*Web Application Pen-Testing*

*AY 2022/2023*

*Week 15 Practical*

*OWASP Top 10 - 2021*

*A08:2021-* *Software and Data Integrity Failures*

*A08 Related Challenges on OWASP Juice Shop*

#### OWASP Top 10 – 2021 – Image

Diagram

Description automatically generated

*Table of Contents*

[1. A08:2021- Software and Data Integrity Failures 3](#_Toc125534936)

[*1.1.* *Description* 3](#_Toc125534937)

[*1.2.* *Example Attack Scenarios* 3](#_Toc125534938)

[2. Setup 4](#_Toc125534939)

[*2.1.* *Start and Login to Kali Linux VM with Host-only enabled* 4](#_Toc125534941)

[*2.2.* *[If not already done] Download, Extract, Power On, & Login: Ubuntu 64-bit-WAPT-VVM* 5](#_Toc125534942)

[*2.3.* *Run OWASP Juice Shop Docker Container* 6](#_Toc125534943)

[*2.4.* *Browse OWASP Juice Shop from Kali Linux VM* 7](#_Toc125534944)

[*2.5.* *[If not already done] Create Account & Login to JuiceBar on your HOST OS (Windows)* 7](#_Toc125534945)

[3. Attempt “A08:2021- Software and Data Integrity Failures” related challenges on OWASP Juice Shop 9](#_Toc125534946)

[*3.1.* *Download OWASP Juice Shop Challenge Walkthroughs from JuiceBar on your HOST OS (Windows)* 9](#_Toc125534950)

# A08:2021- Software and Data Integrity Failures

A new category for 2021 focuses on making assumptions related to software updates, critical data, and CI/CD pipelines without verifying integrity. One of the highest weighted impacts from Common Vulnerability and Exposures/Common Vulnerability Scoring System (CVE/CVSS) data. Notable Common Weakness Enumerations (CWEs) include CWE-829: Inclusion of Functionality from Untrusted Control Sphere, CWE-494: Download of Code Without Integrity Check, and CWE-502: Deserialization of Untrusted Data.

## *Description*

Software and data integrity failures relate to code and infrastructure that does not protect against integrity violations. An example of this is where an application relies upon plugins, libraries, or modules from untrusted sources, repositories, and content delivery networks (CDNs). An insecure CI/CD pipeline can introduce the potential for unauthorized access, malicious code, or system compromise. Lastly, many applications now include auto-update functionality, where updates are downloaded without sufficient integrity verification and applied to the previously trusted application. Attackers could potentially upload their own updates to be distributed and run on all installations. Another example is where objects or data are encoded or serialized into a structure that an attacker can see and modify is vulnerable to insecure deserialization.

## *Example Attack Scenarios*

Scenario #1 Update without signing: Many home routers, set-top boxes, device firmware, and others do not verify updates via signed firmware. Unsigned firmware is a growing target for attackers and is expected to only get worse. This is a major concern as many times there is no mechanism to remediate other than to fix in a future version and wait for previous versions to age out.

Scenario #2 SolarWinds malicious update: Nation-states have been known to attack update mechanisms, with a recent notable attack being the SolarWinds Orion attack. The company that develops the software had secure build and update integrity processes. Still, these were able to be subverted, and for several months, the firm distributed a highly targeted malicious update to more than 18,000 organizations, of which around 100 or so were affected. This is one of the most far-reaching and most significant breaches of this nature in history.

Scenario #3 Insecure Deserialization: A React application calls a set of Spring Boot microservices. Being functional programmers, they tried to ensure that their code is immutable. The solution they came up with is serializing the user state and passing it back and forth with each request. An attacker notices the "rO0" Java object signature (in base64) and uses the Java Serial Killer tool to gain remote code execution on the application server.

[Source: <https://owasp.org/Top10/A08_2021-Software_and_Data_Integrity_Failures/>]

# Setup



## *Start and Login to Kali Linux VM with Host-only enabled*

*Make sure the Virtual Machine Settings 🡪 Network Adapter 🡪 Host-only*

|  |  |
| --- | --- |
| *Graphical user interface, text  Description automatically generated* | ***Login*** *into this Kali Linux VM*  *Type in the KALI\_IP address below:*  *XXX.XXX.XX.XXX*  *Graphical user interface, application  Description automatically generated* |

|  |  |
| --- | --- |
| *Tools with solid fill* | *In case your Kali Linux is* ***not responding*** *to changing to NAT (i.e., still not connected to the Internet). You can restart Kali Linux’s Ethernet Interface (eth0) by typing the following* ***2 commands one after the other*** *into the Kali Linux’s Terminal Emulator and press Enter:* |

*sudo ifdown eth0*

*Text

Description automatically generated*

*sudo ifup eth0*

*Text

Description automatically generated*

## *[If not already done] Download, Extract, Power On, & Login:* *Ubuntu 64-bit-WAPT-VVM*

If not already done: Please download this VM (preferably using your home WiFi). This is a 6.36 GB, 7z compressed VMware Image File consisting of Vulnerable Applications: Juice Shop, Web Goat, Mutillidae II, crAPI, and vAPI. In case of limited storage space you can download this VM into your portable external storage and run the VM from portable external storage.

<https://drive.google.com/file/d/1EEhVRFbfRQFZxhGMu-HcjxHpzmee5ope/view?usp=sharing>

1. After download, extract Ubuntu 64-bit-WAPT-VVM.7z
2. Inside the Ubuntu 64-bit-WAPT-VVM folder, double click on Ubuntu 64-bit-WAPT-VVM.vmx
3. The VM should have opened in your VMWare Workstation, click “Power on this virtual machine”

Graphical user interface, application

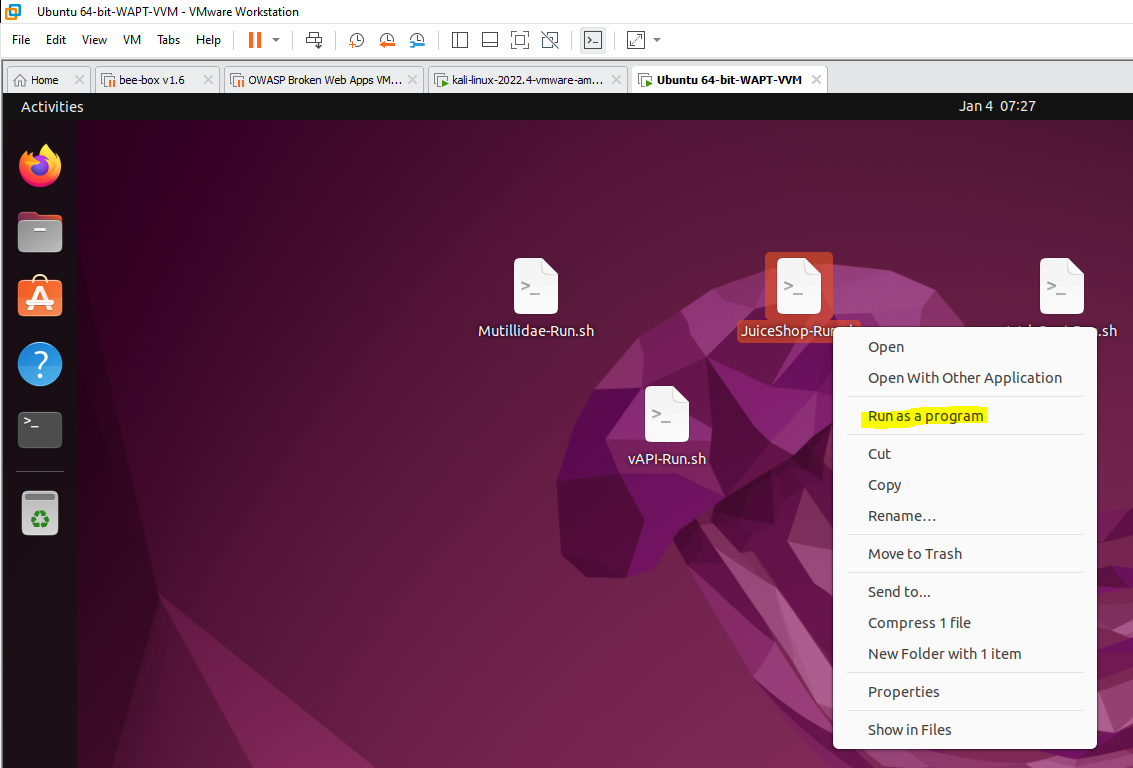
Description automatically generated

1. After the VM bootup, click on “wapt user” to enter the Password: wapt@123 and press Enter

|  |  |
| --- | --- |
| Graphical user interface, application, Teams  Description automatically generated |  |

## *Run OWASP Juice Shop Docker Container*

On the desktop right click on JuiceShop-Run.sh and select “Run as a program”



Make sure that OWASP Juice Shop Docker Container is running. Click the “Terminal” icon from the left-hand side menu. Type the following command into the Terminal and press Enter:

docker container ps

You should notice that the bkimminich/juice-shop status is Up and it is receiving requests on port 3000

|  |  |
| --- | --- |
| A screenshot of a cell phone  Description automatically generated with medium confidence |  |

Obtain the IP address of Ubuntu 64-bit-WAPT-VVM. Type the following command into the Terminal and press Enter:

ip -4 addr

In the output you should be able to see the IP address of Ubuntu 64-bit-WAPT-VVM. Type in the IP address of Ubuntu 64-bit-WAPT-VVM\_IP here: XXX.XXX.XXX.XXX

Text

Description automatically generated

## *Browse OWASP Juice Shop from Kali Linux VM*

Type the following into the address bar of the Kali Linux Web Browser and hit enter:

http:// Ubuntu 64-bit-WAPT-VVM\_IP:3000

Please give it some time to load the webpage

Graphical user interface

Description automatically generated

|  |  |
| --- | --- |
| *Tools with solid fill* | *In case you are not able to load the Juice Shop website, restart* Ubuntu 64-bit-WAPT-VVM and redo the steps in 2.3. |

## *[If not already done] Create Account & Login to JuiceBar on your HOST OS (Windows)*

JuiceBar was developed by Diploma in CSF Year 3 students (Yi Jing, Xin Min, Melvin, and Brayden) as part of their capstone project under the supervision of Mr. Tan Hock Guan. The aim of this capstone project is to use the OWASP Juice Shop to research and develop a set of reliable documentation on the different types of challenges found on the web application. The purpose of the documentation is to share with people who are interested in learning more about different security tools, and in the future, it can also be used in security training.

This website consolidates different challenges categorized into OWASP 2017 top 10 vulnerabilities, OWASP 2021 top 10 vulnerabilities and challenges with different difficulties based on the number of stars they have.

Graphical user interface, text, application, email

Description automatically generated

Type the following into the address bar of your HOST OS (Windows 10/11) Web Browser and hit enter:

<http://34.142.193.111/>

Click Register.

Graphical user interface, text, application

Description automatically generated

Proceed to create an account, use your personal Email address. Confirm by clicking on the link sent to your email inbox with the Subject Title: Confirm your email for JuiceBar.

Graphical user interface, application

Description automatically generated

Proceed to Login to JuiceBar.

Graphical user interface, text, application, chat or text message

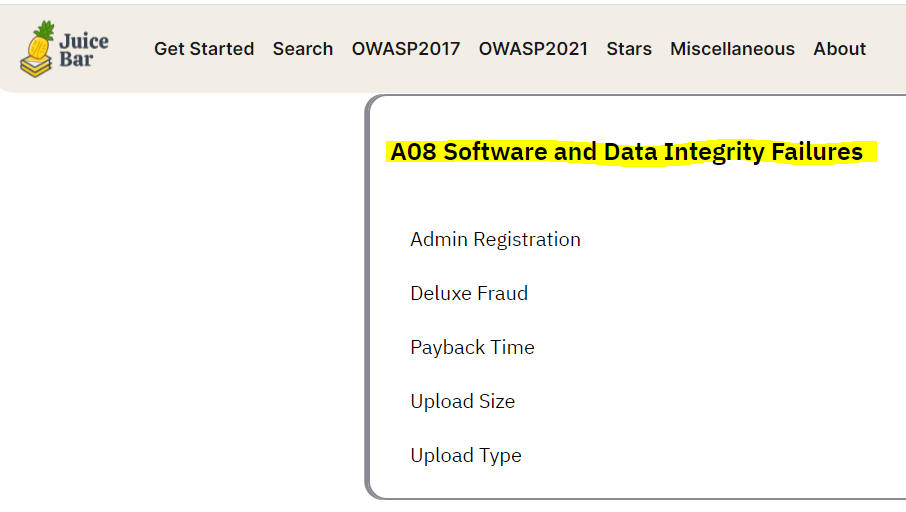
Description automatically generated

# Attempt “A08:2021- Software and Data Integrity Failures” related challenges on OWASP Juice Shop



## *Download OWASP Juice Shop Challenge Walkthroughs from JuiceBar on your HOST OS (Windows)*

Click on “OWASP2021” on the top menu bar and select “A08 Software and Data Integrity Failures” as shown below:



Select a challenge and once inside the challenge scroll down and you will notice “Steps” which could be used as hints to solve this challenge, or you can “Download Document” containing the step by step walkthrough to solve this challenge.

Graphical user interface, text, application

Description automatically generated

Either based on the “Steps” or the “Downloaded Document” you can now use your Kali Linux and OWASP Juice Shop that is running at http:// Ubuntu 64-bit-WAPT-VVM\_IP:3000 to attempt these challenges.