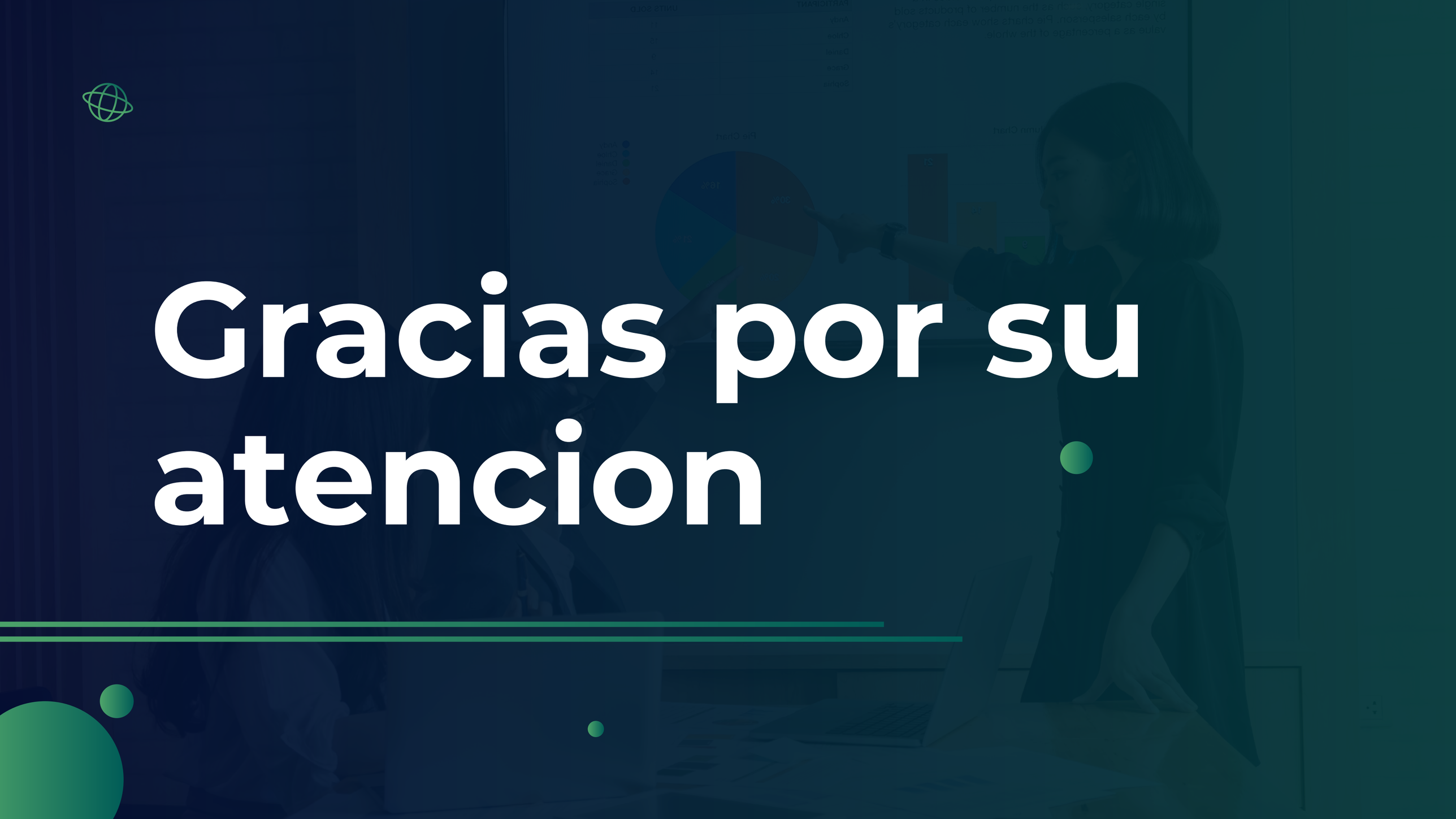
La POC es válida

se uso un entorno seguro (Goat: Juice Shop)

se detectaron vulnerabilidades reales

se mostraron que esas vulnerabilidades podrían explotarse

Todo se hizo con herramientas legítimas, sin dañar nada



Gracias por su atencion