

This room uses the Juice Shop vulnerable web application to learn how to identify and exploit common web application vulnerabilities.

*Open for business!*

*Within this room, we will look at [OWASP's TOP 10 vulnerabilities](#) in web applications. You will find these in all types of web applications. But for today we will be looking at OWASP's own creation, Juice Shop!*



The **FREE** Burpsuite rooms '[Burpsuite Basics](#)' and '[Burpsuite Repeater](#)' are recommended before completing this room!

Juice Shop is a large application so we will not be covering every topic from the top 10.

We will, however, cover the following topics which we recommend you take a look at as you progress through this room.

← ----- →

*Injection*

*Broken Authentication*

*Sensitive Data Exposure*

*Broken Access Control*

*Cross-Site Scripting XSS*

← ----- →

**PLEASE NOTE!**

**[Task 3] and onwards will require a flag, which will be displayed on completion of the task.**

Press enter or click to view image in full size

You successfully solved a challenge: Error Handling (Provok an error that is neither very gracefully nor consistently handled.)

[[169940f83378cc420ae4fdeb9c1f73631a2baee6](#)

 Copy to clipboard

X

## Troubleshooting

The web app takes about 2–5 minutes to load, so please be patient!

Temporarily disable burp in your proxy settings for the current browser. Refresh the page and the flag will be shown.

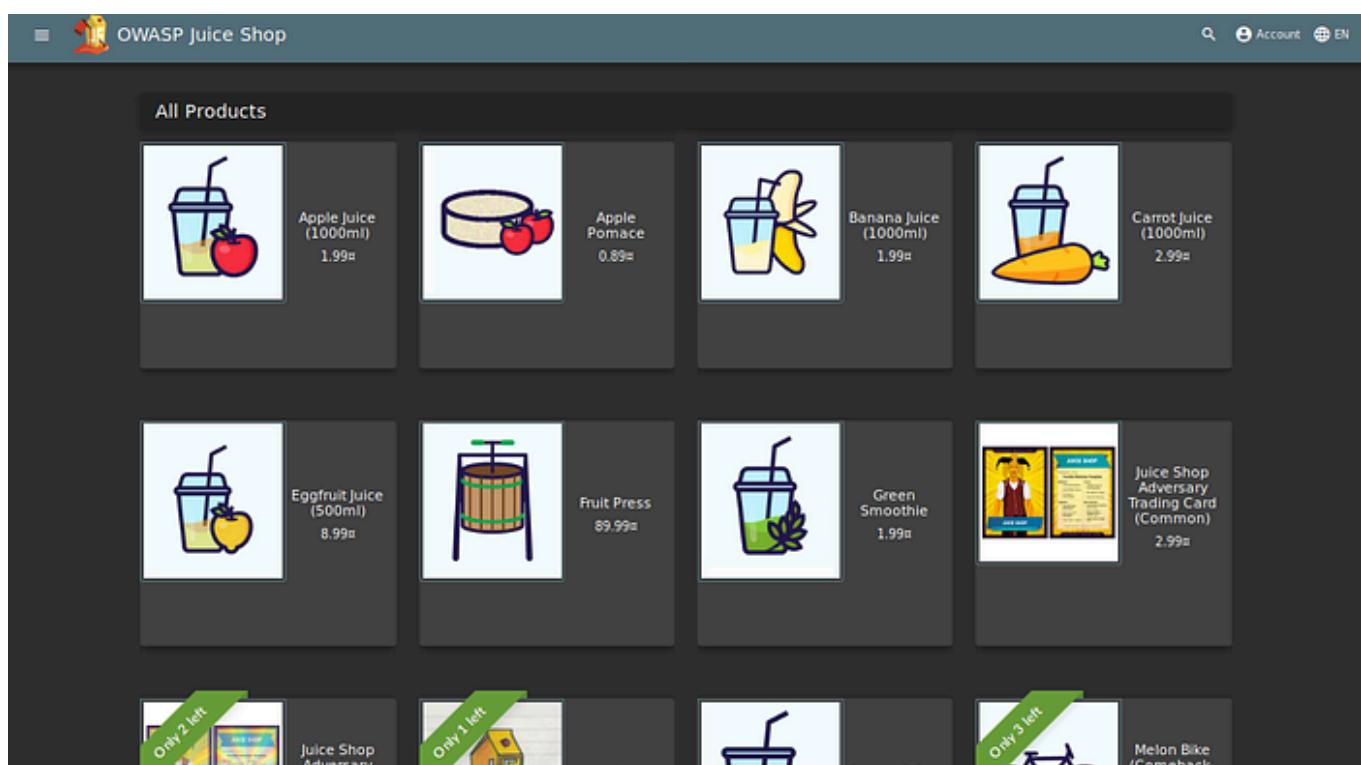
(This is not an issue with the application but an issue with burp stopping the flag from being shown. )

If you are doing the XSS Tasks and they are not working. Clear your cookies and site data, as this can sometimes be an issue.

If you are sure that you have completed the task but it's still not working. Go to **[Task 8]**, as this will allow you to check its completion.

Let's go on an adventure!

Press enter or click to view image in full size

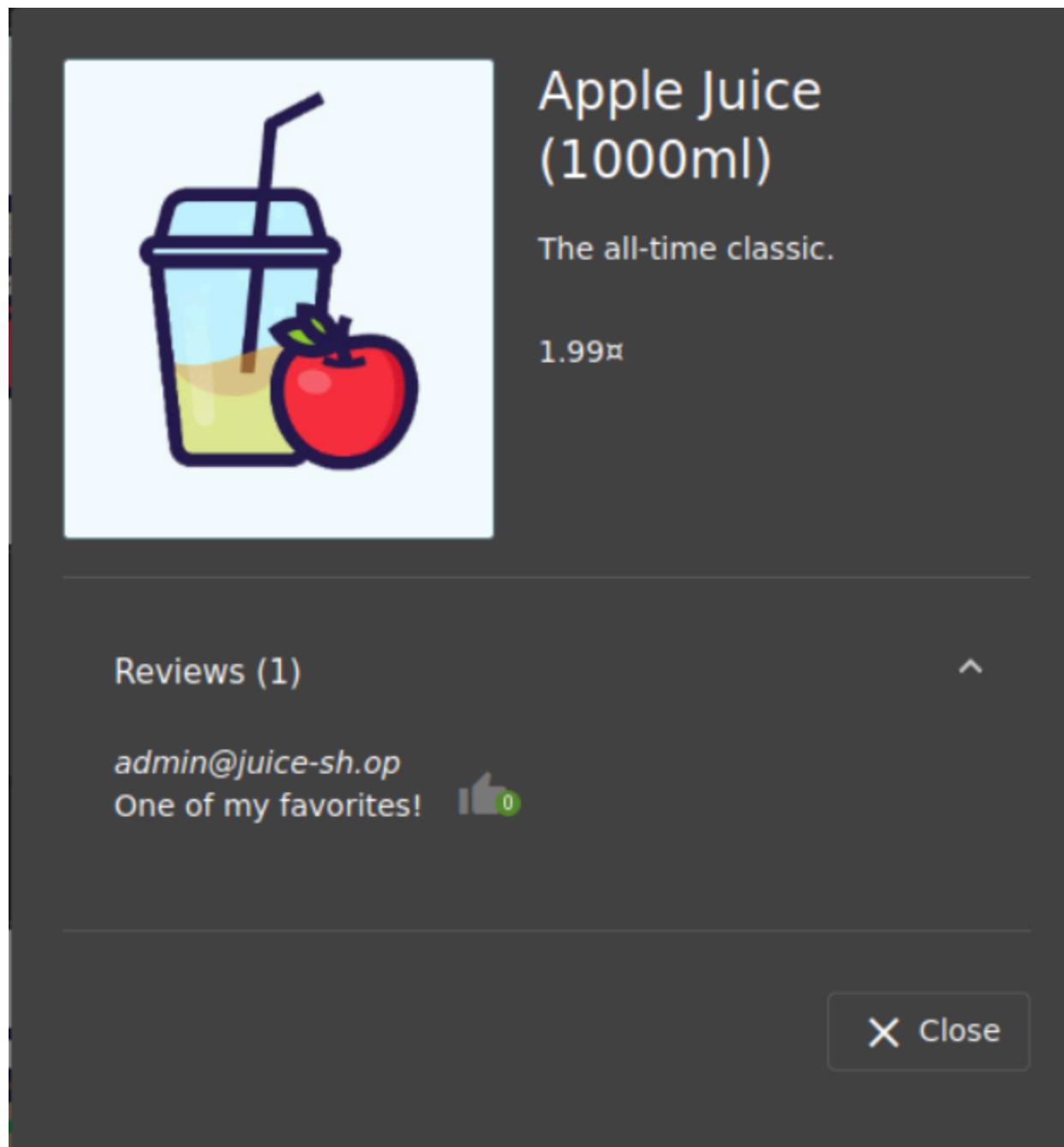


Before we get into the actual hacking part, it's good to have a look around. In Burp, set the Intercept mode to off and then browse around the site. This allows Burp to log different requests from the server that may be helpful later.

This is called **walking through** the application, which is also a form of **reconnaissance**!

**Ques 1:** What's the Administrator's email address?

Ans 1: [admin@juice-sh.op](mailto:admin@juice-sh.op)



**Ques 2:** What parameter is used for searching?

Ans 2: q



**Ques 3:** What show does Jim reference in his review?

Ans 3: Star Trek

*Inject the juice*

Press enter or click to view image in full size

The screenshot shows the OWASP Juice Shop login interface. The page has a dark background with light-colored UI elements. At the top, there's a navigation bar with a menu icon, the logo, and account information. The main area is a modal dialog titled "Login". It contains two input fields: "Email" and "Password", both with placeholder text. Below the password field is a "Forgot your password?" link. At the bottom of the modal are a "Log in" button with a user icon, a "Remember me" checkbox, and a "Not yet a customer?" link.

*This task will be focusing on injection vulnerabilities. Injection vulnerabilities are quite dangerous to a company as they can potentially cause downtime and/or loss of data. Identifying injection points within a web application is usually quite simple, as most of them will return an error. There are many types of injection attacks, some of them are:*

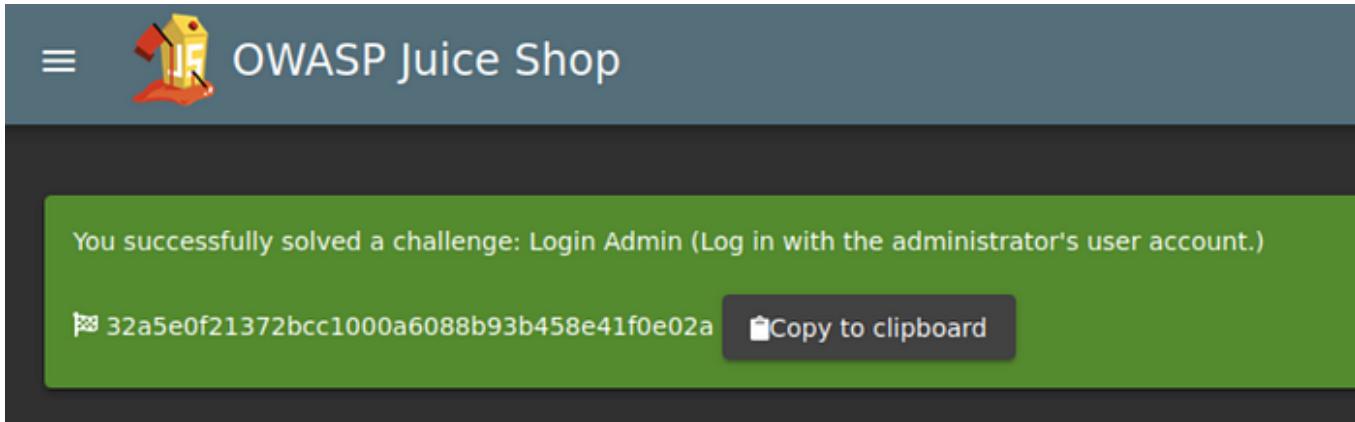
Press enter or click to view image in full size

SQL Injection	SQL Injection is when an attacker enters a malicious or malformed query to either retrieve or tamper data from a database. And in some cases, log into accounts.
Command Injection	Command Injection is when web applications take input or user-controlled data and run them as system commands. An attacker may tamper with this data to execute their own system commands. This can be seen in applications that perform misconfigured ping tests.
Email Injection	Email injection is a security vulnerability that allows malicious users to send email messages without prior authorization by the email server. These occur when the attacker adds extra data to fields, which are not interpreted by the server correctly.

**Ques 4:** Log into the administrator account!

Ans 4: 32a5e0f21372bcc1000a6088b93b458e41f0e02a

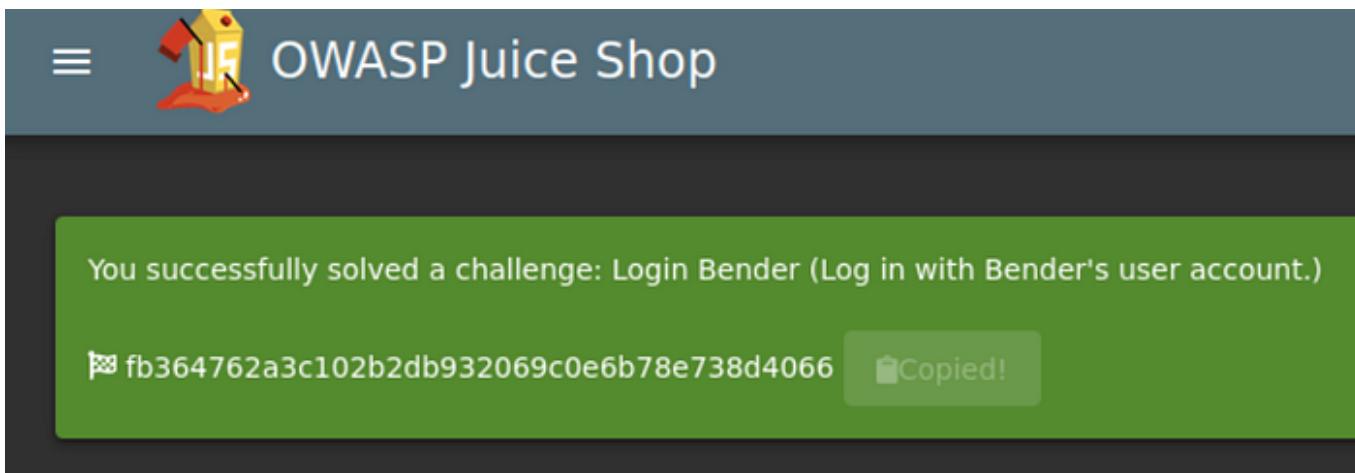
Press enter or click to view image in full size



**Ques 5:** Log into the Bender account!

Ans 5: fb364762a3c102b2db932069c0e6b78e738d4066

Press enter or click to view image in full size



*Who broke my lock?!*

Press enter or click to view image in full size



OWASP Juice Shop

≡

Account EN

Forgot Password

Email ?

Security Question ?

New Password  
• Password must be 5-20 characters long. 0/20

Repeat New Password  
0/20

Show password advice

 Change

In this task, we will look at exploiting authentication through different flaws. When talking about flaws within authentication, we include mechanisms that are vulnerable to manipulation. These mechanisms, listed below, are what we will be exploiting.

## Get Rahul Kumar's stories in your inbox

Join Medium for free to get updates from this writer.

Subscribe

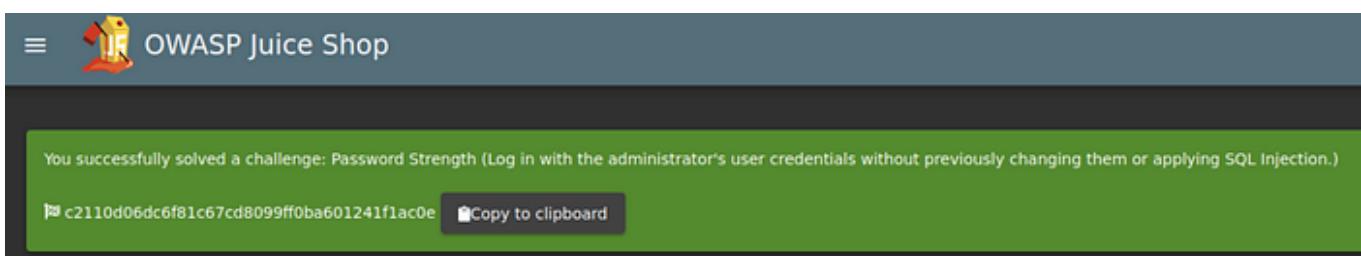
Weak passwords in high privileged accounts.

Forgotten password pages.

**Ques 6:** Bruteforce the Administrator account's password!

Ans 6: c2110d06dc6f81c67cd8099ff0ba601241f1ac0e

Press enter or click to view image in full size



**Ques 7:** Reset Jim's password!

Ans 7: 094fbc9b48e525150ba97d05b942bbf114987257

Press enter or click to view image in full size

The screenshot shows a web browser window for the OWASP Juice Shop application. The title bar says "OWASP Juice Shop". A green success message box contains the text: "You successfully solved a challenge: Reset Jim's Password (Reset Jim's password via the Forgot Password mechanism with the original answer to his security question.)". Below the message is a button labeled "Copy to clipboard".

*AH! Don't look!*

Press enter or click to view image in full size

The screenshot shows the "About Us" page of the OWASP Juice Shop application. The page has a dark header with "About Us" and a dark footer. The main content area contains a heading "Corporate History & Policy" followed by a large amount of placeholder text (Lorem ipsum) that includes a link to check out boring terms of use.

*A web application should store and transmit sensitive data safely and securely. But in some cases, the developer may not correctly protect their sensitive data, making it vulnerable.**Most of the time, data protection is not applied consistently across the web application making certain pages accessible to the public. Other times information is leaked to the public without the knowledge of the developer, making the web application vulnerable to an attack.***Ques 8:** Access the Confidential Document!

Ans 8: edf9281222395a1c5fee9b89e32175f1ccf50c5b

**Ques 9:** Log into MC SafeSearch's account!

Ans 9: 66bdcffad9e698fd534003fbb3cc7e2b7b55d7f0

Press enter or click to view image in full size



OWASP Juice Shop

You successfully solved a challenge: Login MC SafeSearch (Log in with MC SafeSearch's original user credentials without applying SQL Injection or any other bypass.)

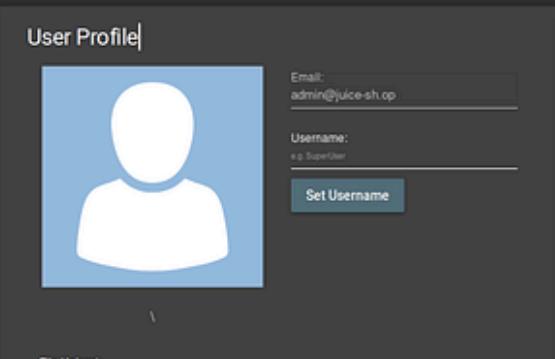
66bdcffad9e698fd534003fbb3cc7e2b7b55d7f0 [Copy to clipboard](#)

### Ques 10: Download the Backup file!

Ans 10: bfc1e6b4a16579e85e06fee4c36ff8c02fb13795

*Who's flying this thing?*

Press enter or click to view image in full size



User Profile

Email: admin@juice-sh.op

Username: e.g. SuperUser

Set Username

File Upload:

Browse... No file selected.

Upload Picture

or

Image URL:

e.g. http://www.practicalsecurity.info/images/5/5d/Tee-Cloud-Trust-75x872

Link Image

*Modern-day systems will allow for multiple users to have access to different pages.*

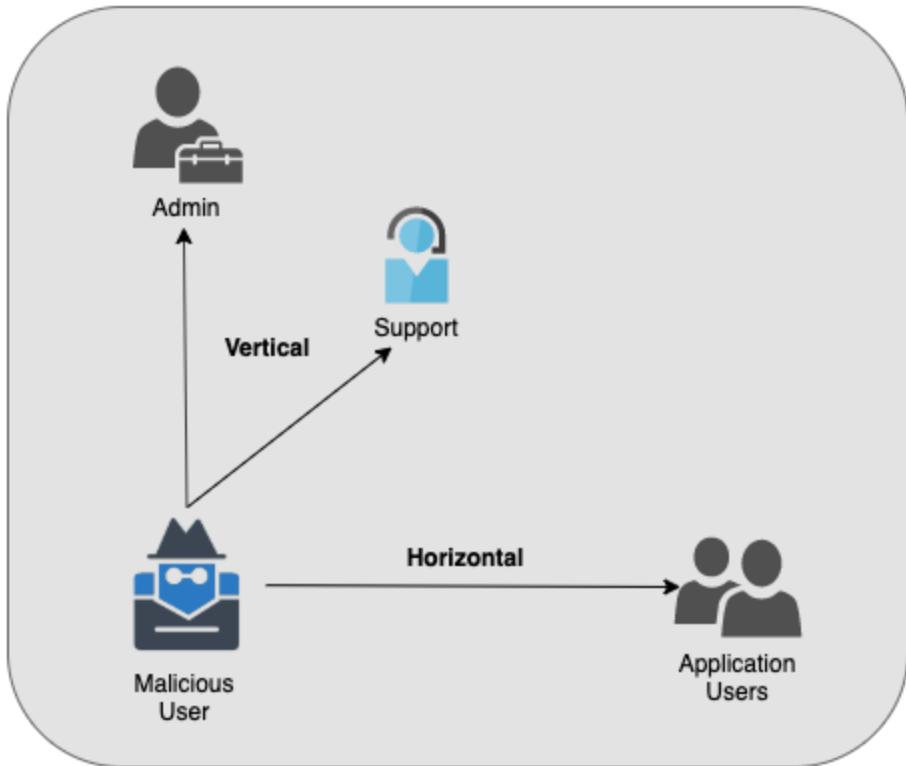
*Administrators most commonly use an administration page to edit, add and remove different elements of a website. You might use these when you are building a website with programs such as Weebly or Wix.*

*When Broken Access Control exploits or bugs are found, it will be categorized into one of two types:*

Press enter or click to view image in full size

Horizontal Privilege Escalation	Occurs when a user can perform an action or access data of another user with the same level of permissions.
Vertical Privilege Escalation	Occurs when a user can perform an action or access data of another user with a higher level of permissions.

# Broken Access Control



**Ques 11:** Access the administration page!

Ans 11: 946a799363226a24822008503f5d1324536629a0

**Ques 12:** View another user's shopping basket!

Ans 12: 41b997a36cc33fbe4f0ba018474e19ae5ce52121

**Ques 13:** Remove all 5-star reviews!

Ans 13: 50c97bcce0b895e446d61c83a21df371ac2266ef

*Where did that come from?*

Press enter or click to view image in full size

The screenshot shows the 'Order History' section of the OWASP Juice Shop application. It displays two completed orders and one order in transit. Each order row includes the order ID, total price, bonus, delivery status, product details, and quantity.

Order ID	Total Price	Bonus	Delivered
#5267-f73dcd000abcc353	26.97€		
Product	Price	Quantity	Total Price
Eggfruit Juice (500ml)	8.99€	3	26.97€
Order ID	Total Price	Bonus	In Transit
#5267-8701a2b3a7333cd9	8.96€		
Product	Price	Quantity	Total Price
Apple Juice (1000ml)	1.99€	3	5.97€
Orange Juice (1000ml)	2.99€	1	2.99€

XSS or Cross-site scripting is a vulnerability that allows attackers to run javascript in web applications. These are one of the most found bugs in web applications. Their complexity ranges from easy to extremely hard, as each web application parses the queries in a different way.

### ***There are three major types of XSS attacks:***

Press enter or click to view image in full size

<u>DOM (Special)</u>	DOM XSS ( <i>Document Object Model-based Cross-site Scripting</i> ) uses the HTML environment to execute malicious javascript. This type of attack commonly uses the <script></script> HTML tag.
<u>Persistent (Server-side)</u>	Persistent XSS is javascript that is run when the server loads the page containing it. These can occur when the server does not sanitise the user data when it is uploaded to a page. These are commonly found on blog posts.
<u>Reflected (Client-side)</u>	Reflected XSS is javascript that is run on the client-side end of the web application. These are most commonly found when the server doesn't sanitise search data.

### **Ques 14: Perform a DOM XSS!**

Ans 14: 9aaf4bbea5c30d00a1f5bbcfce4db6d4b0efe0bf

### **Ques 15: Perform a persistent XSS!**

|Ans 15: 149aa8ce13d7a4a8a931472308e269c94dc5f156

### **Ques 16: Perform a reflected XSS!**

Ans 16: 23cefef1527bde039295b2616eeb29e1edc660a0

*Exploration!*

Press enter or click to view image in full size

Score Board 1%					
1/11 1	0/10 2	0/22 3	0/24 4	0/17 5	0/11 6
<a href="#">Show all</a>		<a href="#">Show solved</a>	<a href="#">Show tutorials only</a>		
<a href="#">Broken Access Control</a>	<a href="#">Broken Anti Automation</a>	<a href="#">Broken Authentication</a>	<a href="#">Cryptographic Issues</a>	<a href="#">Improper Input Validation</a>	<a href="#">Injection</a>
<a href="#">Security Misconfiguration</a>	<a href="#">Security through Obscurity</a>	<a href="#">Sensitive Data Exposure</a>	<a href="#">Unvalidated Redirects</a>	<a href="#">Vulnerable Components</a>	<a href="#">XSS</a> <a href="#">XXE</a> <a href="#">Hide all</a>
Name	Difficulty	Description	Category		Status
Bonus Payload		Use the bonus payload <iframe width="100%" height="166" scrolling="no" frameborder="no" allow="autoplay" src="https://w.soundcloud.com/player/?url=https%3A//api.soundcloud.com/tracks/771984076&color=%23ff5500&auto_play=true&hide_related=false&show_comments=true&show_user=true&show_reposts=false&show_teaser=true"></iframe> in the DOM XSS challenge.	XSS		<input type="checkbox"/> unsolved
Confidential Document		Access a confidential document.	Sensitive Data Exposure		<input type="checkbox"/> unsolved
DOM XSS		Perform a DOM XSS attack with <iframe src="javascript:alert('xss')">.	XSS		<input type="checkbox"/> unsolved
Error Handling		Provoke an error that is neither very gracefully nor consistently handled.	Security Misconfiguration		<input type="checkbox"/> unsolved

If you wish to tackle some of the **harder** challenges that were not covered within this room, check out the [#/score-board](#) section on Juice-shop. Here you can see your completed tasks as well as other tasks in varying difficulty.

**Ques 17:** Access the /#/score-board/ page

Ans 17: 7efd3174f9dd5baa03a7882027f2824d2f72d86e

References: <https://tryhackme.com/room/owaspjuiceshop>