Burp Suite User Manual

Penetration Testing Report By: Mahnoor Intizar August 5, 2024

Table of Contents

Table of	f Contents	
1. Burp	Suite Overview	
1.1.	What is Burp Suite?	4
1.2.	Editions of Burp Suite	4
1.3.	Use Cases	4
1.4.	Download and Installation Steps	5
2.	Intercept HTTP traffic with Burp Proxy	5
2.1.	Launch Burp's browser	5
2.2.	Intercept a request	6
2.3.	Forward the request	6
2.4.	Switch off interception	6
2.5.	View the HTTP history	7
3.	Modifying HTTP requests with Burp Proxy	8
3.1	Access the vulnerable website in Burp's browser	8
3.2	Log in to your shopping account	9
3.3	Find something to buy	9
3.4	Study the add to cart function	9
3.5	Modify the request	
3.6	Exploit the vulnerability	
4	Set the target scope	
4.1	Launch Burp's browser	
4.2	Browse the target site	
4.3	Study the HTTP history	11
4.4	Set the target scope	11
4.5	Filter HTTP history	
5	Reissue requests with Burp Repeater	
5.1	Sending a request to Burp Repeater	
5.2	Identify an interesting request	
5.3	Send the request to Burp Repeater	14
5.4	Send the request and view the response	

5.5	Testing different input with Burp Repeater	15
5.6	Resend the request with different input	15
5.7	View the request history	16
5.8	Try sending unexpected input	17
5.9	Study the response	17
6	Run your first scan	18
6.1	Open the scan launcher	19
6.2	Enter the URL of the target site	19
6.3	Configure the scan	20
6.4	Launch the scan	21
6.5	See the crawl in action	22
6.6	View the identified issues	23

1. Burp Suite Overview

1.1. What is Burp Suite?

Burp Suite is a comprehensive platform for performing security testing of web applications. It includes a variety of tools with unique functionalities that work together seamlessly to support the entire testing process, from initial mapping and analysis of an application's attack surface to finding and exploiting security vulnerabilities.

1.2. Editions of Burp Suite

Community Edition: This is the free version of Burp Suite. It provides essential manual tools, such as the Proxy, Intruder, Repeater, and Decoder. However, it lacks the automated scanning and some advanced features found in the Professional and Enterprise editions.

Professional Edition: A paid version that offers additional features, including the automated scanner, advanced manual tools, and support for extensions. This edition is aimed at security professionals who need to perform more comprehensive and efficient testing.

Enterprise Edition: This edition is designed for organizations needing to scale their security testing across many applications. It offers continuous, automated scanning and integrates with CI/CD pipelines, making it suitable for large enterprises.

1.3. Use Cases

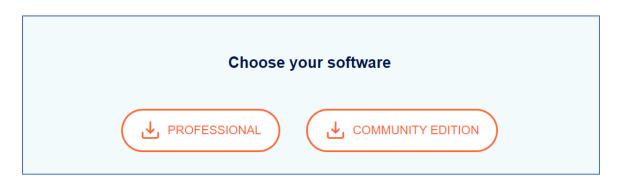
Penetration Testing: Burp Suite is widely used by penetration testers to identify and exploit vulnerabilities in web applications.

Security Audits: Organizations use Burp Suite for regular security audits to ensure their applications remain secure over time.

Bug Bounty Hunting: Security researchers and bug bounty hunters use Burp Suite to find and report vulnerabilities in web applications for rewards.

1.4. Download and Installation Steps

First of all, Visit the official Burp Suite website. Download the appropriate version (Community or Professional).



Run the installer and launch Burp Suite. When asked to select a project file and configuration, just click **Next** and then Start Burp to skip this for now.

2. Intercept HTTP traffic with Burp Proxy

Burp Proxy lets you intercept HTTP requests and responses sent between Burp's browser and the target server. This enables you to study how the website behaves when you perform different actions.

2.1. Launch Burp's browser

Go to the Proxy > Intercept tab.

Set the intercept toggle to Intercept on.

Dashboard	Target	Proxy	Intruder	Repeater	Collaborator	Sequencer	Decoder	Comparer	Logger
Intercept	HTTP history	y Web	Sockets hist	tory දිදි	Proxy settings				
						_			
Intercontended	cept on	→ 1	orward	~]	Drop]			

Click Open Browser. This launches Burp's browser, which is preconfigured to work with Burp right out of the box.

Position the windows so that you can see both Burp and Burp's browser.

2.2. Intercept a request

Using Burp's browser, try to visit **https://portswigger.net** and observe that the site doesn't load. Burp Proxy has intercepted the HTTP request that was issued by the browser before it could reach the server. You can see this intercepted request on the Proxy > Intercept tab.

Dashboard	Target	Proxy	Intruder	Rep	peater Collabo	orator	Sequencer	Decoder	Comparer	Logge
Intercept	HTTP histor	/ We	bSockets h	istory	Proxy sett	ings				
💿 Inte	ercept on	\rightarrow	Forward		Drop					
Time	Тур	be		Directior	ı	Host			Metho	d
09:42:32 3 Ju	1 2024 HT	TP		→ Rea	uest	norts	wigger.net		GET	
				, 104	0001	ports	nggornor			
Request						porto				
Request Pretty R	aw Hex					porte				
Pretty R	HTTP/1.1					porto				
Pretty R 1 GET / H 2 Host: J	HTTP/1.1 portswigg				2C%2022%20No	·				

The request is held here so that you can study it, and even modify it, before forwarding it to the target server.

2.3. Forward the request

Click the Forward button to send the intercepted request. Click Forward again to send any subsequent requests that are intercepted, until the page loads in Burp's browser. The Forward button sends all the selected requests.

2.4. Switch off interception

Due to the number of requests browsers typically send, you often won't want to intercept every single one of them. Set the intercept toggle to Intercept off.

Dashboard	Target	Proxy	Intruder	Repeater	Collaborate	or Se	equencer	Decoder
Intercept	HTTP history	WebS	Sockets hist	tory (බු	Proxy settings	6		
© Inte	ercept off) > F	orward	<u>~</u>	Drop			
Time	Тур	e	Di	irection		Host		

Go back to the browser and confirm that you can now interact with the site as normal.

2.5. View the HTTP history

In Burp, go to the Proxy > HTTP history tab. Here, you can see the history of all HTTP traffic that has passed through Burp Proxy, even while intercept was switched off.

Click on any entry in the history to view the raw HTTP request, along with the corresponding response from the server.

Das	shboard Target Prox	y Intruder	Repeate	er Collaborator	Sequencer	Decode	er Compare	er Log	ger Exter	nsions (Organizer
Inte	rcept HTTP history	WebSockets his	story	🔅 Proxy settings							
∇	Filter settings: Hiding CSS, in	mage and genera	al binary cor	ntent							
#~	Host	Method	Params L	JRL		Edited	Status code	Length	MIME type	Extension	Title
46	https://portswigger.net	GET	/	content/images/svg/a	rrow-youtube.		200	1901	XML	svg	
45	https://portswigger.net	GET	/	content/images/patter	ns/dots-spac.		200	9553	XML	svg	
30	https://portswigger.net	GET	/	content/images/logos/	/portswigger-r.		200	11444	XML	svg	
29	https://portswigger.net	GET	/	content/images/logos/	/portswigger		200	7190	XML	svg	
27	https://portswigger.net	GET	/	images/validate-your-	certification.s.		200	35113	text	svg	
26	https://portewigger.net	GET	/	imanae/raeaarch_ema	lleva		200	12301	tevt	eva	
Pret	,			ର 🗐 ଏମ		,		lender			🗐 \n
						TTP/2 2		(c) loci			•••
	GET /content/images, lost: portswigger.ne		yourube	3vg 1111/2		,	d, 03 Jul :	2024 08	:49:07 GM	1T	
	Cookie: stg returni		=				Type: imag				
W	ved%2C%2022%20Nov%2	02023%2009;	06:36%20	0GMT; t=	4 (Content-I	Length: 36	1			
		1001/49.309.30	ALICAL I								
H	IIRDfA007iUBE1hC%2B					Server:					
H	WSALBAPP-1=_remove	; AWSALBAP			6	ccept-R	anges: byt				
H A A	AWSALBAPP-1=_remove	_; AWSALBAF _;	PP-2=_rer	nove_;	6	ccept-Rache-Co	anges: byt ntrol: mus	t-reval	idate, ma	ax-age=0	
H A A	WSALBAPP-1=_remove WSALBAPP-3=_remove _pk_id.287552c2-491	_; AWSALBAF _; 7-42e0-8982	PP-2=_rer 2-ba994a2	nove_; 2a73d7.1467=	6 7 8	Accept-Ra Cache-Con tag: "10	anges: byt ntrol: mus dacbb15796	t–reval c669"		5	смт
H A A	<pre>\WSALBAPP-1=_remove \WSALBAPP-3=_remove _pk_id.287552c2-491 l2cb2d3d7a58eff5.17</pre>	_; AWSALBAF _; 7-42e0-8982 00643910.7.	PP-2=_rer 2-ba994a2	nove_; 2a73d7.1467=	6 / 7 (8 • ; 9	Accept-Ra Cache-Con tag: "10 .ast-Mod	anges: byt ntrol: mus dacbb15796 ified: Mon	t–reval c669" , 01 Ju	ıl 2024 12	2:22:30 (
H A d s	WSALBAPP-1=_remove WSALBAPP-3=_remove _pk_id.287552c2-491	_; AWSALBAF _; 7-42e0-8982 00643910.7. n=	PP-2=_rer 2-ba994a 17195784	nove_; 2a73d7.1467= 486.1719578486	6 / 7 (8 9 10	Accept-Ra Cache-Con tag: "10 Last-Mod Strict-T	anges: byt ntrol: mus dacbb15796 ified: Mon	t-reval c669" , 01 Ju ecurity	ıl 2024 12 : max—age	2:22:30 (GMT 00; preload

This lets you explore the website as normal and study the interactions between Burp's browser and the server afterward, which is more convenient in many cases.

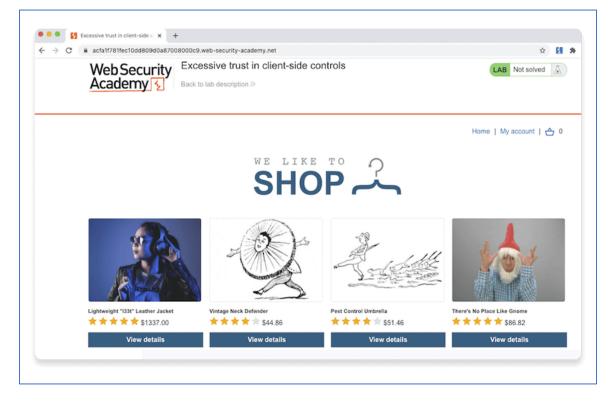
3. Modifying HTTP requests with Burp Proxy

This enables you to manipulate these requests in ways that the website isn't expecting, in order to see how it responds. Using one of our deliberately vulnerable websites, known as "labs", you'll see how this can help you identify and exploit real vulnerabilities.

3.1 Access the vulnerable website in Burp's browser

In Burp, go to the Proxy > Intercept tab and make sure interception is switched off. Launch Burp's browser and use it to visit the following URL: https://portswigger.net/websecurity/logic-flaws/examples/lab-logic-flaws-excessive-trust-in-client-side-controls

When the page loads, click Access the lab. If prompted, log in to your portswigger.net account. After a few seconds, you will see your own instance of a fake shopping website.



3.2 Log in to your shopping account

On the shopping website, click My account and log in using the following credentials:

Username: wiener

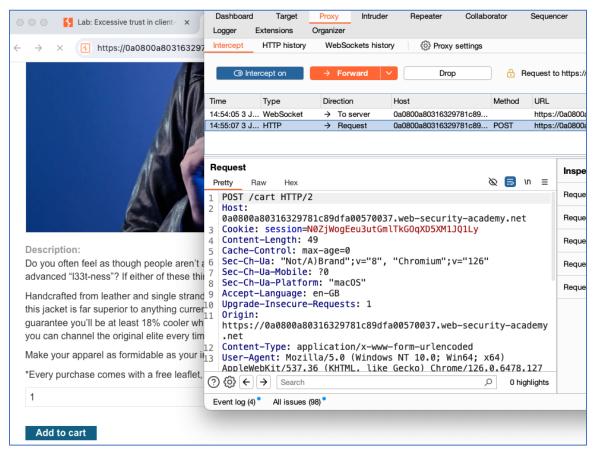
Password: peter

3.3 Find something to buy

Click Home to go back to the home page. Select the option to view the product details for the Lightweight "133t" leather jacket.

3.4 Study the add to cart function

In Burp, go to the Proxy > Intercept tab and switch interception on. In the browser, add the leather jacket to your cart to intercept the resulting POST /cart request.



Study the intercepted request and notice that there is a parameter in the body called price, which matches the price of the item in cents.

3.5 Modify the request

Change the value of the price parameter to 1 and click Forward > Forward all to send the modified request to the server, along with any other intercepted requests.

```
20
21 productId=1&redir=PRODUCT&quantity=1&price=1
```

Switch interception off again so that any subsequent requests can pass through Burp Proxy uninterrupted.

3.6 Exploit the vulnerability

In Burp's browser, click the basket icon in the upper-right corner to view your cart. Notice that the jacket has been added for just one cent.

Click the **Place order** button to purchase the jacket for an extremely reasonable price.

4 Set the target scope

The target scope tells Burp exactly which URLs and hosts you want to test. This enables you to filter out the noise generated by your browser and other sites, so you can focus on the traffic that you're interested in.

4.1 Launch Burp's browser

Launch Burp's browser and use it to visit the following URL: https://portswigger.net/web-security/information-disclosure/exploiting/lab-infoleak-in-error-messages

When the page loads, click Access the lab. If prompted, log in to your portswigger.net account. After a few seconds, you will see your own instance of a fake shopping website.

4.2 Browse the target site

In the browser, explore the site by clicking on a couple of the product pages.

4.3 Study the HTTP history

In Burp, go to the Proxy > HTTP history tab. To make this easier to read, keep clicking the header of the leftmost column (#) until the requests are sorted in descending order. This way, you can see the most recent requests at the top.

Dashb	ooard	Target	Proxy	Intr	uder	Repeater	Collaborator	Sequencer	Decoder	Comparer	Logger
Interc	ept	HTTP histo	ry Wel	Sock	ets history	y Option	s				
Filter: H	Hiding C	CSS, image a	and genera	l bina	ry content	t					
#~		Hos	st		Method	1			UF	۱L	
220	https:/	//0a2d00c90	34b19ffc04	400	GET	/academy	/LabHeader				
219	https:/	//0a2d00c90	34b19ffc04	400	GET	/					
218	https:/	/0a2d00c90	34b19ffc04	400	GET	/academy	/LabHeader				
217	https:/	//0a2d00c90	34b19ffc04	400	GET	/product?	productId=2				
215	https:/	/0a2d00c90	34b19ffc04	400	GET	/academy	/LabHeader				
214	https:/	//0a2d00c90	34b19ffc04	400	GET	/resource	s/labheader/ima	ges/ps-lab-notso	olved.svg		
213	https:/	/0a2d00c90	34b19ffc04	400	GET	/resource	s/labheader/ima	ges/logoAcaden	ny.svg		
212	https:/	/0a2d00c90	34b19ffc04	400	GET	/resource	s/images/shop.s	vg			
185	https:/	//0a2d00c90	34b19ffc04	400	GET	/resource	s/labheader/js/la	bHeader.js			
184	https:/	//0a2d00c90	34b19ffc04	400	GET	/resource	s/labheader/js/si	ubmitSolution.js			
183	https:/	//www.youtu	be.com		POST	/api/stats/	/atr?ns=yt⪙=em	bedded&cpn=-\	/oeLIKGDj7fH	hdR&ver=2&cm	nt=0&fs=0&rt=
181	https:/	/0a2d00c90	34b19ffc04	400	GET	1					
180	https:/	/portswigge	r.net		GET	/academy	/labs/launch/874	3ae75bedd9ef1	9ce2134472f	6df1c700ed51e	9a571f0b56a
179	https:/	/www.youtu	be.com		POST	/youtubei/	/v1/log_event?alt	=json&key=Alza	SyAO_FJ2Slq	U8Q4STEHLGO	Cilw_Y9_11qc
178	https:/	/portswigge	r.net		GET	/content/i	mages/svg/ps-lo	go-lines-white.s	vg		
177	https:/	/jnn-pa.goog	gleapis.cor	n	POST	/\$rpc/goo	gle.internal.waa.	v1.Waa/Genera	teIT		
176	https:/	/www.youtu	be.com		POST	/youtubei/	/v1/log_event?alt	=json&key=Alza	SyAO_FJ2SIq	U8Q4STEHLGO	Cilw_Y9_11qc
175	https:/	/www.youtu	be.com		GET	/generate	_204?uG_lZg				
174	https:/	/www.gstati	c.com		GET	/cv/js/sen	der/v1/cast_sen	der.js			
171	https:/	/www.youtu	be.com		GET	/s/player/	7a062b77/player	_ias.vflset/en_G	B/embed.js		
170	https:/	/www.goog	e.com		GET	/is/th/RLc	wZH2Xcwti3dY	vGSeKf8RcILu2	Ri3JTO2BWy	vP7U.is	

Notice that the HTTP history shows details about each request that the browser has made, including requests to third-party websites that you're not interested in, such as YouTube and Google Analytics.

4.4 Set the target scope

Go to Target > Site map. In the left-hand panel you can see a list of hosts that your browser has interacted with.

Right-click on the node for the target site and click Add to scope. When prompted in a pop-up window, click Yes to exclude out-of-scope traffic.

Dashboard	Target	Proxy	Intruder	Repeater	Collaborator	Sequencer	Decoder	Comparer	Logger
Site map	Scope	Issue defir	nitions						
Filter: Hiding n	not found ite	ms; hiding	CSS, image	and general bi	nary content; hid	ing 4xx respons	es; hiding em	pty folders	
https://0	a2d00c903	34b19ffc040	074900a400	39.web-securi	tv-academv.net	Contonte			
					2d00c9034b19ffc		security-acad	lemy.net/	
https://g			k.net	Add to sco	pe				0039.web-sec
https://jr				Scan					0039.web-sec
All https://p https://s				Passively s	can this host				0039.web-sec
https://v			om	Actively sc	an this host				0039.web-sec
https://v	0 0	,		Extensions				\	0039.web-sec
https://v			r.com						0039.web-sec
https://v	www.gstatic	.com	_	Engageme					0039.web-sec
http://w	ww.youtube	e.com		Compare s	ite maps				0039.web-sec
https://v	www.youtub	e.com		Expand bra	anch				0039.web-sec
				Expand red	quested items				0039.web-sec
				Delete hos					NO23.Web-sec

4.5 Filter HTTP history

Click on the display filter above the HTTP history and select Show only in-scope items.

Dasht	board Target Proxy Int	ruder	Repeater Colla	borator Sequence	r Decoder	Comparer	Logger	Extensions	Learn AT	OR v2.1	.0 bad	ckupFinder	r
Interc	ept HTTP history WebSock	kets histor	y Options										
ilter: H	Hiding CSS, image and general bina	ary conten	t										
#~	Host	Method	I .		ι	JRL			P	arams	Edited	Status	
20	https://0a2d00c9034b19ffc0400	GET	/academyLabHea	ader								101	14
19	https://0a2d00c9034b19ffc0400	GFT				E114 + + 1							10
18	https://0a2d00c9034b19ffc0400	G	•			Filter settin	gs						1.
17	https://0a2d00c9034b19ffc0400	G ?	Filter by request ty	pe		Filter by MIME ty	00		Filter by sta	tus code			4
15	https://0a2d00c9034b19ffc0400	G	Show only in-sc	ope items			V Oth	er text	Zxx [suc	cessi			1
14	https://0a2d00c9034b19ffc0400	G 💮								-			9
13	https://0a2d00c9034b19ffc0400	G	Hide items with	out responses		Script	🗌 Ima	ges	3xx [red	irection]			8
12	https://0a2d00c9034b19ffc0400	G	Show only para	meterized requests		🗹 XML	🗹 Flas	h	🗹 4xx [req	uest erro	r]		7
85	https://0a2d00c9034b19ffc0400	G				CSS	Oth	ər binarv	✓ 5xx [ser	ver error	1		8
B4	https://0a2d00c9034b19ffc0400	G					0.041	or bindiy					1
83	https://www.youtube.com	P	Filter by search ter	m	Filter by file ex	xtension		Filter by an	notation		Filter by li	stener	5
81	https://0a2d00c9034b19ffc0400	G											1
80	https://portswigger.net	G			Show only:	asp,aspx,jsp,p	hp	Show o	nly commented		Port		2
79	https://www.youtube.com	P	Regex		Hide:	js,gif,jpg,png,c		Show o	nly highlighted i		Pon		73
78	https://portswigger.net	G	Case sensitive	Negative search		ja,gii,jpg,piig,o	55		ing ing nou i				2
77	https://jnn-pa.googleapis.com	P											7.
76	https://www.youtube.com	P	Show all Hid	e all Revert chang	ges					Can	cel 🛛 🖊	pply	7
75	https://www.youtube.com	GE	/generate_2041u	a_izy						v		204	2
74	https://www.gstatic.com	GET	/cv/js/sender/v1/	cast_sender.js								200	5
71	https://www.youtube.com	GET	/s/player/7a062b	77/player_ias.vflset/er	_GB/embed.js							200	26

Scroll back through your HTTP history. Notice that it now only shows entries from the target website. All other entries have been hidden.

This greatly simplifies the history to only include items you're interested in. If you continue to browse the target site, notice that out-of-scope traffic is no longer logged in the site map or proxy history.

Dashb	board	Target	Proxy	Intr	uder	Repeater	Collaborator	Sequencer	Decoder	Comparer	Logger
Interc	ept	HTTP histor	ry Wel	Sock	ets histor	y Optior	IS				
Eilter: L	liding ou	it of coope	itomo, hid	na 00	C image	and gapara	I binon (content				
riiter: r	hiaing ou	ut of scope	items; nia	ing CS	o, image	and genera	I binary content				
# \sim		Hos	st		Method	ł			UF	۲L.	
220	https://	0a2d00c90	34b19ffc0	400	GET	/academ	yLabHeader				
219	https://	0a2d00c90	34b19ffc0	400	GET	1					
218	https://	0a2d00c90	34b19ffc0	400	GET	/academ	yLabHeader				
217	https://	0a2d00c90	34b19ffc0	400	GET	/product?	productId=2				
215	https://	0a2d00c90	34b19ffc0	400	GET	/academ	yLabHeader				
214	https://	0a2d00c90	34b19ffc0	400	GET	/resource	es/labheader/ima	ges/ps-lab-notso	olved.svg		
213	https://	0a2d00c90	34b19ffc0	400	GET	/resource	es/labheader/imag	ges/logoAcadem	ny.svg		
212	https://	0a2d00c90	34b19ffc0	400	GET	/resource	es/images/shop.s	vg			
185	https://	0a2d00c90	34b19ffc0	400	GET	/resource	es/labheader/js/la	bHeader.js			
184	https://	0a2d00c90	34b19ffc0	400	GET	/resource	es/labheader/js/su	ubmitSolution.js			
181	https://	0a2d00c90	34b19ffc0	400	GET	/					
					•						

5 Reissue requests with Burp Repeater

This lets you study the target website's response to different input without having to intercept the request each time. This makes it much simpler to probe for vulnerabilities, or confirm ones that were identified by Burp Scanner.

5.1 Sending a request to Burp Repeater

The most common way of using Burp Repeater is to send it a request from another of Burp's tools. In this example, we'll send a request from the HTTP history in Burp Proxy.

5.2 Identify an interesting request

In the previous tutorial, you browsed a fake shopping website. Notice that each time you accessed a product page, the browser sent a GET /product request with a productId query parameter.

Das	hboard	Targ	get P	roxy	Intruder	Repeater	Sequencer	Decod	er C	omparer	Lo
Inte	rcept	HTTP	history	WebSo	ckets histo	ory Option	5				
ilter	Hiding C	SS, ima	ge and ge	neral binary	/ content						
ŧ ~		Host		Method		URL		Params	Edited	Status	Len
<i>'</i>			1147 10	GEI	/ /					200	1004
6 5			o1f4713	GET		LabHeader		1		101 200	147 4242
5 4			o1f4713	GET		ProductId=3		~		101	4242
4 3			o1f4713	GET	/academy	/LabHeader				200	1064
3 2			o1f4713	GET	/	d ab Uaadax				200	147
∠ 1	https://a		o1f4713	GET		/LabHeader		./		200	147
			0.0000000000								
Req	uest							Respons	se		
Prett	y Raw	Hex	n 🔳					Pretty R	aw Hex	Render	\n
2 Ho	st:			=3 HTTP/ 19008800		ecurity-aca	demy.net	1 HTTP/1 2 Conter 3 Connec	t-Type:	text/ht	tml; c

Let's use Burp Repeater to look at this behavior more closely.

5.3 Send the request to Burp Repeater

Right-click on any of the GET /product?productId=[...] requests and select Send to Repeater.

Das	hboard	Target	Proxy	Intruder	Repeater	Sequencer	Decod	ler C	omparer	Logge
Inter	rcept	HTTP history	WebSo	ockets histo	ory Options	3				
Filter:	Hiding C	SS, image and	general binar	ry content						
# ~		Host	Method		URL		Params	Edited	Status	Length
1	•	100 1130 1147 13		1					200	10044
6		ac5b1f3b1f4713			yLabHeader				101	147
15	https://a	ac5b1f3b1f4713	GET	/product	?productId=3		1		200	4949
4		ac5b1f3b1f4713		/academ	yLabHeader	https://a	c5b1f3b1f47	713de805	eademy.r	net/produc
3	https://a	ac5b1f3b1f4713	GET	/						
2	https://a	ac5b1f3b1f4713	GET	/academ	yLabHeader	Add to s	cope			
1	httne.//a	0561f261f/712	GET	/product/	productId=2	Scan				
						Do passi	vo coop			
Requ	lost					Do passi	ve scan			
nequ	uest					Do active	e scan			
Pretty	y Raw	Hex \n \equiv				Send to	Intruder			
1 GE		luct?product	Id=3 HTTP	/1.1		Send to	Repeater			
		f4713de805e	481900880	0c4.web-s	security-acad	Send to	Sequencer			

Go to the **Repeater** tab to see that your request is waiting for you in its own numbered tab.

5.4 Send the request and view the response

Click Send and view the response from the server. You can resend this request as many times as you like and the response will be updated each time.

Dashboard Target Proxy Intruder Repeater	Sequencer Decoder Comparer Logge
1 ×	
Send Cancel < v > v	Target: https://ac5b1f3b1f4713de
Request	Response
Pretty Raw Hex In =	Pretty Raw Hex Render \n =
<pre>1 GET /product?productId=3 HTTP/1.1 2 Host: ac5blf3blf4713de805e4819008800c4.web-security-ac ademy.net 3 Sec-Ch-Ua: " Not A;Brand";v="99", "Chromium";v="92" 4 Sec-Ch-Ua-Mobile: ?0 5 Upgrade-Insecure-Requests: 1 6 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/92.0.4515.131 Safari/537.36 7 Accept: text/html,application/xhtml+xml,application/xml;</pre>	<pre>1 HTTP/1.1 200 OK 2 Content-Type: text/html; charset=utf-8 3 Connection: close 4 Content-Length: 4142 5 6 <!DOCTYPE html> 7 <html> 8 <head> 9 <link 10="" 11="" <link="" <title="" href="/resources/labheader/css"/> 11 formation disclosure in error m </head></html></pre>

5.5 Testing different input with Burp Repeater

By resending the same request with different input each time, you can identify and confirm a variety of input-based vulnerabilities. This is one of the most common tasks you will perform during manual testing with Burp Suite.

5.6 Resend the request with different input

Change the number in the productId parameter and resend the request.

Try this with a few arbitrary numbers, including a couple of larger ones.

Dashboard	Target	Proxy	Intruder	Repeater	Sequencer	Decoder	Comparer	Logge
1 ×								
Send	Cancel	< v >			Target: h	ttps://ac5b1f3b	1f4713de	
Request				Response	•			
Pretty Raw	Hex ∖n ≡	=			Pretty Rav	v Hex Render	\n ≡	
ademy.net Sec-Ch-Ua "Chromium Sec-Ch-Ua Upgrade-I User-Agen x64) App	f4713de805 : " Not A; ";v="92" -Mobile: ? nsecure-Re t: Mozilla leWebKit/5 .0.4515.13	6e48190088 Brand";v= 20 20 20 20 20 20 20 20 20 20 20 20 20	800c4.web-s ="99", L ndows NT 10 HTML, like	2 Content 3 Connect	1 404 Not Fo -Type: appli- ion: close -Length: 11 und"		charse	

5.7 View the request history

Use the arrows to step back and forth through the history of requests that you've sent, along with their matching responses. The drop-down menu next to each arrow also lets you jump to a specific request in the history.

Dashboard	Target	Proxy	Intruder	Repeater	Sequencer	Decoder	Comparer	Logge
1 ×								
Send	Cancel	< v >	•			Target: h	ttps://ac5b1f3b	1f4713de
		8. http	os://ac5b1f3b1	lf4713de8ade	my.net/product?	productId=2		
Request		7. http	os://ac5b1f3b1	lf4713de8ade	my.net/product?	productId=1		
Pretty Raw I	Hex ∖n ≡	6. http	os://ac5b1f3b1	lf4713de8emy	.net/product?pr	oductId=245	\n ≡	
1 GET /produ 2 Host: ac5b1f3b1: ademy.net 3 Sec-Ch-Ua 5 Upgrade-In 6 User-Agen	f4713de805 : " Not A; ";v="92" -Mobile: 3 nsecure-Re t: Mozilla	5 4. http 3. http 2. http 1. http	os://ac5b1f3b ⁻ os://ac5b1f3b ⁻ os://ac5b1f3b ⁻ os://ac5b1f3b ⁻	lf4713de8dem lf4713de8ade lf4713de8emy lf4713de8ade	ny.net/product?p ny.net/product?p my.net/product? /.net/product?pr my.net/product?pr	productId=25 productId=2 oductId=100 productId=3	ml; charset	der/cs
Chrome/92 7 Accept:	.0.4515.13	31 Safari,		Gecko) ation/xml;	11 <tit]< td=""><td>Le> Formation di</td><td>ources/css/la</td><td></td></tit]<>	Le> Formation di	ources/css/la	

This is useful for returning to previous requests that you've sent in order to investigate a particular input further.

Compare the content of the responses, notice that you can successfully request different product pages by entering their ID, but receive a Not Found response if the server was unable to find a product with the given ID. Now we know how this page is supposed to work, we can use Burp Repeater to see how it responds to unexpected input.

5.8 Try sending unexpected input

The server seemingly expects to receive an integer value via this productId parameter. Let's see what happens if we send a different data type. Send another request where the productId is a string of characters.

1 ×	_							
Send	Cancel	< v >				Target: h	ttps://ac5b1f3b	1f4713de
	_							
Request	Response							
Pretty Raw	Hex ∖n ≡	=						
1 GET /pro	duct?prod	ductId=te	st HTTP/1	1.1				
	5h1f2h1f/	171346805	A 8190089	200c4 web-s	ecurity-aca	demy net		

5.9 Study the response

Observe that sending a non-integer productId has caused an exception. The server has sent a verbose error response containing a stack trace.

Dashboard	Target	Proxy	Intruder	Repeater	Sequencer	Decoder	Comparer	Logge
×								
Send	Cancel	< 🔻 🔁 >				Target: h	ttps://ac5b1f3b	1f4713de
Request	Response							
retty Raw I	Hex Render	\n ≡						
HTTP/1.1 5	500 Intern	al Server	Error					
Connection	n: close							
Content-Le		8						
Content-Le	ength: 268		long Numbe		ntion. Ton i	ant stairs	"test"	
Content-Le	ength: 268 Server Err	or: java.			ption: For in			7a • 68)
Content-Le Internal s at java.ba	ength: 268 Server Err ase/java.l	or: java. ang.Numbe	erFormatExc	eption.forI	nputString(Nu			va:68)
Content-Le Internal S at java.ba at java.ba	ength: 268 Server Err ase/java.l ase/java.l	or: java. ang.Numbe ang.Integ	erFormatExc ger.parseIn	eption.forI t(Integer.j	nputString(Nu ava: <mark>658</mark>)			va: <mark>6</mark> 8)
Content-Le Internal S at java.bs at java.bs at java.bs	ength: 268 Server Err ase/java.l ase/java.l ase/java.l	or: java. ang.Numbe ang.Integ ang.Integ	erFormatExc ger.parseIn ger.parseIn	eption.forI t(Integer.j t(Integer.j	nputString(Nu ava: <mark>658</mark>)	umberFormatI	Exception.jav	,
Content-Le Internal S at java.ba at java.ba at java.ba at java.ba at lab.dat	ength: 268 Server Err ase/java.l ase/java.l ase/java.l ta.product splay.prod	cor: java. ang.Numbe ang.Integ ang.Integ catalog.c luctcatalo	erFormatExc ger.parseIn ger.parseIn catalog.Def og.filter.N	eption.forI ht(Integer.j ht(Integer.j aultProduct NoFilterStra	nputString(Nu ava:658) ava:776) CatalogDataSo tegy.getProdu	umberFormatI purce.getPro	Exception.jav oduct(Default rStrategy.jav	tProduc va:47)
Content-Le Internal S at java.bs at java.bs at java.ba at lab.da at lab.da at lab.da	ength: 268 Server Err ase/java.l ase/java.l ta.product splay.prod splay.prod	or: java. ang.Numbe ang.Integ ang.Integ catalog.c luctcatalo luctcatalo	erFormatExc ger.parseIn ger.parseIn catalog.Def og.filter.N og.page.pro	eption.forI at(Integer.j at(Integer.j aultProduct loFilterStra oduct.Simple	nputString(Nu ava:658) ava:776) CatalogDataSo tegy.getProdu ProductStrate	umberFormatI ource.getPro uct(NoFilter egy.handle(S	Exception.jay oduct(Default rStrategy.jay SimpleProduct	tProduc va:47) tStrate
Content-Le Internal S at java.ba at java.ba at java.ba at lab.da at lab.da at lab.da at lab.da	ength: 268 Server Err ase/java.l ase/java.l ta.product splay.prod splay.prod splay.prod	or: java. ang.Numbe ang.Integ catalog.c luctcatalo luctcatalo luctcatalo	erFormatExc ger.parseIn ger.parseIn catalog.Def og.filter.N og.page.pro	eption.forI at(Integer.j at(Integer.j aultProduct IoFilterStra oduct.Simple apleProductP	nputString(N ava:658) ava:776) CatalogDataSo tegy.getProdu ProductStrate ageStrategy.	umberFormatI ource.getPro uct(NoFilte egy.handle(S lambda\$hand]	Exception.jay oduct(Default rStrategy.jay SimpleProduct	tProduc va:47) tStrate
Content-Le Internal S at java.ba at java.ba at lab.da at lab.da at lab.da at lab.da at lab.da at lab.da	ength: 268 Server Err ase/java.l ase/java.l ta.product splay.prod splay.prod splay.prod rtswigger.	or: java. ang.Numbe ang.Integ catalog.c luctcatalo luctcatalo luctcatalo uctcatalo util.Unch	erFormatExc ger.parseIr ger.parseIr catalog.Def og.filter.N og.page.pro og.page.Sin necked.lamb	eption.forI at(Integer.j faultProduct IoFilterStra oduct.Simple pleProductP oda\$null\$3(U	nputString(Nu ava:658) ava:776) CatalogDataSo tegy.getProdu ProducStrate ageStrategy. nchecked.java	umberFormatI ource.getPro uct(NoFilte egy.handle(S lambda\$hand]	Exception.jay oduct(Default rStrategy.jay SimpleProduct	tProduc va:47) tStrate
Content-Le Internal 3 at java.ba at java.ba at lab.da at lab.da at lab.da at lab.da at lab.da at lab.da at lab.da at lab.da at lab.da	ength: 268 Server Err ase/java.l ase/java.l ta.product splay.prod splay.prod splay.prod splay.prod rtswigger.	or: java. ang.Numbe ang.Integ catalog.c luctcatalo luctcatalo luctcatalo util.Unch	erFormatExc ger.parseIr ger.parseIr satalog.Def og.filter.N og.page.pro og.page.Sin necked.lamb necked.unch	eption.forI at(Integer.j aultProduct IoFilterStra oduct.Simple pleProductP oda\$null\$3(U neck(Uncheck	nputString(Nu ava:658) ava:776) CatalogDataSo tegy.getProdu ProductStrate ageStrategy. nchecked.java	umberFormatI ource.getPro uct(NoFilte) egy.handle(s lambda\$hand) a:46)	Exception.jav oduct(Default rStrategy.jav SimpleProduct leSubRequest:	tProduc va:47) tStrate
Content-Le Internal 3 at java.ba at java.ba at lab.da at lab.da at lab.da at lab.da at lab.da at lab.da at lab.da at lab.da at lab.da at lab.da	ength: 268 Server Err ase/java.l ase/java.l ta.product splay.prod splay.prod splay.prod ttswigger. rtswigger.	or: java. ang.Numbe ang.Integ catalog.c luctcatalc luctcatalc util.Unch util.Unch	erFormatExc ger.parseIr ger.parseIr satalog.Def og.filter.N og.page.pro og.page.Sin necked.lamb necked.unch	<pre>teption.forI t(Integer.j aultProduct oFilterStra oduct.Simple pleProductP oda\$null\$3(U ueck(Uncheck oda\$unchecke</pre>	nputString(Nu ava:658) ava:776) CatalogDataSc tegy.getProdu ProductStrata ageStrategy. nchecked.java ed.java:73) dFunction\$4(1	umberFormatI ource.getPro uct(NoFilte) egy.handle(s lambda\$hand) a:46)	Exception.jav oduct(Default rStrategy.jav SimpleProduct leSubRequest:	tProduc va:47) tStrate
Internal 3 at java.ba at java.ba at lab.da at lab.da at lab.da at lab.da at lab.da at lab.da at lab.da at net.po at net.po at net.po at java.ba	ength: 268 Server Err ase/java.l ase/java.l ase/java.l ta.product splay.prod splay.prod splay.prod splay.prod rtswigger. rtswigger. ase/java.u	or: java. ang.Numbe ang.Integ catalog.c luctcatalo luctcatalo luctcatalo util.Unch util.Unch util.Unch	erFormatExc ger.parseIr ger.parseIr ger.parseIr catalog.Def og.filter.N og.page.pro og.page.Sin necked.lamb necked.lamb pocked.lamb	eption.forI tt(Integer.j aultProduct boFilterStra bduct.Simple ppleProductP bda\$null\$3(U heck(Uncheck dda\$unchecke btional.java	nputString(Nu ava:658) ava:776) CatalogDataSd tegy.getProdu ProductStrate ageStrategy. nchecked.java ed.java:73) dFunction\$4(I :265)	umberFormatI ource.getPro uct(NoFilte) egy.handle(lambda\$hand) a:46) Unchecked.ja	Exception.jav oduct(Default rStrategy.jav SimpleProduct leSubRequest: ava:46)	tProduc va:47) tStrate \$0(Simp
Content-Le Internal S at java.ba at java.ba at lab.da at lab.da at lab.da at lab.da at lab.da at lab.da at net.po at net.po at net.po at java.ba at lab.da	ength: 268 Server Err ase/java.l ase/java.l ase/java.l ta.product splay.prod splay.prod rtswigger. rtswigger. rtswigger. sse/java.u splay.prod	cor: java. ang.Numbe ang.Integ catalog.c luctcatalc luctcatalc util.Unch util.Unch util.Unch lutil.Optic luctcatalc	erFormatExc ger.parseIn ger.parseIn ger.parseIn ger.parseIn og.page.Def og.page.sin hecked.lamb hecked.lamb hecked.lamb onal.map(Op gg.page.Sin	eption.forI tt(Integer.j saultProduct loFilterStra oduct.Simple upleProductP oda\$null\$3(U eeck(Uncheck oda\$unchecke otional.java upleProductP	nputString(Nu ava:658) ava:776) CatalogDataSc tegy.getProdu ProductStrata ageStrategy. nchecked.java ed.java:73) dFunction\$4(1	umberFormatI ource.getPro uct(NoFilte: egy.handle(S lambda\$hand) a:46) Unchecked.ja handleSubRed	Exception.jav oduct(Default rStrategy.jav SimpleProduct leSubRequest: ava:46)	tProduc va:47) tStrate \$0(Simp

Notice that the response tells you that the website is using the Apache Struts framework - it even reveals which version.



6 Run your first scan

Burp Scanner can be used as both a fully automated scanner and a powerful means of augmenting your manual testing workflow. The list of vulnerabilities that Burp Scanner can detect is constantly growing. We work closely with our world-class research team to make sure that it stays up to speed with the latest techniques for finding both classic bugs and newly discovered vulnerabilities alike.

Scanning a website involves two phases:

- 1. **Crawling for content and functionality:** Burp Scanner first navigates around the target site, closely mirroring the behavior of real users. It catalogs the structure and content of the site, and the paths used to navigate it, in order to build a comprehensive map of the site.
- 2. Auditing for vulnerabilities: The audit phase of a scan involves analyzing the website's behavior to identify security vulnerabilities and other issues. Burp Scanner employs a wide range of techniques to deliver a high-coverage, accurate audit of the target.

6.1 Open the scan launcher

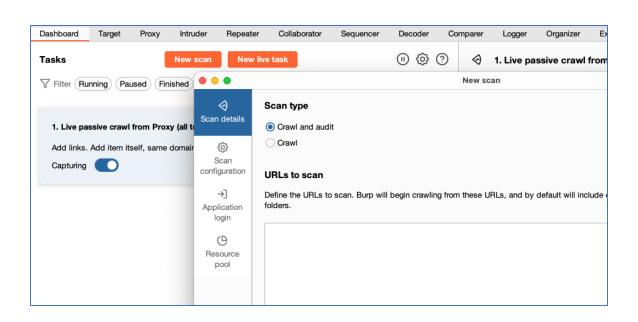
Go to the Dashboard tab and select new scan.

Dashboard	Target	Proxy	Intruder	Repeater	Collaborator	Sequencer	Decoder	Comp	arer Logg	er O	rganizer	E
Tasks			New scan	New live	task		@ ₿	?	1. Live	e passiv	e crawl	from
Tilter Ru	nning Pa	used Fin	ished	ve task Sca	n Intruder atta	ck		Q	Summary	_		
1. Live pa	ssive crawl	from Prox	xy (all traffic)				(1)		ltems	added to	o site ma	ър
Add links.	Add item it	self, same	domain and l	JRLs in suite s	cope.				Host		Method	URL
Capturing									ginandjuid	ce.shop	GET	/
									ginandjuid	e.shop	GET	/reso
									ginandjuid		GET	/reso
									ginandjuid		GET	/reso
									ginandjuid		GET	/reso
									ginandjuid		GET	/reso
									ginandjuid		GET	/reso
									ginandjuid		GET	/cata
									ginandjuid		GET GET	/blog
									ginandjuid	•	GET	/abou
									ginandjuid		GET	/my-a /cata
L									ginandjuid	e.snop	GET	/cata

The Scan launcher dialog opens. This is where you can adjust various settings to control Burp Scanner's behavior.

6.2 Enter the URL of the target site

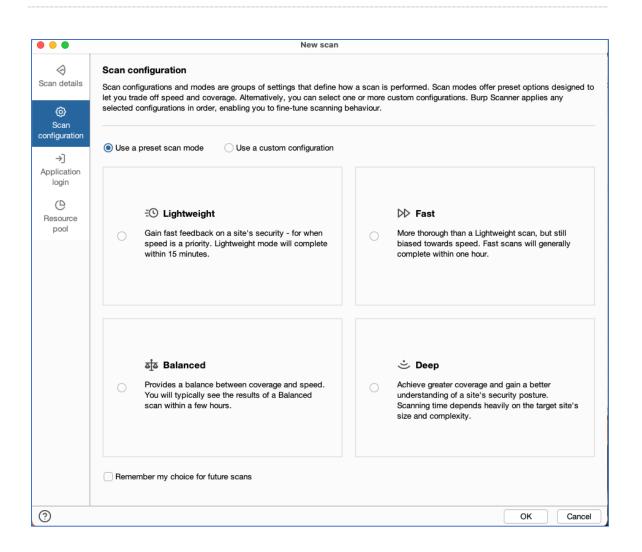
In the **URLs to scan** field, enter ginandjuice. shop. If necessary, remove the URL for the website that you set as a target scope in the earlier tutorial, Leave all the other settings as their default for now.



6.3 Configure the scan

Select Scan configuration. From here, you can fine-tune many aspects of Burp Scanner's behavior to suit different use cases and target sites

Make sure that use a preset scan mode is selected and click Lightweight. The Lightweight scan mode is intended to give a very high-level overview of a target as quickly as possible. Scans using this mode run for a maximum of 15 minutes.



6.4 Launch the scan

Click **OK** to launch the scan. Burp Scanner begins crawling from the URL you entered in the previous step.

Notice that a new task has been added to the **Dashboard** to represent this scan. You can select the task to view more details about its status and what it's currently doing.

Dashboard	Target	Proxy	Intruder	Repeater	Collaborator	Sequencer	Decoder	Comparer	Logger	Organizer	E
asks	New scan	N	ew live task	(I) (Q	€ © €	3. Crawl a	nd audit of	ginandjuice.s	hop		
🏹 Filter 🗸			1	Search	و و	Summary A	udit items	lssues Ev	ent log	Logger A	Audit
1. Live pa	ssive crawl fr	om Prox	xy (all traffic)	(1)	:	▲ Most serio	us vulnerabili	ties found (live)		
Add links. scope.	. Add item itse	lf, same	domain and l	JRLs in suite		Issue type			Host		Tim
Capturing						Cross-site s	cripting (refle	cted)	https://gi	nandjuice.s	11:
						Cross-site s	cripting (DON	1-based)	https://gi	nandjuice.s	11:
						? SQL injection	n		https://gi	nandjuice.s	11:
			er -)	0		! Password fi	eld with autoo	complete enabl.	. https://gi	nandjuice.s	11:
2. Live au	dit from Prox	y (all tra	піс)	(1)	:	Strict transp	ort security n	ot enforced	https://gi	nandjuice.s	11:
Audit cheo	cks - passive					Open redire	ction (DOM-b	ased)	https://gi	nandjuice.s	11:
Capturing						Open redire	ction (DOM-b	ased)	https://gi	nandjuice.s	11:
5		laa	Jes: 🚺			Vulnerable .	JavaScript de	pendency	https://gi	nandjuice.s	11:
		1551	<i>les.</i>			Cacheable I	HTTPS respon	se	https://gi	nandjuice.s	11:
						Cookie with	out HttpOnly 1	flag set	https://gi	nandjuice.s	11:
						Cookie with	out HttpOnly f	flag set	https://gi	nandjuice.s	11:
3. Crawl a	and audit of g	inandjui	ce.shop	(1)		Input returne	ed in response	e (reflected)	https://gi	nandjuice.s	11:
Crawl and	Audit - Lightv	veight				Input returne		(https://gi	nandjuice.s	11:
	-						ed in response			nandjuice.s	
	g					Input returned		e (reflected)		nandjuice.s	
		lssu	Jes: 🖪	0 5	13	TLS certifica	ate		https://gi	nandjuice.s	11:
			-			TI C apoleio	without a cours	flag aat	https://ai	andiulan a	44.

6.5 See the crawl in action

Go to the Target > Site map tab and notice the new entry for ginandjuice. shop. Expand this node to see all of the content that the crawler has managed to discover so far. If you wait a few seconds, you'll see the map being updated in real time.

Dashboard	Target	Proxy	Intruder	Repeater	Collaborator	Sequencer	Decoder	Comparer	Logg	ər Org	anizer Ex
Site map	Crawl paths	s (beta)	Issue definit	tions { දි	Scope settings						
∏ Filter: Hid	ding not found	d items; hi	ding CSS, ima	ge and genera	I binary content; h	iding 4xx respon	ses; hiding (empty folders			
e v 🖳 http:	://ginandjuice	chon		Conte	ents Issues	•					
	s.//ginanujuice	e.snop			133063						
·-• 🗅 /											
💛 🗋 ab	out			Host		Method				Params	Status Co
ble 🔂 🖳	og			https://	ginandjuice.shop	GET	/				200
🛥 > 🔂 bl	og			https://	ginandjuice.shop	GET	/about				200
🛶 > 🔂 ca	italog			https://	ginandjuice.shop	GET	/blog				200
				https://	ginandjuice.shop	GET	/blog/				
🛁 > 🌼 ca	0			https://	ginandjuice.shop	GET	/blog/?s	earch=&back=9	62Fbl	~	200
🗀 > 🌼 loạ	gger			https://	ginandjuice.shop	GET	/blog/?s	earch=PjctNs&b	back=	~	200
🔄 > 🍪 log	gin				ginandjuice.shop	GET	/blog/po				
	y-account				ginandjuice.shop	GET	01	st?postld=1		~	200
	,			https://	ainandiuice.shop	GET	/bloa/pa	st?postId=2		· _	200
🕒 > 🔂 re:	sources										
> 🎒 https	://github.com	ı		Requ	lest					R	esponse
> 🎒 https	://html.spec.	whatwg.o	rg	Pretty	Raw He	ĸ			.⇒ \n	Ξ F	retty Rav
> 🎒 https	://infra.spec.	whatwg.o	rg	1 GE	T / HTTP/2						1 HTTP/2
> iii http:	//modernizr.c	om		2 Ho	st: ginandju	ice.shop					2 Date: W

6.6 View the identified issues

Monitor the scan's status in the Dashboard tab. After a minute or two, the crawl will finish and Burp Scanner will begin auditing for vulnerabilities. To monitor the scan for any issues it finds, select the scan from the Tasks list. In the main panel, go to the Issues tab.

Dashboard	Target Proxy	Intruder	Repeater	Collaborato	r Sequenc	er De	coder	Comparer	Logger	Organizer	Ext
Tasks ∵ Filter ∽	New scan	New live task	earch	0 0	Summary	vi and au Audit it	•	andjuice.	shop vent log	Logger A	Audit log
	ssive crawl from Pro		(1)		Ţ Filter (H	igh Med	lium Low	Info	Certain	Firm Tentati	ve) (
Add links. scope. Capturing	Add item itself, same	e domain and Uf	RLs in suite		Time ∨ 12:05:36 29 N 12:04:46 29 N 12:02:35 29 N 12:02:35 29 N	ov 2023 ov 2023 ov 2023	Source Task 3 Task 3 Task 3 Task 3	Input	s-site script returned in nal service nal service	ting (reflected) response (refle interaction (HT interaction (DN	ΓP)
	dit from Proxy (all tr cks - passive				Advisory	Reques		onse C e interac		HTTP interactio	vn F
	and audit of ginandju Audit - Lightweight	uice.shop	Ø	:	Issue: Severity: Confidenc Host: Path:	High e: Certai	n //ginandju	interaction	n (HTTP)		
Pause	d task due to: Reache	ed time limit for t sues: 9	ask 0 5 (16		le to indu				erver-side HTT	

If you select an issue, you can see an **Advisory** tab, which contains key information about the issue type, including a detailed description and some remediation advice. Next to this are several tabs that provide evidence that Burp Scanner found for this issue. This is typically a **Request** and **Response** but will differ depending on the issue type.