*Web Application Pen-Testing*

*AY 2022/2023*

*Total Recall Challenge II*

*(A challenge to recall all the skills you learnt thus far)*

*Submission Deadline: Sunday 11th Dec 2022 23:59*

*Pen-Test “The BodgeIt Store” Website*

*Table of Contents*

[*1.* *The BodgeIt Store* 3](#_Toc120709183)

[*2.* *Reconnaissance Phase – Enumeration* 3](#_Toc120709184)

[*2.1.* *Output from dirb tool (Ref. Week 2.1)* 3](#_Toc120709187)

[*2.2.* *Output from ZAP Spidering tool (Ref. Week 2.1)* 4](#_Toc120709188)

[*2.3.* *Any other Enumeration tools used (Optional)* 4](#_Toc120709189)

[*2.4.* *Manually browse and understand the The BodgeIt Store website as if you are a customer of this website and note down what interesting aspects you observed* 4](#_Toc120709190)

[*3.* *Discovery Phase – Automated Vulnerability Scan* 5](#_Toc120709191)

[*3.1.* *Output from Authenticated Scan using ZAP (Ref. Week 4.1)* 5](#_Toc120709195)

[*3.2.* *Any other Automated Vulnerability Scanners used (Optional)* 5](#_Toc120709196)

[*4.* *Exploitation & Post Exploitation Phase* 6](#_Toc120709197)

# *The BodgeIt Store*

The BodgeIt Store is a vulnerable web application. It is available inside the OWASPBWA VM as shown below.

Text, application

Description automatically generated

Please fill up this document with ONLY important screenshots mainly indicating the tool used and the outputs observed. Submit this filled up document via POLITEMall by Sunday 11th Dec 2022 23:59.

Table

Description automatically generated

# *Reconnaissance Phase – Enumeration*



## *Output from dirb tool (Ref. Week 2.1)*

|  |
| --- |
|  |

## *Output from ZAP Spidering tool (Ref. Week 2.1)*

|  |
| --- |
|  |

## *Any other Enumeration tools used (Optional)*

|  |
| --- |
|  |

## *Manually browse and understand the The BodgeIt Store website as if you are a customer of this website and note down what interesting aspects you observed*

Understand what the website is doing, e.g., you can register an account, login using the registered account, and explore other pages that are visible after your login, etc.

|  |
| --- |
|  |

# *Discovery Phase – Automated Vulnerability Scan*

The BodgeIt Store is known to have the following vulnerabilities. See if you can discover (scan), exploit, and post exploit these vulnerabilities (as many as possible).

The “About Us” page leads to a “Scoring Page” (<http://OWASPBWAP_IP/bodgeit/score.jsp>)

As you solve the challenges the indicators will turn green:

Graphical user interface, text

Description automatically generated



## *Output from Authenticated Scan using ZAP (Ref. Week 4.1)*

While doing the authenticated scan please scan for individual vulnerabilities listed above rather than scan the entire website, it might take a long time and the output could be quite confusing.

|  |
| --- |
| **http://192.168.36.130/bodgeit/login.jsp**  **http://192.168.36.130/bodgeit/contact.jsp**  **http://192.168.36.130/bodgeit/admin.jsp**  [**http://192.168.36.130/bodgeit/register.jsp**](http://192.168.36.130/bodgeit/register.jsp) |

## *Any other Automated Vulnerability Scanners used (Optional)*

#### [14 best open-source web application vulnerability scanners <https://resources.infosecinstitute.com/topic/14-popular-web-application-vulnerability-scanners/>]

|  |
| --- |
|  |

# *Exploitation & Post Exploitation Phase*

Try to exploit or do post exploitation by yourself, if you need help follow these 2 links:

<https://resources.infosecinstitute.com/topic/the-bodgeit-store-part-1-2/>

<https://resources.infosecinstitute.com/topic/the-bodgeit-store-part-2/>

* Document with screenshot evidence of completing each of the above listed vulnerabilities.
* Title each vulnerability and attach the screenshot below each vulnerability.
* Categorize each of the vulnerabilities under one of the OWASP Top 10 2021.

**A3 – Injection**

Reflected XSS

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application, Teams

Description automatically generated

Stored XSS

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application

Description automatically generated

XSS (with some input validation)

A picture containing graphical user interface

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text

Description automatically generated with medium confidenceA picture containing chart

Description automatically generated

Thus, it is inferred that double quotation marks (“”) and the script tag have been stripped

Graphical user interface, text, application

Description automatically generatedGraphical user interface, application

Description automatically generated

SQLi

Graphical user interface, text, application

Description automatically generated

**A7 – Identification & Authentication Failures**

Brute forcing (given the email/username)

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application

Description automatically generatedGraphical user interface, application

Description automatically generated

**Graphical user interface, text, application

Description automatically generated**

Access someone else’s basket (using cookies)

Graphical user interface, text

Description automatically generated

**Graphical user interface, text, application, email

Description automatically generated**

**A picture containing graphical user interface

Description automatically generated**

**Logic flaw (get store to owe you money)**

**Graphical user interface

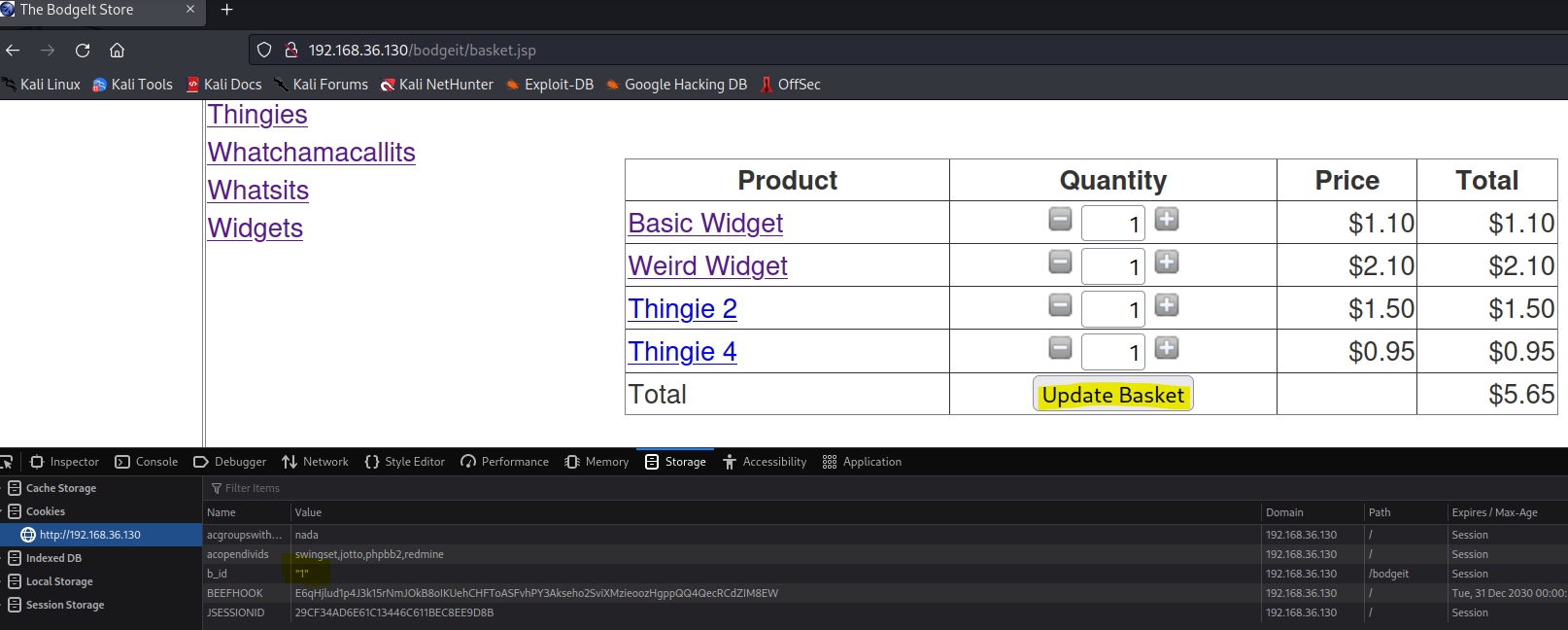
Description automatically generated with medium confidence**

**Graphical user interface, text, email

Description automatically generated**

**Graphical user interface

Description automatically generated with low confidence**

****