

# Practical demonstration - Main application hacking

Introduction Owasp Juice Shop Installation Judging our target Exploring the application Setting up burp suite Manually walking the application SQLi Testing for IDORs Testing for vertical priviledge escalation The login section CSRF Exploring the requests LFI/RFI OS Command injection

# Introduction

When we hunt, it's important to look at every target in it's own right. We are going to look at the OWASP juice shop. In this demonstration you will be show all the topic we went over and which parameters we will be using to test. Not all of our tests will lead to existing issues but still we **Have** to do all these tests. We are no longer practicing right now, this is bug bounties.

# **Owasp Juice Shop**

## Installation

Before we can hack on the OWASP juice shop, we ofcourse need to install it first. I chose to use herokuapp for this as it's free but you can also pick the docker image.

First navigate to <u>https://dashboard.heroku.com/apps</u> if you need to create an account, do that first. You will now see the option to create a new app, select this and pick the option to create a new app.

Next navigate to https://elements.heroku.com/buttons/bkimminich/juice-shop and click the "Deploy to heroku" button.



Pick a unique name for you app, it doesn't matter which it is but it has to be unique.

	Deploy your own	
	OWASP Juice Shop	
	Probably the most modern and sophisticated insecure web application	
	bkimminich/juice-shop#master	
App name		
thexssrat	uiceshop	
thexssratjui	c <b>eshop</b> is available	
Choose a r	egion	
Unit	ed States	\$

Click the "Deploy app" button, the app will now start deploying. It will take a while before the app is deployed so give it some take, sit back and make yourself a good coffee or tea with a nice piece of cheese.

	Create New App	
j	Deploy your own <u>OWASP Juice Shop</u> Probably the most modern and sophisticated insecure web application <u>bkimminich/juice-shop#master</u>	
App name		
thexssratjui	ceshop	0
hexssratjuice	shop is available	
hoose a reg	ion	
United	States	
Add to pip	eline	
Deploy ap	p	
Create app		$\oslash$
Configure er	nvironment	•••
Build app		
Run scripts 8	k scale dynos	
Deploy to He	eroku	

	Create New App	
J	Deploy your own <u>OWASP Juice Shop</u> Probably the most modern and sophisticated insecure web application <u>bkimminich/juice-shop#master</u>	
App name		
thexssratj	uiceshop	0
thexssratjui	ceshop is available	
Choose a r	egion	
Unit	ed States	
Add to p	ipeline	
Deploy a	арр	
Create app		$\odot$
Configure	environment	$\bigcirc$
Build app	Show build log	$\odot$
Run scripts	s & scale dynos	$\odot$
Deploy to	Heroku	Ø
	Your app was successfully deployed.	
	Manage App	

We can now view our app 😌

### Judging our target

We recognize this is a webshop so to hack this target properly we will have to make a small investment. Since this is a webshop we want to test for at least the following functions:

- Registration
- Login
- Buying an item
- Possibly returning an item
- Our wallet functionality if exists
- Logic flaws
- XSS
  - Stored
  - Reflected

- Basket functionality
- Adresses
- IDORs
- CSRF
- Broken access control if we can get to admin functions

This is an initial judgement and we might add to this as we explore the website and find more functionality.

## Exploring the application

First of all we need to know what functionality exists before we can start attacking our target properly. We need to fill up our site map in burp and we need to be able to epxlore the paramatirised requests. To do this, we need to set up burp properly first. This includes setting up our scope and setting the options that we need. In this course we will use burp suite but feel free to use any other MiTM proxy with the same functionality.

#### Setting up burp suite

It really helps to have burp suite pro, you don't have to but the fact that you can save a project is a major plus for me. I can only hunt in bursts of 1 to 3 hours so i have to revisit my target often. This means two things.

- I have to take very dilligent notes so i don't retest things 10 times needlesly and so that i make sure i do test all of my functionality. Part of this documentation is the "Judging our target" section.
- If u can set up my project settings in burp suite, save them and reload them whenever you want, that is a major plus. The biggest part of any activity is getting yourself to do it and if you can skip part of the setup, that will help you get started. If you are doing something time seems to fly but if you are sitting in your sofa it takes tremendous power to get yourself up and go hunting. Anything you can do to make this easier is a major win.

🔸 Bur	p Suite Professional v2021.2.1 - licensed to A	Avnu [single us	er license]		-	- 🗆 X
?	Welcome to Burp Suite Professional. Us	se the options	below to create or open a p	roject.	Burp Professi	o Suite
	Temporary project					
	O New project on disk	Name:	thexssratjuiceshop			
		File:	2021-03-10-thexssratjuicesho	op.burp		Choose file
	Open existing project		Name		File	
		File:				Choose file
			✓ Pause Automated Tasks		Cancel	Next

First i like to setup my scope, make sure you add the propper URLs that are in scope.

Burp	Project	Intruder	Repeater	Window	Help							
Dash	board	Target	Proxy	Intruder	Repeater	Sequencer	Decoder	Comparer	Extender	Project options	User options	Au
Site	map	Scope	Issue defin	itions								
<u>ې</u>	Target Define th menus in	Scope e in-scope the site m	targets for ap to include	your curren e or exclude	t work. This config e URL paths.	guration affects	the behavior o	f tools througho	ut the suite.	The easiest way to c	onfigure scope is to	o brov
	Use a	idvanced s	cope contro									
	Ado	<mark>1</mark> E	Inabled			Prefix						
	Edi	t										
	Remo	ove										
	Paste	URL										
	Load											
	Exclude f	rom scope										
	Ado	3	Enabled			Prefix						
	Edi	t										
	Remo	ove										
	Paste	URL										
	Load											

Hackerone has configuration files for burp you can download.



You can then import this file via the project options import functionality under "Project > Project options > Load Project options"

	Project	Intruder	Repeater	Window	Help	
1	Proje	ect options	>	Restore	defaults	>
I	Rena	ime		Load pr	oject option	s
	Save	сору		Save pr	oject options	;
	Impo	ort project f	file		the second second	
	Impo	ort legacy s	tate file	e or exclude	t work. This of URL paths.	cont

Dashb	ooard	Target	Proxy	Intruder	Repeater	Sequencer	Decoder	Comparer	Extender	Project options
Site m	пар	Scope	Issue de	finitions						
	Target Define t menus i Vse nclude	Scope he in-scope n the site n advanced in scope	e targets fo nap to incli scope cont	or your current ( ude or exclude ( trol	work. This configu URL paths.	uration affects th	ne behavior o	f tools through	out the suite	The easiest way to cc
(	Ac	ld .	Enabled	Protocol	Host /	IP range $\land$	Port		File	
(	Ec	lit	$\checkmark$	Any	thexssratjuices	hop.herokuapp.	c			
(	Rem	ove								•
(	Paste	URL								
(	Loa	d								
E	Exclude	from scop	e	Destand		(10	Dest		<b>5</b> 3-	
l	AC		Enabled	Protocol	Host /	/ IP range	Port		File	

After setting up our scope we will move on to setting up our proxy options. I always configure several options to make it easier for myself to see things like hidden fields and to remove any javascript validation from my responses.

Dashbo	bard	Target	Proxy	Intruder	Repeater	Sequencer
Interce	pt	HTTP history	WebS	ockets history	Options	
		ept server to	, eneme mes	50905		
? R	espon	se Modific	ation			
€j} ™	iese se	ttings are use	ed to perfo	rm automatic r	nodification of	f responses.
~	Unhio	de hidden for	<mark>m fiel</mark> ds			
		rominently h	ighlight unh	idden fields		
~	Enab	le disabled fo	orm fields			
~	2 Remo	ove input field	l length lim	its		
~	2 Remo	ove JavaScrip	t form valid	ation		
	Remo	ove all JavaSo	ript			
	Remo	ove <object></object>	tags			
	Conv	ert HTTPS lin	ks to HTTP			
[	Remo	ove secure fla	iq from coo	kies		

This will make any hidden fields easier to see as it will unhide them and draw a big red square around them. If you see this big red square you know you are looking at a hidden field.

Hidden field	[it[sig]	Ģ
Google	test	

Now that we have burp suite set up in the background we can start exploring our application.

#### Manually walking the application

Just because we are manually walking the application does not mean we should not be hacking. This is the most important phase in bug bounties and most of you will know it as the recon phase.

In this phase we want to get to know our application. We want to start by exploring the functionality and as we do that we want to take note of our prvilidge levels. Even though it might not seem like it, since we don't have access to the admin functionality (yet), but there are different levels of priviledges.

- Unauthenticated accounts (not logged in)
- Authenticated accounts

As we hack our application, there might be more levels we can add to this such as administrators.

I myself use excel to make a quick mindmap but you can use whatever tool suits you best.

	Authenticated	Unauthenticated	Admin??
View items			?
Add items to cart			?
Buy items			?
View reviews			?
Add reviews to items			?
	?	?	?

When i register my account, I register using an attack vector that automatically tests for JS XSS, HTML injection and HTML tag attribute injection.



I use this attack vector wherever possible when attacking my target.

User Profile	
Email: test@gmail.com Username: "> <u>THE XSS RAT WAS HERE Set Username</u>	

This way, whenever the application uses my username anywhere, i am automatically testing for all of the described attack vectors. If my username is reflected in the JS context anywhere, the " will try to break out of the context it is being reflected into. If the username is reflected in a tag, like it is in the picture above. In this picture, when we save the username, we can see that we have broken out of the VALUE attribute of the input tag. From here, we can try to insert our own javascript.

<input class="form-control mdl-textfield\_\_input" id="usernam
e" type="text" name="username" value="'">THE XSS RAT WAS HER
E" style="color: #FFFFF;" placeholder="e.g. SuperUser" arialabel="Text field for the username"> == \$0

I will try this WHEREVER i can, this includes adresses, nicknames, ...

On the profile page however i can still see some more things i can test for. I see a profile picture so i can test for XXE via SVG here.

File Upload:
Bestand kiezen xxe.svg
Upload Picture
or
Image URL:
e.g. https://www.gravatar.com/avatar/1aedb8d9dc4751e229a335e371db8058
Link Image

I also see a link option here. This link will resolve to a picture which gives the option for SSRF. To test for this, i start my burp collaborator and grab a URL that i can insert into this field. Running a public burp collaborator server is a premium option and only available in the paid version of burp. If you don't have the paid version of burp you can use:

- Your own webserver
  - If you don't configure this properly, you can only capture HTTP requests
- A public burp collaborator
  - This may suffer from availability issues as it's shared between all users

Whichever option we go for, we need to copy our payload to the clipboard and paste it in our URL field that the server tries to resolve.

							_			
🗲 Bu	ırp Collaborator clien	t				-		×		
?	Click "Copy to clip appear below.	oboard" to generate Bur	p Collaborator pa	yloads that you can use in your own	testing. Any interactions the	t result from using the payload	ds will			
	Generate Colla	borator payloads								
	Number to generate: 1 Copy to clipboard I locude Collaborator server location									
	Poll Collaborat	seconds Poll	now							
	# ^	Time	Туре	Payload	Comment					
	Click Copy to clipboard to generate sup Consolution payloads that you can use in your own testing. Any interactions that result non using the payloads will appear below.         Generate Collaborator payloads         Number to generate: <ul> <li>Copy to clipboard</li> <li>Include Collaborator server location</li> </ul> Poll Collaborator interactions <ul> <li>Poll every 60</li> <li>seconds</li> <li>Poll now</li> <li> <ul> <li>Time</li> <li>Type</li> <li>Payload</li> <li>Comment</li> </ul></li></ul>									
				888						

We might need to put HTTP:// in front of our URL if the server checks for syntax.

http://s5yf4ljmn9wgf95dcykd4iu7zy5otd.burpcollaborator.net

User Profile	
Department of the second secon	Email: test@gmail.com
\"'>THE XSS RAT WAS HERE	Username: ">THE XSS BAT WAS HERE
File Upload:	
Bestand kiezen Geen bestand gekozen	Set Username
Upload Picture	
or	
Image URL:	
http://s5yf4ljmn9wgf95dcykd4iu7zy5otd.burpcollaborat	
Link Image	

If you a lot of DNS requests coming into your burp collaborator but no HTTP requests, than there probably is no way to pull off SSRF is there is a possible egress filter in place. Egress filters can stop certain types of outgoing traffic. As of this writing OWASP juice shop does not have any SSRF vulnerabilities but if we'd find one here, we would continue on our SSRF path.

🔸 Bu	rp Colla	borator client				-		×			
?	Click " appea	from using the pay	loads will								
	Gene Numb	erate Collaborator pay	ation								
	Poll Collaborator interactions										
	Poll every 60 Poll now										
	# ^	Time	Туре	Payload	Comment						
	1	2021-Mar-10 20:19:51	UTC DNS	s5yf4ljmn9wgf95dcykd4iu7zy5otd							
	2	2021-Mar-10 20:19:51	UTC DNS	s5yf4ljmn9wgf95dcykd4iu7zy5otd							
	3	2021-Mar-10 20:19:51	UTC DNS	s5yf4ljmn9wgf95dcykd4iu7zy5otd							
	4	2021-Mar-10 20:19:51	UTC DNS	s5yf4ljmn9wgf95dcykd4iu7zy5otd							
	5	2021-Mar-10 20:19:51	UTC HTTP	s5yf4ljmn9wgf95dcykd4iu7zy5otd							
	Des	cription Request to	Collaborator Res	nonse from Collaborator							
	The Collaborator server received an HTTP request. The request was received from IP address 34.242.128.128 at 2021-Mar-10 20:19:51 UTC.										
Му	save	ed addresses									
"> HE	≪u>TH ERE	E XSS RAT WAS "> ">	- <u>THE XSS RAT WA -<u>THE XSS RAT WA</u></u>	IS HERE, "'> <u>THE XSS RAT WAS HERE, IS HERE, 13265</u>	"> <u>THE XSS RAT WAS HERE</u>	Ľ	Ŵ				

+ Add New Address

I will try the same XSS technique for adress any stored or reflected fields i can find. If i suspect a server side template engine is being used, i will add an SSTI attack vector.

'"><u>THE XSS RAT WAS HERE\${7'\*7'}

if this resolves to 7777777 or 49 i will investigate further into SSTI or CSTI attack techniques.

#### SQLi

The previous attack vector also automically tests for SQLi as well

...

These special characters will also test for SQLi since these are special characters also used in SQL statements. If we get a SQL error, i usually run a tool like SQLmap to better investigate what SQLi i have on my hands if any. See SQLi section.

#### **Testing for IDORs**

To test for IDOR's i will have to create a second account. In this case i can test for IDORs in several ways. First of all i really need that second account. I am going to copy his session cookie or authorisation header. This may seem confusing as you can't copy your victims cookie or headers in production **but** this is not our goal. We just need that cookie or header to automate our IDOR search. How to do this has been demonstrated in the tools section.

#### Testing for vertical priviledge escalation

When we want to test for vertical priviledge escalation we do need accounts of different priviledge levels. Since we don't have any admin accounts yet, we can't test for this yet. If we did have different priviledge levels, we would create accounts of all different priviledge levels and test for BAC. Example:

- Administrator
- · Content editor
- Customers

Again, how to test for this specifically has been illustrated in the tools section.

#### The login section

In the login section we can play around with the requests a little bit. For example if we request a password reset link, and append our own email adress to the request, the server might send the password reset link of the victim to the email adress of the attacker.



#### Might turn into

```
POST /api/resetPassword HTTP/1.1
Host: ferretshop.herokuapp.com
Connection: close
sec-ch-ua: ";Not A Brand";v="99", "Chromium";v="88"
Accept: application/json, text/plain, */*
Authorization: Bearer eyJ0eXAi0iJKV1QiLCJhbGci0iJSUz11NiJ9.eyJzdGF0dXMi0iJzdWNjZXNzIiwiZGF0YSI6eyJpZCI6MTcsInVzZXJuYW1lljoiIiwiZW1haWwi0iJ0
sec-ch-ua-mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/88.0.4324.150 Safari/537.36
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: cors
Sec-Fetch-Dest: empty
Referer: https://ferretshop.herokuapp.com/
Accept-Encoding: gzip, deflate
Accept-Language: nl-NL, nl; g=0.9, en-US; g=0.8, en; g=0.7
Cookie: language=en; welcomebanner_status=dismiss; token=eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJzdGF0dXMiOiJzdWNjZXNzIiwiZGF0YSI6eyJpZCI6M
Content-Length: 24
{email:"victim@gmail.com"
email:"attacker@gmail.com"}
```

This might prompt the "GenerateLink" server to generate a password reset link for the first email address but the "SendLink" server might send the link to the attacker.

#### CSRF

Whenever i see a CSRF token, i will try to replace it:

- with an empty parameter (CSRF=)
- with a parameter of the same restrictions (like same length and alphanumeric) (CSRF=2475455dfs1)
- CSRF=1
- A CSRF token that does not belong to that account

I can use the tools "Match and replace" or "Autorepeater" for this although the "Autorepeater" extensions seems to have broken with the latets burp update though this might get fixed later on.

See the tools section.

#### **Exploring the requests**

Now comes the fun part, we are going to look at all the requests and parameters in their own right. To do this we need to go back to burp suite and look at our site map. This has been filling up in the background while we click around.



Now we can see the parameterised requests, we don't really care about the static requests.



I am going to look at all of these requests and see if i can find some parameters i can manipulate that i should not be able to manipulate such as:

- {userType:"User"} > {userType:"Admin"}
- {accountType:"Basic"} > {accountType:"Advanced"} (might be more expensive)
- {rating:5} > {rating:-5000} (might be able to negatively affect the rating of a video on youtube)
- ... Use your imagination

Burp	Project	Intruder	Repeater	Window H	Help			
Das	nboard	Target	Proxy	Intruder	Repeater	Sequencer	Decoder	Compar
1 ×	2 ×	3 ×						
S	end	Cancel	<   • )	>   •				
Req	uest							1
Prett	v Raw	\n Actio	ns 🗸					
1 1	T /rest	/product	e/6/romi	owe WTTD/1	1			
2 H	st: fer	retshop.	herokuan	ews mir/i p.com	• 1			
3 Co	nnectio	n: close	ince on adap	proom				
4 Co	ntent-L	ength: 5	4					
5 56	c-ch-ua	: ";Not	A Brand"	;v="99", "	Chromium";v	-"88"		
6 Ac	cept: a	pplicati	.on/json,	text/plai	n, */*			
7 A.	thoriza	tion: Be	arer eyJ	OeXAiOiJKV	lQiLCJhbGci	OiJSUzI1NiJ	9.eyJzdGFO	aXMiOi.
DI	XLTAZLT	EwIDIwOj	AzOjEyLj	c5MiArMDA6	MDAiLCJkZWx	ldGVkQXQiOm	51bGx9LCJp3	(XQiOj]
8 56	c-ch-ua	-mobile:	20					
9 08	er-Agen	t: Mozil	1a/5.0 (	Windows NI	10.0; Wine	4; x64) App	lewebKit/53	37.36
11 0	igin, h	ype: app	erretebo	/ Json n borokuan				
12 56	c-Fetch	_Site e	ame_orig	p.nerokuap in	p.com			
13 56	c-Fetch	-Mode: c	ors	111				
14 56	c-Fetch	-Dest: e	mptv					
15 Re	ferer:	https://	ferretsh	op.herokua	pp.com/			
16 Ac	cept-En	coding:	gzip, de	flate				
17 Ac	cept-La	nguage :	nl-NL,nl	;q=0.9,en-	US;q=0.8,en	ı;q=0.7		
18 Co	okie: l	anguage=	en; welc	omebanner_	status=dism	iss; token=	eyJOeXAiOiJ	JKV1Qil
C5	MiArMDA	6MDAiLCJ	lcGRhdGV	kQXQiOiIyM	DIXLTAZLTEW	IDIWOjAzOjE	yLjc5MiArMI	DA6MDA:
19								
20 {								
	"messag	e":"frdf	gdfgfdgd	fg",				
	author	":"test@	gmail.co	m''				
,								

In this case for example, we can change the author of a review which should not be possible.

#### LFI/RFI

Whenever a parameters shows that it is grabbing a file from the local file system or whenever it seems to be from a remote location, i will try either LFI or RFI respectively. See those sections to learn more about them.

- GET /avatar.php?file=image1.png
- GET /avatar.php?file=s3.bucket.org/image1.png

LFI/RFI in and off- itself is usually not that impactful. If we find this issue we should try to find files on the system that we should not be able to access like private pictures of other users or we should try to include a file which executes remote code execution. This can cause use to create a reverse shell allowing us access on the server. If you ever achieve this, you should stop and report, don't explore a production server with the risk of seeing data you should not or crashing the production server.

#### **OS Command injection**

This is the last vulnerability type i check for. The only way to check for this issue type is to fuzz all of those parameters with your fuzzing list that you created in the OS command injection section. To do this I use the intruder tool that's built into burp.

lurp Project Int	ruder Repeater	Window H	lelp								
Dashboard T	arget Proxy	Intruder	Repeater	Sequencer	Decoder	Comparer	Extender	Project options	User options	Autorize	AutoRepeater
1 × 2 ×											
Target Positio	ons Payloads	Options									
? Payload Se	ts										Start attack
You can defir and each pay	e one or more pa load type can be o	yload sets. Th customized in	e number of p different ways	ayload sets dep	ends on the a	ttack type define	ed in the Positi	ons tab. Various pay	load types are ava	ilable for each	i payload set,
Payload set:	1	~	Payload co	ount: 169							
Payload type	Simple list	~	Request co	ount: 1,859							
Paste	{base}))))))))))										
This payload	type lets you confi	igure a simple	list of strings	that are used as	payloads.						
Paste	{base}   ping -i 3	0 127.0.0.1 ; x	ping -n 30 1	27.0.0							
Load	{base}  ping -i 30	) 127.0.0.1									
Remove	{base}  ping -n 3	0 127.0.0.1		<b> </b>							
Clear	{base}& ping -i a	30 127.0.0.1 &		7							
	{base}& ping -n	107001									
	{base}; ping -c 5	-i 30 127 0 0 1	%0a								
Add	Enter a new item										
Add from lis	it			~							