

# **LAPORAN**

## **KEMAMAN JARINGAN**

*Teacher : Dr. Ferry Astika Saputra S. ST M. Sc*

**OWASP 10 JUICE SHOP**



DISUSUN OLEH :

**SYIHAB MUHAMMAD UBAIDILLAH**

**3122640043**

**PROGRAM STUDI TEKNIK INFORMATIKA**  
**POLITEKNIK ELEKTRONIKA NEGERI SURABAYA**  
**D4 TI LANJUT JENJANG**

**Tugas :** Jelaskan proses instalasi aplikasi WEB dengan kerentanan OWASP JUICE SHOP <https://owasp.org/www-project-juice-shop/> pada web server yang dijalankan diatas sembarang OS dengan virtualisasi VMWARE ataupun Virtual Box. Jelaskan juga hubungan anantara OWASP 10 2022 dengan aplikasi Juiceshop ! Jelaskan juga 10 kerentanan yang populer di aplikasi web (OWASP 10)

**Jawaban :**

OWASP Juice Shop adalah sebuah aplikasi web yang dirancang untuk melatih dan menguji keamanan aplikasi web. Aplikasi ini dibuat oleh Björn Kimminich, seorang ahli keamanan web dan peneliti keamanan asal Jerman, dan merupakan salah satu proyek dari Open Web Application Security Project (OWASP), sebuah organisasi global nirlaba yang fokus pada keamanan aplikasi web.

Dalam OWASP Juice Shop, pengguna dapat melihat contoh-contoh kerentanan keamanan aplikasi web seperti XSS, CSRF, SQL Injection, dan sebagainya. Pengguna juga dapat mencoba melakukan serangan pada aplikasi web dan mempelajari cara melindungi sistem dari serangan-serangan tersebut.

Untuk menginstall OWASP Juice Shop, pengguna harus mengunduh kode sumbernya dari repositori git resminya di GitHub dan menginstall dependensi yang dibutuhkan. Setelah dependensi terinstal, pengguna dapat menjalankan OWASP Juice Shop di dalam lingkungan pengembangan lokal atau di server web. Setelah Juice Shop berjalan, pengguna dapat mengaksesnya melalui browser dan mulai mengeksplorasi aplikasi web ini untuk melihat contoh-contoh kerentanan keamanan aplikasi web.

Penting untuk diingat bahwa OWASP Juice Shop adalah aplikasi web yang sengaja dirancang untuk memiliki kerentanan dan kelemahan, sehingga sebaiknya penggunaannya hanya untuk tujuan pengujian dan pelatihan keamanan, bukan untuk penggunaan produksi. Selalu perhatikan keamanan sistem dan jangan gunakan OWASP Juice Shop pada sistem yang terhubung ke internet secara langsung.

**Cara Install OWASP Juice Shop**

Cara untuk menginstall aplikasi WEB OWASP JUICE SHOP ada beberapa langkah, dibawah ini adalah cara untuk menginstall aplikasi WEB OWASP JUICE SHOP :

1. Unduh kode sumber OWASP Juice Shop dari repositori git resminya di GitHub.
2. Instal Node.js dan NPM (Node Package Manager) di komputer Anda jika belum terinstal.
3. Buka terminal atau command prompt dan arahkan ke direktori tempat kode sumber OWASP Juice Shop diunduh.
4. Jalankan perintah "npm install" untuk menginstall semua dependensi yang dibutuhkan oleh OWASP Juice Shop.
5. Setelah semua dependensi terinstal, jalankan perintah "npm start" untuk menjalankan OWASP Juice Shop di localhost.
6. Buka browser dan masukkan URL "http://localhost:3000" untuk mengakses OWASP Juice Shop.

Untuk mempelajari dan menguji kerentanan OWASP Juice Shop, pengguna dapat mengikuti instruksi dan petunjuk yang tersedia pada halaman web. Penting untuk diingat bahwa OWASP Juice Shop adalah aplikasi web yang sengaja dibuat dengan kerentanan dan kelemahan untuk keperluan pengujian dan pelatihan keamanan, sehingga tidak seharusnya digunakan pada sistem yang terhubung langsung ke internet dan diharapkan hanya digunakan untuk tujuan pengujian dan pelatihan keamanan. Selalu

perhatikan keamanan sistem dan jangan gunakan OWASP Juice Shop pada sistem yang terhubung ke internet secara langsung.

### **Hubungan OWASP 10 2022 dengan Aplikasi Juice Shop**

OWASP 10 adalah sebuah proyek yang diluncurkan oleh Open Web Application Security Project (OWASP) yang menyoroti sepuluh kerentanan keamanan web paling umum pada aplikasi web. Proyek ini direvisi setiap beberapa tahun sekali untuk mencakup tren terbaru dalam keamanan aplikasi web.

Saat ini, versi terbaru dari OWASP 10 adalah OWASP 102017, yang masih menjadi acuan utama dalam industri keamanan aplikasi web. Namun, OWASP juga telah mengumumkan bahwa mereka sedang mengembangkan OWASP 10 2022, yang diperkirakan akan dirilis dalam waktu dekat. Aplikasi OWASP Juice Shop dikembangkan sebagai sebuah proyek OWASP, yang berarti bahwa aplikasi ini dirancang khusus untuk menunjukkan contoh-contoh dari kerentanan keamanan web yang terdapat pada OWASP 10, seperti SQL Injection, Cross-Site Scripting (XSS), dan sebagainya. OWASP Juice Shop dirancang untuk membantu pengembang dan peneliti keamanan dalam mengembangkan keterampilan dan pengetahuan dalam bidang keamanan aplikasi web.

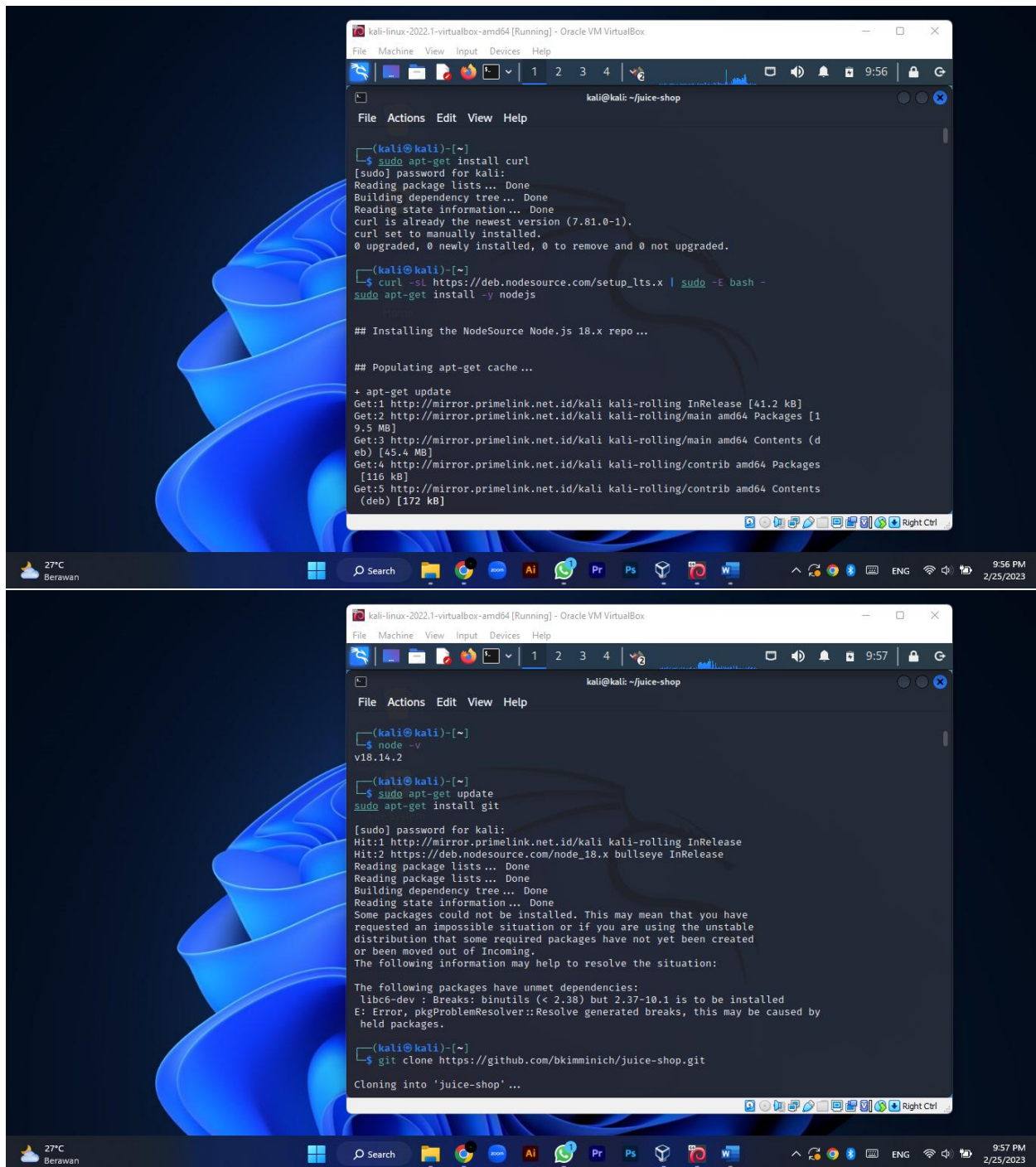
Dengan demikian, dapat dikatakan bahwa OWASP Juice Shop memiliki hubungan yang sangat erat dengan OWASP 10, karena aplikasi ini dibuat untuk menunjukkan dan memberikan pengalaman langsung tentang kerentanan keamanan web yang terdapat pada OWASP 10. Dengan menggunakan OWASP Juice Shop, pengguna dapat mempelajari cara menangani dan menghindari kerentanan keamanan web yang terdapat pada OWASP 10 sehingga dapat mengurangi risiko keamanan pada aplikasi web yang dibuat di masa depan.

### **Kerentanan yang populer di aplikasi web**

1. Injection - Injection terjadi ketika data input dari pengguna tidak divalidasi dengan benar dan disisipkan ke dalam perintah SQL, XPath, atau kode program lainnya, yang dapat mengakibatkan kebocoran data dan penyerangan terhadap aplikasi web.
2. Broken Authentication and Session Management - Kerentanan ini terjadi ketika fungsi autentikasi dan manajemen sesi tidak diimplementasikan dengan benar, yang dapat mengizinkan penyerang untuk mencuri kredensial dan mengambil alih akun pengguna.
3. Cross-Site Scripting (XSS) - XSS terjadi ketika penyerang menyisipkan skrip jahat ke dalam halaman web yang kemudian dieksekusi pada peramban pengguna, yang dapat mengakibatkan pencurian informasi sensitif dan pengambilalihan akun pengguna.
4. Broken Access Control - Kerentanan ini terjadi ketika kontrol akses ke fitur dan sumber daya aplikasi web tidak diimplementasikan dengan benar, yang dapat mengakibatkan penggunaan tidak sah atau tidak sah dari fitur dan sumber daya oleh pengguna yang tidak berwenang.
5. Security Misconfiguration - Kerentanan ini terjadi ketika konfigurasi sistem dan aplikasi web tidak dilakukan dengan benar, sehingga mengakibatkan eksposur data sensitif, akses tidak sah, dan kerentanan keamanan lainnya.
6. Insecure Cryptographic Storage - Kerentanan ini terjadi ketika data sensitif disimpan di tempat yang tidak aman, seperti dalam bentuk plain text atau dengan algoritma enkripsi yang lemah.

7. Insufficient Logging and Monitoring - Kerentanan ini terjadi ketika sistem dan aplikasi web tidak memonitor aktivitas dan kejadian penting secara efektif, yang dapat mengakibatkan kegagalan untuk mendeteksi serangan dan mengambil tindakan yang tepat.
8. Insecure Communication - Kerentanan ini terjadi ketika komunikasi antara aplikasi web dan klien atau server tidak dilindungi dengan benar, sehingga terbuka untuk serangan seperti man-in-the-middle.
9. Using Components with Known Vulnerabilities - Kerentanan ini terjadi ketika aplikasi web menggunakan komponen dan kerangka kerja yang sudah diketahui memiliki kerentanan keamanan yang belum diperbaiki.
10. Insufficient Input Validation - Kerentanan ini terjadi ketika data input dari pengguna tidak divalidasi dengan benar sebelum digunakan dalam kode program, yang dapat mengakibatkan kerentanan seperti SQL injection dan XSS.

## CAPTURE INSTALASI OWASP JUICE SHOP



The image displays two sequential screenshots of a Kali Linux terminal window, showing the process of installing Node.js and cloning the OWASP Juice Shop repository.

**Top Screenshot:**

```
kali@kali: ~/juice-shop
File Actions Edit View Help

(kali@kali)~$ sudo apt-get install curl
[sudo] password for kali:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
curl is already the newest version (7.81.0-1).
curl set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.

(kali@kali)~$ curl -sL https://deb.nodesource.com/setup_18.x | sudo -E bash -
sudo apt-get install -y nodejs

## Installing the NodeSource Node.js 18.x repo ...

## Populating apt-get cache ...

+ apt-get update
Get:1 http://mirror.primelink.net.id/kali kali-rolling InRelease [41.2 kB]
Get:2 http://mirror.primelink.net.id/kali kali-rolling/main amd64 Packages [1
9.5 MB]
Get:3 http://mirror.primelink.net.id/kali kali-rolling/main amd64 Contents (d
eb) [45.4 MB]
Get:4 http://mirror.primelink.net.id/kali kali-rolling/contrib amd64 Packages
[116 kB]
Get:5 http://mirror.primelink.net.id/kali kali-rolling/contrib amd64 Contents
(deb) [172 kB]
```

**Bottom Screenshot:**

```
(kali@kali)~$ node -v
v18.14.2

(kali@kali)~$ sudo apt-get update
sudo apt-get install git

[sudo] password for kali:
Hit:1 http://mirror.primelink.net.id/kali kali-rolling InRelease
Hit:2 https://deb.nodesource.com/node_18.x bullseye InRelease
Reading package lists... Done
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Some packages could not be installed. This may mean that you have
requested an impossible situation or if you are using the unstable
distribution that some required packages have not yet been created
or been moved out of Incoming.
The following information may help to resolve the situation:

The following packages have unmet dependencies:
 libc6-dev : Breaks: binutils (< 2.38) but 2.37-10.1 is to be installed
E: Error, pkgProblemResolver::Resolve generated breaks, this may be caused by
held packages.

(kali@kali)~$ git clone https://github.com/bkimminich/juice-shop.git
Cloning into 'juice-shop' ...
```

```
kali-linux-2022.1-virtualbox-amd64 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
1 2 3 4
kali@kali: ~/juice-shop
File Actions Edit View Help
(kali@kali)~$ git clone https://github.com/bkimminich/juice-shop.git
Cloning into 'juice-shop' ...
remote: Enumerating objects: 118011, done.
remote: Counting objects: 100% (29/29), done.
remote: Compressing objects: 100% (20/20), done.
remote: Total 118011 (delta 11), reused 23 (delta 9), pack-reused 117982
Receiving objects: 100% (118011/118011), 181.56 MiB | 2.28 MiB/s, done.
Resolving deltas: 100% (91670/91670), done.
(kali@kali)~$ cd juice-shop/
(kali@kali)~/juice-shop$ npm install
npm WARN deprecated samsam@1.1.2: This package has been deprecated in favour
of @sinonjs/samsam
npm WARN deprecated formatio@1.1.1: This package is unmaintained. Use @sinonj
s/formatio instead
npm WARN deprecated topo@2.0.3: This module has moved and is now available at
@hapi/topo. Please update your dependencies as this version is no longer mai
ntained and may contain bugs and security issues.
npm WARN deprecated hoek@5.0.4: This version has been deprecated in accordanc
e with the hapi support policy (hapi.im/support). Please upgrade to the latest
version to get the best features, bug fixes, and security patches. If you are
unable to upgrade at this time, paid support is available for older versions
(hapi.im/commercial).
npm WARN deprecated eivindfeldstad-dot@0.0.1: Use @eiviffj/dot instead
(kali@kali)~/juice-shop$
```

```
kali-linux-2022.1-virtualbox-amd64 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
1 2 3 4
kali@kali: ~/juice-shop
File Actions Edit View Help
npm ERR! /home/kali/.npm/_logs/2023-02-25T14_28_23_778Z-debug-0.log
(kali@kali)~/juice-shop$ npm start
> juice-shop@14.5.1 start
> node build/app

info: All dependencies in ./package.json are satisfied (OK)
info: Chatbot training data botDefaultTrainingData.json validated (OK)
info: Detected Node.js version v18.14.2 (OK)
info: Detected OS linux (OK)
info: Detected CPU x64 (OK)
info: Configuration default validated (OK)
info: Entity models 19 of 19 are initialized (OK)
info: Required file server.js is present (OK)
info: Required file index.html is present (OK)
info: Required file styles.css is present (OK)
info: Required file main.js is present (OK)
info: Required file tutorial.js is present (OK)
info: Required file polyfills.js is present (OK)
info: Required file runtime.js is present (OK)
info: Required file vendor.js is present (OK)
info: Port 3000 is available (OK)
info: Server listening on port 3000
^C
```