

Aprende Análisis de vulnerabilidades con OWASP Juice Shop

OWASP Capítulo Ciudad de México

04 de noviembre, 2023



OWASP FOUNDATION

Agenda

1. ¿Qué es OWASP?
 - ¿Qué son los capítulos?
 - ¿Quiénes conformamos el Capítulo CDMX?
 - ¿Qué son los proyectos OWASP?
2. Lecturas nocturnas amenas introductorias.
3. ¿Qué es una vulnerabilidad web?
 - Análisis de vulnerabilidades
4. Manos a la obra
 - ¿Qué es Juice Shop?
 - Instalación
 - Retos

¿Qué es OWASP?

- ❖ **O**pen **W**orldwide **A**pplication **S**ecurity **P**roject.
- ❖ Fundación sin fines de lucro cuyo objetivo es mejorar la **seguridad** del **software**. Arrancó el 1ro de Diciembre, 2001.
- ❖ ¿Cómo logra su objetivo?
Mediante sus **proyectos** de software de **código abierto** liderados por su **comunidad**, sus miembros, sus **Capítulos** y **eventos** locales e internacionales.
- ❖ Todos los Proyectos, Documentos, Herramientas, Foros Y Capítulos son **gratuitos** y abiertos a todo interesado en fortalecer la seguridad de aplicaciones.



¿Qué son los Capítulos?

- Los Capítulos locales buscan conformar comunidad de profesionales de seguridad informática, a través de eventos y reuniones alrededor del **mundo**.
- Dirigidos por líderes locales conforme a políticas bien establecidas – [Política de los capítulos](#).
- Existe alrededor de 300 Capítulos.
- En México hay 4: Aguascalientes, Querétaro, Rivera Maya y Ciudad de México.

¿Quiénes conformamos el Capítulo CDMX?

- Página principal del Capítulo:
<https://owasp.org/www-chapter-mexico-city/>
- Actualmente, está conformado por 5 líderes.
- Reactivó actividades en julio del 2022.
- Redes sociales disponibles: **owasp_cdmx**



¿Qué son los Proyectos OWASP?

- ❑ Los proyectos OWASP son de **código abierto** y contruidos por **miembros de la comunidad de voluntarios**.
- ❑ Existen proyectos de **herramientas** (Code Projects) y de **documentación** (Documentation Projects).
- ❑ Actualmente, el inventario tiene 302 proyectos.
- ❑ Para mayor detalle de todos los poryectos OWASP, visitar <https://owasp.org/projects/>

Lecturas nocturnas amenas introductorias

- **Web Security Testing Guide** (465pp) – [Enlace](#).

Guía de cómo aplicar la Metodología de Revisión de OWASP. Basado en el enfoque de caja negra. Lista de pruebas de seguridad – método de evaluación de la seguridad mediante la validación y verificación metodológica de los controles de seguridad. Incluye reporte de hallazgos.

- **Application Security Verification Standard** (74 pp). – [Enlace](#).

Marco de requisitos y controles de seguridad requeridos al diseñar, desarrollar y probar aplicaciones web y servicios web modernos. Define tres niveles de verificación de seguridad. Verificable y revisable.

Lecturas nocturnas amenas introductorias

- **Automated Threats to Web Applications** (80pp) – [Enlace](#).
Estándar de facto de la industria de detección y mitigación de amenazas automatizadas web (Escaner de vulnerabilidades).
- **OWASP Top 10:2021** (aprox. 25pp, 2017) – [Enlace](#).
Lista los 10 principales riesgos de seguridad en aplicaciones web. Abarca las vulnerabilidades más comunes asociadas a los riesgos, medidas de prevención, escenarios de ataque de ejemplo, referencias. Principalmente para concientización.
- **OWASP API Security Top 10:2023** (aprox. 31pp, 2019) – [Enlace](#).
Application Programming Interface (API). Estructura similar al OWASP Top 10. ¿Cómo saber si el API es vulnerable al riesgo?. Principalmente para concientización.

Lecturas nocturnas amenas introductorias

- **OWASP Top 10 Privacy Risks:2021** – [Enlace](#).

Enfocado a riesgos de privacidad en aplicaciones web y sus controles. Provee información de como implementar privacidad por diseño. ¿Como revisar si la aplicación web es susceptible al riesgo?

- **OWASP Top 10 Proactive Controls:2018** (40pp) – [Enlace](#).

Describe el top 10 de controles de seguridad que los desarrolladores deben incluir al desarrollar. Incluye descripción, mejores prácticas de implementación y las vulnerabilidades prevenidas.

- Y más.

Lecturas nocturnas amenas introductorias

- [OWASP Cloud-Native Application Security Top 10](#)
- [OWASP Desktop App Security Top 10](#)
- [OWASP Docker Top 10](#)
- [OWASP Low-Code/No-Code Top 10](#)
- [OWASP Machine Learning Security Top Ten](#)
- [OWASP Mobile Top 10](#)
- [OWASP TOP 10](#)
- [OWASP Top 10 CI/CD Security Risks](#)
- [OWASP Top 10 Client-Side Security Risks](#)
- [OWASP Top 10 Privacy Risks](#)
- [OWASP Serverless Top 10](#)

¿Qué es vulnerabilidad web?

- ❑ Falla, debilidad, flag (bandera).
- ❑ ¿Dónde puede haber vulnerabilidades? En el diseño de un sistema, en su implementación, en su operación o en su administración.*
- ❑ ¿Qué se puede hacer con ella? Explotarla para **comprometer** los objetivos de seguridad del sistema.*

* Fuente: Web Security Testing Guide v4.2

Análisis de vulnerabilidades

- ❑ Es la **identificación** y **validación** de vulnerabilidades*.
- ❑ Se emplea para identificar y evaluar los riesgos de seguridad que pudiera haber debido a las vulnerabilidades identificadas*.

1. Identificación.

- Herramientas automatizadas que buscan e identifican componentes, codificaciones vulnerables conocidas.
- De forma manual.
- Mediante pruebas de seguridad - Método de evaluación de la seguridad mediante la validación y verificación metodológica de los controles de seguridad**.

2. Validación.

- Para reducir las vulnerabilidades identificadas a solo las válidas.
- Verificando que la vulnerabilidad es explotable.

Fuente: Penetration Testing Execution Standard (PTES)

** Fuente: OWASP Web Security Testing Guide.

Manos a la obra

Requisitos mínimos: Contar con NodeJS 20.5 o superior y Docker Desktop 4.22 o superior, instalados.

¿Qué es Juice Shop?

- ❑ Aplicación web intencionalmente vulnerable.
- ❑ Es lo opuesto a las “mejores prácticas” de seguridad para los desarrolladores web.
- ❑ Más de 105 retos, diferentes niveles de dificultad.
- ❑ Puede usarse como target de herramientas de seguridad, para CTFs.
- ❑ Abarca diversos riesgos o tipos de vulnerabilidades*.
- ❑ Documento detallado en formato PDF (422pp) u online – [Enlace](#).



Instalación de Juice Shop con NodeJS 20.5

Ver el Anexo A

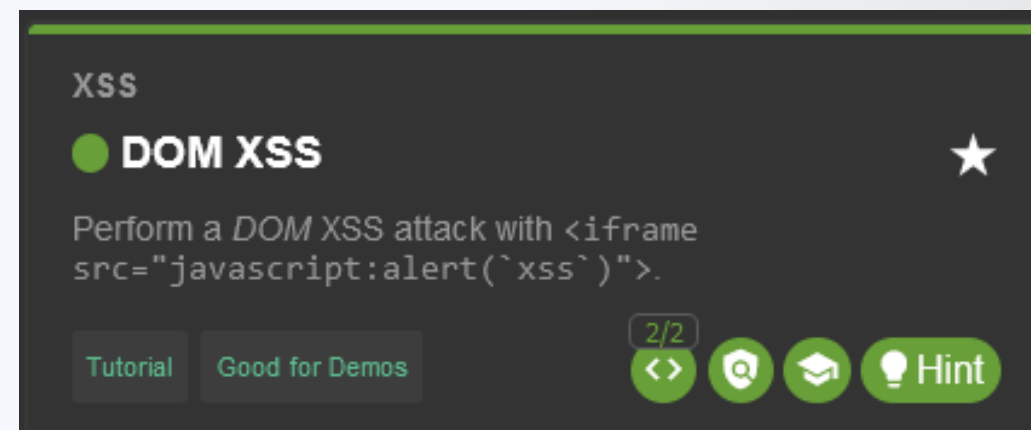
Instalación de Juice Shop con Docker Desktop 4.22

Ver el Anexo A

Reto 1: DOM XSS

1. ¿Qué es un ataque de DOM XSS?
2. ¿Qué es un <iframe>?
3. Recursos disponibles para el reto:
 - Pistas del reto.
 - Tutorial del reto.
 - Código fuente vulnerable.
 - Info adicional de mitigación.

¡Manos a la obra!



Fuentes de información adicionales:

- OWASP Web Security Testing Guide: Testing for DOM-Based Cross Site Scripting.
- <https://owasp.org/www-community/attacks/>

Reto 1: DOM XSS. Reto de código.

Find It

Fix It 

```
1 filterTable () {
2   let queryParam: string = this.route.snapshot.queryParams.q
3   if (queryParam) {
4     queryParam = queryParam.trim()
5     this.dataSource.filter = queryParam.toLowerCase()
6     this.searchValue = this.sanitizer.bypassSecurityTrustHtml(queryParam)
7     this.gridDataSource.subscribe((result: any) => {
8       if (result.length === 0) {
9         this.emptyState = true
10      } else {
11        this.emptyState = false
12      }
13    })
14  } else {
15    this.dataSource.filter = ''
16    this.searchValue = undefined
17    this.emptyState = false
18  }
19 }
```

Reto 1: DOM XSS. Reto de código.

funcionCuandoBusca:

`queryParam` = parámetro "q" de barra de búsqueda

Si `queryParam` es válido:

Quita espacios vacíos a `queryParam`

Establece un filtro = `queryParam` a minúsculas

Establece `searchValue` = Bypass de seguridad y confía en `queryParam`

Valida si hubo resultados:

Muestra resultados

Si no hubo:

No muestra resultados

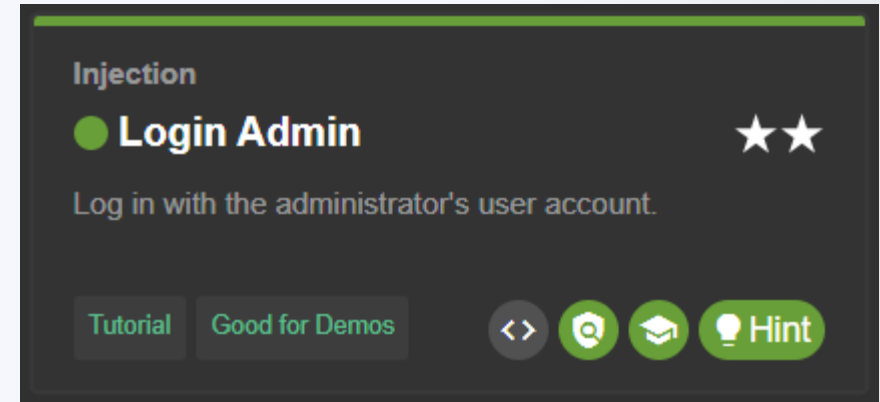
Si no:

Establece un filtro = ""

`searchValue` = undefined

Reto 2: SQL Injection

1. ¿Qué es un ataque de SQL injection?
2. ¿Cómo identificar que se puede realizar SQLi?
3. Recursos disponibles para el reto:
 - Pistas del reto.
 - Tutorial del reto.



```
SELECT * FROM users WHERE username = 'wiener' AND password = 'bluecheese'
```

ejemplo de consulta SQL

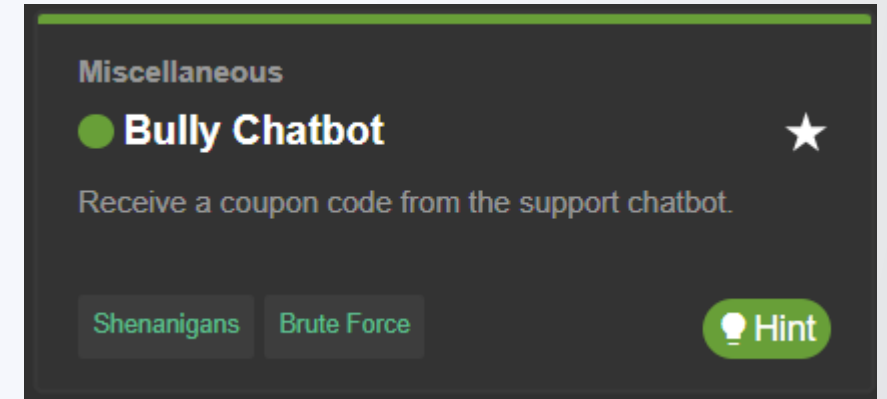
¡Manos a la obra!

Fuentes de información adicionales:

- OWASP Web Security Testing Guide: <https://owasp.org/www-project-web-security-testing-guide/>.
- https://owasp.org/www-community/attacks/SQL_Injection

Reto 3: Bully Chatbot

1. ¿Qué es un Chatbot?
2. Recursos disponibles para el reto:
 - Burp suite
 - FoxyProxy



¡Manos a la obra!

Fuentes de información adicionales:

– OWASP Web Security Testing Guide: Authentication Cheat Sheet.

https://cheatsheetseries.owasp.org/cheatsheets/Authentication_Cheat_Sheet.html#protect-against-automated-attacks

¿Preguntas?

¡Muchas gracias!



OWASP FOUNDATION

Anexo A

Instalación de Juice Shop

Instalación con NodeJS (1/3)



Ir a la página del proyecto y entrar al enlace mostrado.

← → ↻ 🔒 https://owasp.org/www-project-juice-shop/ 🔍 ⚙️ ☆

Watch 152 Star 8,734

OWASP Juice Shop

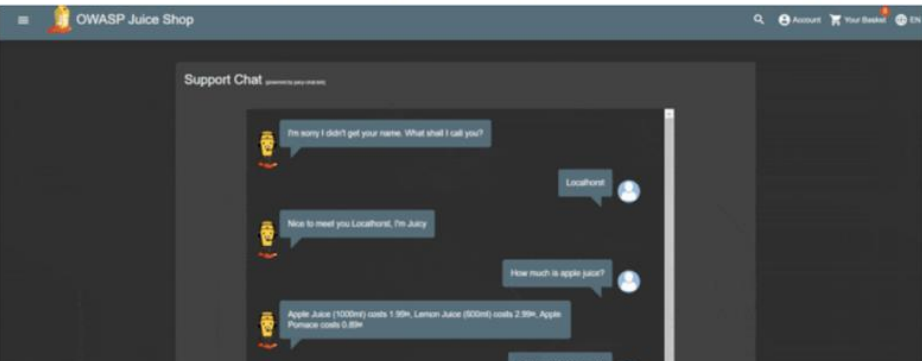
Main Overview News Challenges Learning CTF Ecosystem Supporters



owasp flagship project release v15.2.1 GitHub ★ 8.7k Follow

openssf best practices gold Contributor Covenant v2.0 adopted

OWASP Juice Shop is probably the most modern and sophisticated insecure web application! It can be used in security trainings, awareness demos, CTFs and as a guinea pig for security tools! Juice Shop encompasses vulnerabilities from the entire [OWASP Top Ten](#) along with many other security flaws found in real-world applications!



Support Chat

I'm sorry I didn't get your name. What shall I call you?

Localhost

Nice to meet you Localhost, I'm Julia

How much is apple juice?

Apple Juice (1000ml) costs 1.99€, Lemon Juice (500ml) costs 2.99€, Apple Pommes costs 0.99€

Project Information

Flagship Project

Classification

Tool

Audience

Builder Breaker Defender

Installation

[From Source](#)

[Packaged \(GitHub/SourceForge\)](#)

[Docker Image](#)

Instalación con NodeJS (2/3)



Descargar el paquete correspondiente a tu versión de NodeJS y sistema operativo.

https://github.com/juice-shop/juice-shop#from-sources

README.md

Packaged Distributions

downloads 245k sourceforge downloads 1.3k/month sourceforge downloads 53k

1. Install a 64bit [node.js](#) on your Windows, MacOS or Linux machine
2. Download `juice-shop-<version>_<node-version>_<os>_x64.zip` (or `.tgz`) attached to [latest release](#)
3. Unpack and `cd` into the unpacked folder
4. Run `npm start`
5. Browse to <http://localhost:3000>

Each packaged distribution includes some binaries for `sqlite3` and `libxmljs` bound to the OS and node.js version which `npm install` was executed on.



Instalación con NodeJS (3/3)

- Descomprimir y ejecutar `npm start` dentro del folder.
- Ir a <http://localhost:3000> en tu navegador Google Chrome.

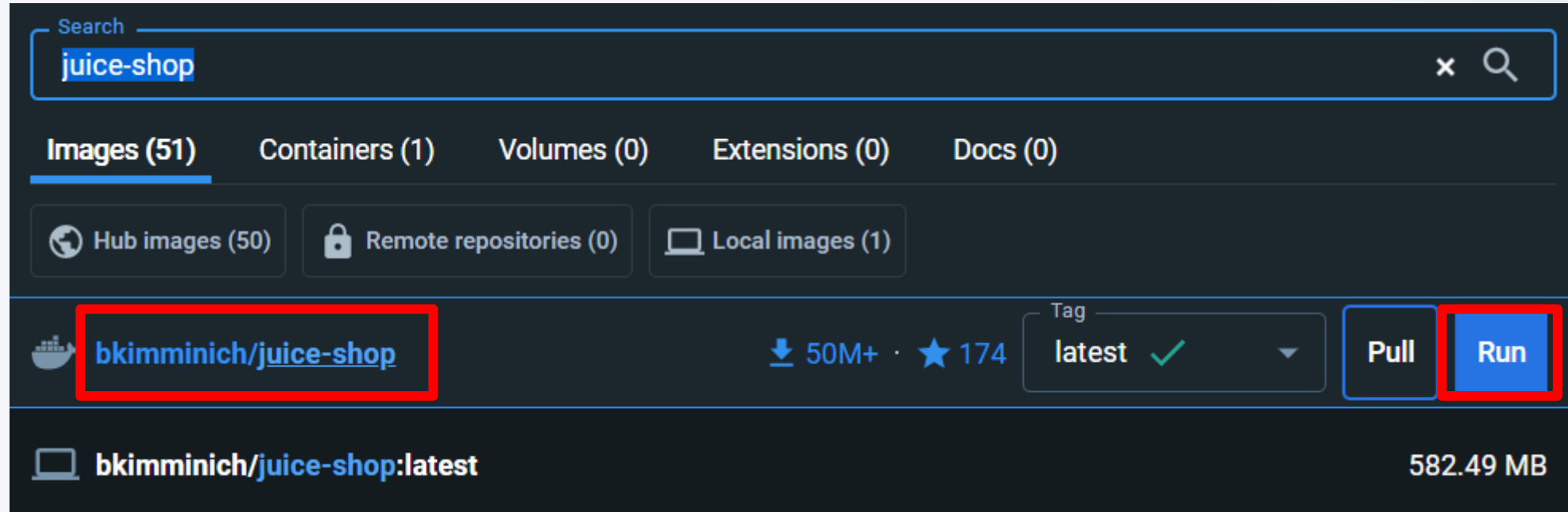
From pre-packaged distribution

1. Install a 64bit [Node.js](#) on your Windows, MacOS or Linux machine.
2. Download `juice-shop-<version>_<node-version>_<os>_x64.zip` (or `.tgz`) attached to the [latest release on GitHub](#).
3. Unpack the archive and run `npm start` in unpacked folder to launch the application
4. Browse to <http://localhost:3000>

Instalación con Docker (1/3)



Abrir Docker Desktop y buscar la imagen “juice-shop” de **bkimminich**. Al ubicarla, dar clic en Run.





Instalación con Docker(2/3)

Establecer las siguientes configuraciones. En nombre del contenedor, indicar como gustes. El puerto debe ser 3000.

Run a new container
bkimminich/juice-shop:latest

Optional settings

Container name
TallerJuiceShop
Random name is generated if you do not provide one.

Ports
Enter "0" to assign randomly generated host ports.

Host port
3000 :3000/tcp

Instalación con Docker (3/3)



Al continuar, ir al apartado de Contenedores y ubicar el contenedor de Juice Shop, dar clic en el link de la columna Puerto.

Containers [Give feedback](#)

Container CPU usage ⓘ
3.32% / 400% (4 cores allocated)

Container memory usage ⓘ
168.6MB / 7.53GB

☰ ☐ Only show running containers

☐	Name	Image	Status	Port(s)
☐	 TallerJuiceShop 9fec31751c4c 	bkimminich/juice-shop:latest	Running	3000:3000 

