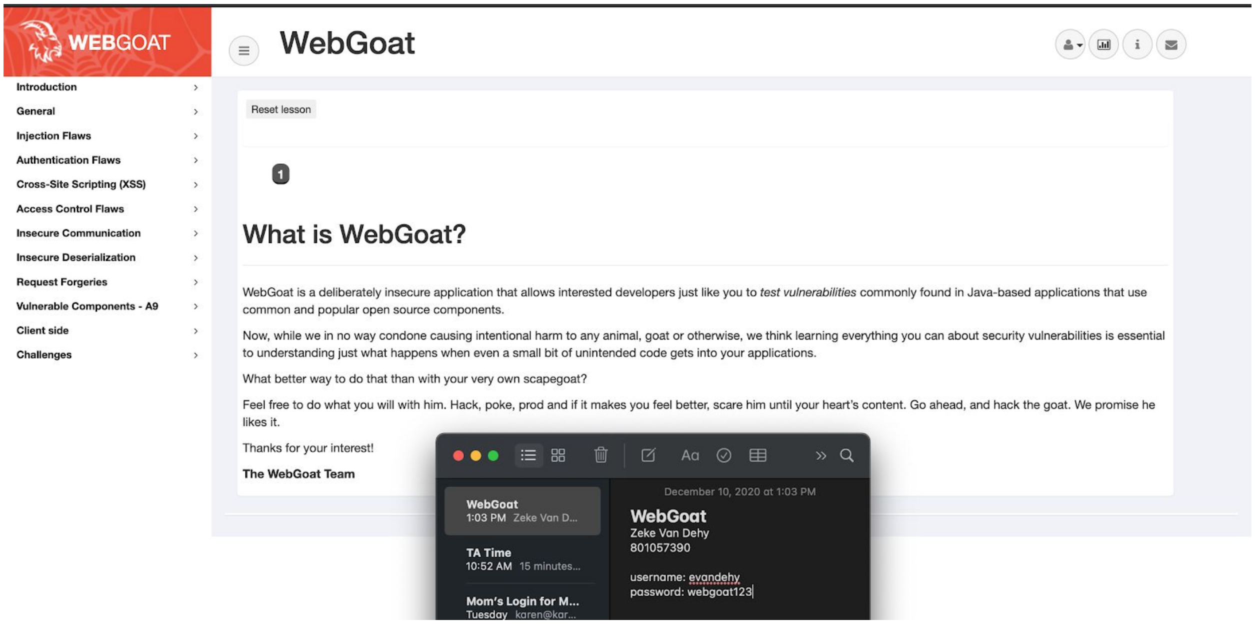
Part 1:



WebGoat

What is WebGoat?

WebGoat is a deliberately insecure application that allows interested developers just like you to *test vulnerabilities* commonly found in Java-based applications that use common and popular open source components.

Now, while we in no way condone causing intentional harm to any animal, goat or otherwise, we think learning everything you can about security vulnerabilities is essential to understanding just what happens when even a small bit of unintended code gets into your applications.

What better way to do that than with your very own scapegoat?

Feel free to do what you will with him. Hack, poke, prod and if it makes you feel better, scare him until your heart's content. Go ahead, and hack the goat. We promise he likes it.

Thanks for your interest!

The WebGoat Team

WebGoat
1:03 PM Zeke Van D...

TA Time
10:52 AM 15 minutes...

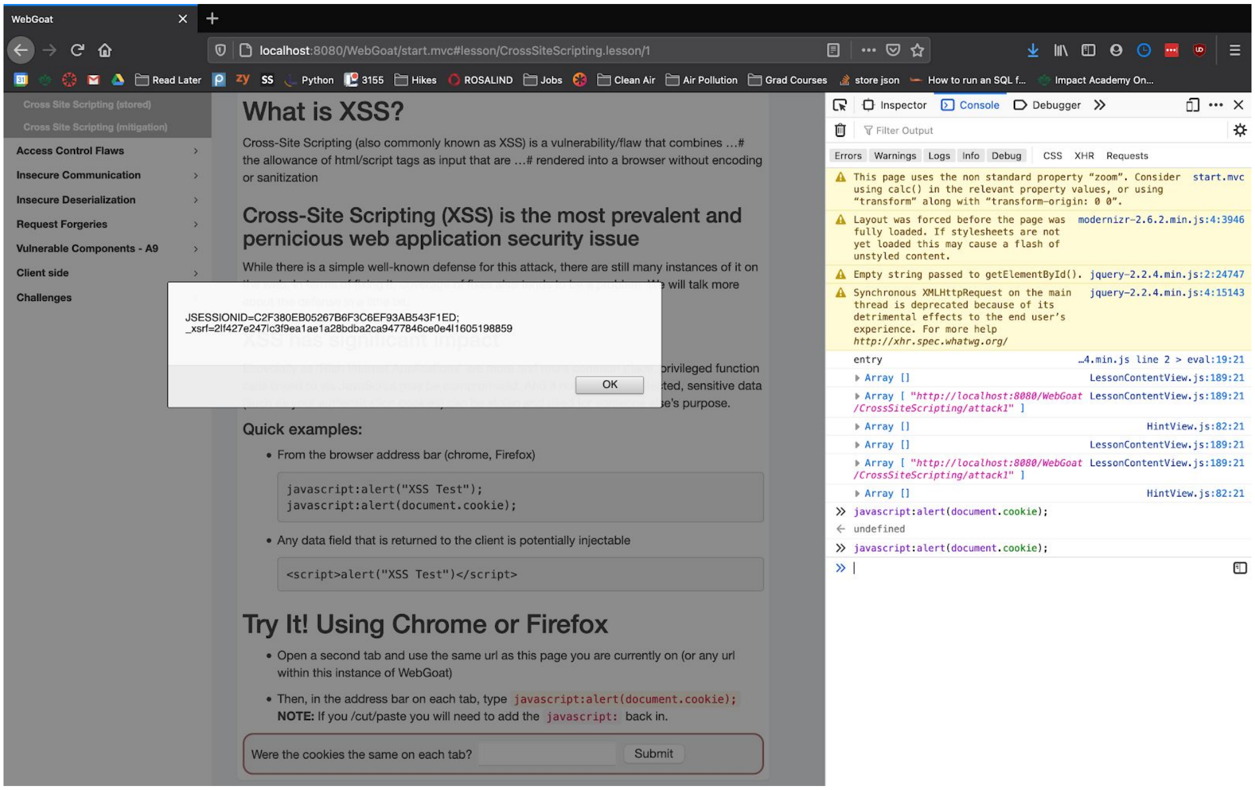
Mom's Login for M...
Tuesday karen@kar...

December 10, 2020 at 1:03 PM

WebGoat
Zeke Van Dehy
801057390

username: evandehy
password: webgoat123

Screenshot 1:



What is XSS?

Cross-Site Scripting (also commonly known as XSS) is a vulnerability/flaw that combines ...# the allowance of html/script tags as input that are ...# rendered into a browser without encoding or sanitization

Cross-Site Scripting (XSS) is the most prevalent and pernicious web application security issue

While there is a simple well-known defense for this attack, there are still many instances of it on ... we will talk more

... privileged function ... ted, sensitive data ... e's purpose.

```
JSESSIONID=C2F380EB05267B6F3C6EF93AB543F1ED;
_xsrf=2|f427e247|c3f9ea1ae1a28bdba2ca9477846ce0e4|1605198859
```

Quick examples:

- From the browser address bar (chrome, Firefox)

```
javascript:alert("XSS Test");
javascript:alert(document.cookie);
```

- Any data field that is returned to the client is potentially injectable

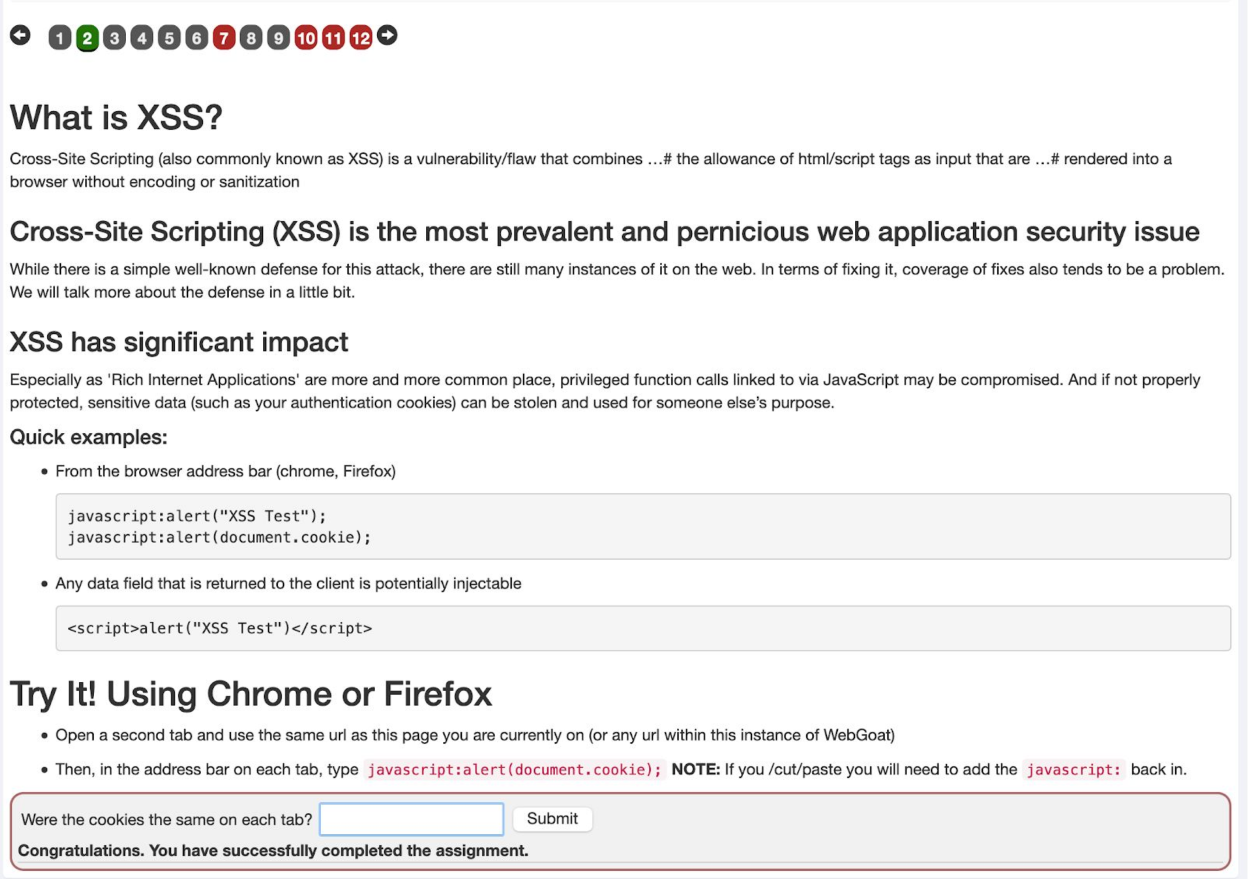
```
<script>alert("XSS Test")</script>
```

Try It! Using Chrome or Firefox

- Open a second tab and use the same url as this page you are currently on (or any url within this instance of WebGoat)
- Then, in the address bar on each tab, type `javascript:alert(document.cookie);` **NOTE:** If you /cut/paste you will need to add the `javascript:` back in.

Were the cookies the same on each tab?

```
This page uses the non standard property “zoom”. Consider     start.mvc
using calc() in the relevant property values, or using
“transform” along with “transform-origin: 0 0”.
Layout was forced before the page was    modernizr-2.6.2.min.js:4:3946
fully loaded. If stylesheets are not
yet loaded this may cause a flash of
unstyled content.
Empty string passed to getElementById().  jquery-2.2.4.min.js:2:24747
Synchronous XMLHttpRequest on the main    jquery-2.2.4.min.js:4:15143
thread is deprecated because of its
detrimental effects to the end user’s
experience. For more help
http://xhr.spec.whatwg.org/
entry                                     …4.min.js line 2 > eval:19:21
▸ Array []                                LessonContentView.js:189:21
▸ Array [ "http://localhost:8080/WebGoat  LessonContentView.js:189:21
/CrossSiteScripting/attack1" ]
▸ Array []                                         HintView.js:82:21
▸ Array []                                LessonContentView.js:189:21
▸ Array [ "http://localhost:8080/WebGoat  LessonContentView.js:189:21
/CrossSiteScripting/attack1" ]
▸ Array []                                         HintView.js:82:21
» javascript:alert(document.cookie);
← undefined
» javascript:alert(document.cookie);
» |
```



What is XSS?

Cross-Site Scripting (also commonly known as XSS) is a vulnerability/flaw that combines …# the allowance of html/script tags as input that are …# rendered into a browser without encoding or sanitization

Cross-Site Scripting (XSS) is the most prevalent and pernicious web application security issue

While there is a simple well-known defense for this attack, there are still many instances of it on the web. In terms of fixing it, coverage of fixes also tends to be a problem. We will talk more about the defense in a little bit.

XSS has significant impact

Especially as 'Rich Internet Applications' are more and more common place, privileged function calls linked to via JavaScript may be compromised. And if not properly protected, sensitive data (such as your authentication cookies) can be stolen and used for someone else’s purpose.

Quick examples:

- From the browser address bar (chrome, Firefox)

```
javascript:alert("XSS Test");
javascript:alert(document.cookie);
```

- Any data field that is returned to the client is potentially injectable

```
<script>alert("XSS Test")</script>
```

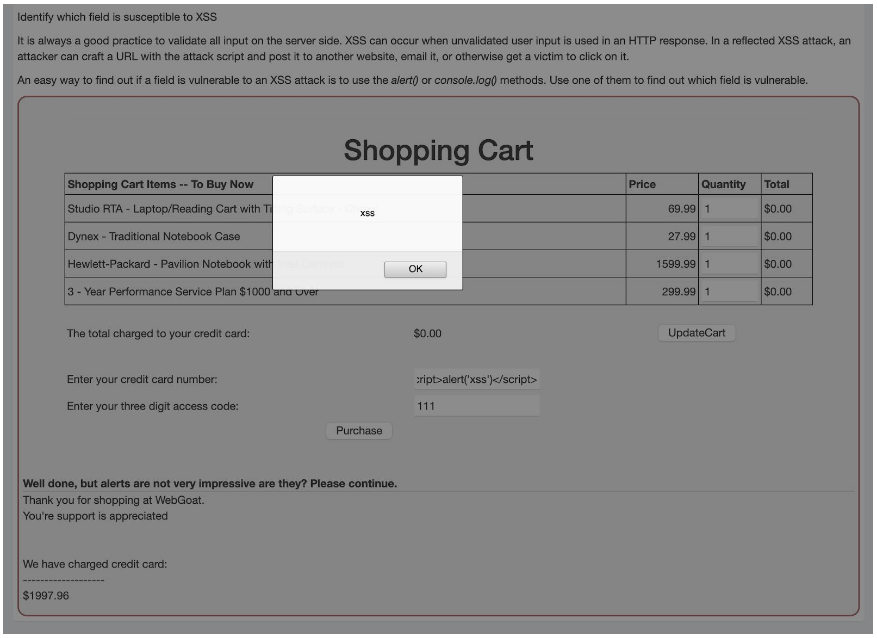
Try It! Using Chrome or Firefox

- Open a second tab and use the same url as this page you are currently on (or any url within this instance of WebGoat)
- Then, in the address bar on each tab, type `javascript:alert(document.cookie);` **NOTE:** If you /cut/paste you will need to add the `javascript:` back in.

Were the cookies the same on each tab?

Congratulations. You have successfully completed the assignment.

Screenshot 2:



Identify which field is susceptible to XSS

It is always a good practice to validate all input on the server side. XSS can occur when unvalidated user input is used in an HTTP response. In a reflected XSS attack, an attacker can craft a URL with the attack script and post it to another website, email it, or otherwise get a victim to click on it.

An easy way to find out if a field is vulnerable to an XSS attack is to use the *alert()* or *console.log()* methods. Use one of them to find out which field is vulnerable.

Shopping Cart

Shopping Cart Items -- To Buy Now	Price	Quantity	Total
Studio RTA - Laptop/Reading Cart with Ti	69.99	1	$0.00
Dynex - Traditional Notebook Case	27.99	1	$0.00
Hewlett-Packard - Pavilion Notebook with Intel Centrino	1599.99	1	$0.00
3 - Year Performance Service Plan $1000 and Over	299.99	1	$0.00

The total charged to your credit card:	$0.00	UpdateCart
Enter your credit card number:	:ript>alert('xss')</script>	
Enter your three digit access code:	111	

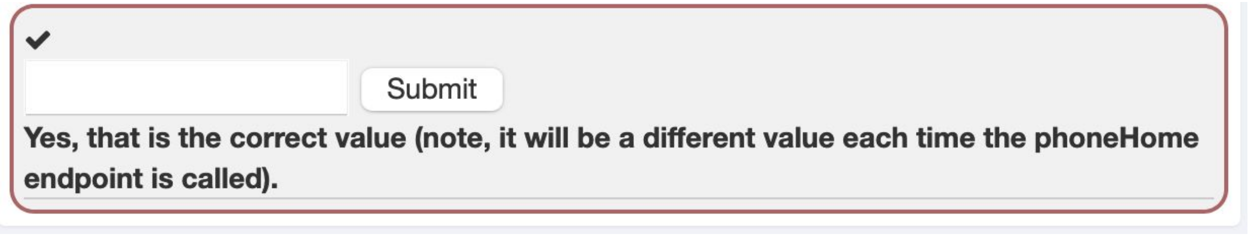
Well done, but alerts are not very impressive are they? Please continue.

Thank you for shopping at WebGoat.
You're support is appreciated

We have charged credit card:

$1997.96

Screenshot 3:



Yes, that is the correct value (note, it will be a different value each time the phoneHome endpoint is called).

Q1: When an attacker enters javascript into vulnerable text fields, the attacker makes the web server execute that javascript. This means that the attacker can write any malicious javascript code to steal data, show the user incorrect information, or other harmful attacks, and execute that javascript on the vulnerable website. DOM-based XSS is similar where the link provided to the user is malicious and doesn't use script tags and the attack happens on the client-side.

Q2: A stored XSS attack takes advantage of the websites storing mechanisms (databases), by storing malicious text (javascript) code in the database. This means that a user doesn't have to navigate to a malicious link.

Q3: Some measures to block an XSS attack are validating data, encoding data before redisplay, not relying on only client-side validation, and escaping key characters.

Q4: An XSS attack can steal user credentials, private information, or session cookies.

Q5: Common locations vulnerable to an XSS attack include search fields & input fields (that echo data back to the user), error messages with user-entered information, hidden fields, http headers, and any page that displays user-supplied data (comments & message boards).

Q6: We should care because attackers can steal cookies, create false requests, collect private information or user credentials, redirect users to malicious sites, insert hostile or inappropriate content, or lead to phishing attacks.