# OWASP

# OWASP

> The Open Web Application Security Project® (OWASP) is a nonprofit foundation that works to improve the security of software. Through community-led open-source software projects, hundreds of local chapters worldwide, tens of thousands of members, and leading educational and training conferences, the OWASP Foundation is the source for developers and technologists to secure the web.

# Core Values

- **Open**: Everything at OWASP is radically transparent from our finances to our code.

- **Innovative**: We encourage and support innovation and experiments for solutions to software security challenges.

- **Global**: Anyone around the world is encouraged to participate in the OWASP community.

- **Integrity**: Our community is respectful, supportive, truthful, and vendor neutral

# OWASP Projects

# OWASP Projects

An OWASP project is a collection of related tasks that have a defined roadmap and team members.

| Project Type | Examples |
|---|---|
| Tool | ZAP, Dependency Check, DefectDojo, Dependency-Track, Juice Shop |
| Code | ModSecurity Core Rule Set, Java Encoder |
| Documentation | OWASP Top 10, Application Security Verification Standard (ASVS), Cornucopia, Security Knowledge Framework, Cheat Sheet Series |

# Project Lifecycle

| Level | Icon | Description |
|---|---|---|
| Incubator | | OWASP Incubator projects represent the experimental playground where projects are still being fleshed out, ideas are still being proven, and development is still underway. |
| Labs | | OWASP Labs projects represent projects that have produced an OWASP reviewed deliverable of value. |
| Flagship | | The OWASP Flagship designation is given to projects that have demonstrated strategic value to OWASP and application security as a whole. |

# OWASP Chapters

# OWASP Chapters

OWASP Local Chapters build community for application security professionals around the world. Our Local Chapter Meetings are free and open to anyone to attend so both members and non-members are always welcomed. Local meetings include:

- Training to improve your skills
- Talks relevant to your work
- Networking opportunities

# OWASP German Chapter

There is one Chapter for Germany in total which is complemented by a so-called OWASP Stammtisch each in several metropolitan areas such as München, Frankfurt, Hamburg, Stuttgart, Köln, Hannover, Karlsruhe, Dresden, Ruhrpott, Heilbronn-Franken.

# OWASP Top 10

# OWASP Top 10
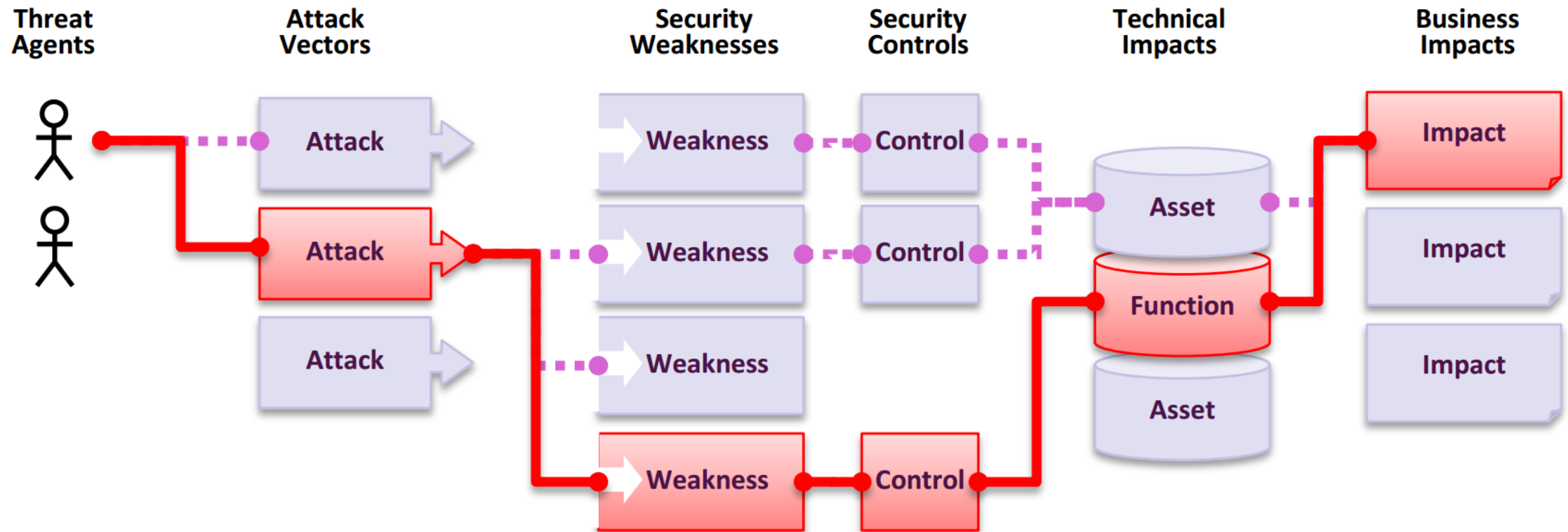
| 1 | Injection | 6 | Security Misconfiguration |
|---|---|---|---|
| 2 | Broken Authentication | 7 | Cross-Site-Scripting (XSS) |
| 3 | Sensitive Data Exposure | 8 | Insecure Deserialization |
| 4 | XML External Entities | 9 | Using Components with Known Vulnerabilities |
| 5 | Broken Access Control | 10 | Insufficient Logging & Monitoring |

# Application Security Risks

12

# Risk Rating Table

| Threat Agents | Exploitability | Weakness Prevalence | Weakness Detectability | Technical Impacts | Business Impacts |
|---|---|---|---|---|---|
| App Specific | EASY: 3 | WIDESPREAD: 3 | EASY: 3 | SEVERE: 3 | App / Business Specific |
| | AVERAGE: 2 | COMMON: 2 | AVERAGE: 2 | MODERATE: 2 | |
| | DIFFICULT: 1 | UNCOMMON: 1 | DIFFICULT: 1 | MINOR: 1 | |

ℹ️ *Based on the OWASP Risk Rating Methodology*

# Some(!) Additional Risks to Consider

| | |
|---|---|
| Cross-Site Request Forgery (CSRF) | Unvalidated Forward and Redirects |
| Uncontrolled Resource Consumption ('Resource Exhaustion', 'AppDoS') | Improper Control of Interaction Frequency (Anti-Automation) |
| Unrestricted Upload of File with Dangerous Type | Inclusion of Functionality from Untrusted Control Sphere (3rd Party Content) |
| User Interface (UI) Misrepresentation of Critical Information (Clickjacking etc.) | Server-Side Request Forgery (SSRF) |

# Other Resources on AppSec

- **SANS** Software Security Community

    - CWE/SANS TOP 25 Most Dangerous Software Errors

    - Securing Web Application Technologies [SWAT] Checklist

- **CWE** Common Weakness Enumeration

    - Community-developed list of common software security weaknesses

# OWASP Juice Shop

# OWASP Juice Shop

> 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!

# Installation

- Individual (local or hosted) instance per student

- Runs on node.js, Docker, Vagrant and in the ☁️

## ⚠️ Hacking Rules

- Do **not** look at the source code on GitHub

- Do **not** look at GitHub issues, PRs etc.

- Do **not** cheat (with online tutorials or walkthroughs) before trying

- Report problems during exercises immediately

# Official Companion Guide

Pwning OWASP Juice Shop […] will give you a complete overview of the vulnerabilities found in the application including hints how to spot and exploit them. In the appendix you will even find complete step-by-step solutions to every challenge. The ebook is published under CC BY-NC-ND 4.0 and is available **for free** online-readable. The latest officially released edition is available **for free** on LeanPub in PDF, Kindle and ePub format.

This is the official companion guide to the OWASP Juice Shop application. Being a web application with a vast number of intended security vulnerabilities, the OWASP Juice Shop is supposed to be the opposite of a best practice or template application for web developers: it is an awareness, training, demonstration and exercise tool for security risks in modern web applications. The OWASP Juice Shop is an open-source project hosted by the non-profit Open Web Application Security Project (OWASP) and is developed and maintained by volunteers.

Björn Kimminich has over two decades of programming experience with expertise on software sustainability, clean code and test automation as well as application security. He is the project leader of the OWASP Juice Shop and member of the German OWASP chapter board.

# Exercise 1.1

## Install the OWASP Juice Shop `v12.x`

1. Install the latest Node.js 14.x (or 12.x) release on your computer

2. On https://github.com/bkimminich/juice-shop#setup follow the instructions for either

   - From Sources or

   - Packaged Distributions

🐳 *If you want to use Docker you need to run the container with* `docker run -d -e "NODE_ENV=unsafe" -p 3000:3000 bkimminich/juice-shop` *or you won't be able to solve several of the exercises.*

# Exercise 1.2

**Happy path shopping tour**

1. Register a user account at your local Juice Shop

2. Browse the inventory and purchase some products

3. Try out all other functionality you find in the application

# Exercise 1.3

## Score Board

    1. Find the hidden Score Board in the Juice Shop (⭐)

ℹ️ *You can let the application's friendly Hacking Instructor guide you through this exercise by clicking "Help getting started" on the welcome banner or in the side bar.*

# Exercise 1.4

## Transfer your hacking progress

1. If you keep using the same computer *and* do not delete your cookies your browser will persist and restore your hacking progress.

2. It is still recommended to make a backup of your progress regularly.

3. You can also use this `JSON` backup to restore your progress and settings on any other computer.