



University of Colorado **Boulder**

Secure Web Programming

Joe McManus

Interdisciplinary Telecommunications Program

University of Colorado

Spring 2016



University of Colorado
Boulder

Course Goals

- Learn how to develop web application that will be resilient to attacks, protect data and systems.
- Learn to develop complex distributed systems using PHP, MySQL and Python.
- No knowledge of SQL, PHP or Python required.



Course Methodology

- Hands on labs will be used throughout the course.
- Labs not completed in class will be finished as homework.
- Final project of your choice.



About me

- Scholar in Residence at ITP University of Colorado.
- Sr. Security Researcher CERT Carnegie Mellon University.
- Past:
 - Director of Security and IT Solidfire
 - Sr. Research Manager Webroot
 - Security Researcher CERT/SEI/CMU



Office Hours

- Monday 3-4 or by appointment.
joe.mcmanus@colorado.edu



Learning Environment

- You will use two Fedora Linux virtual machines.
- One will be the web server and php server. The second will be the MySQL database server.
- Can use IDE or editor of choice for coding. I use vi.
- One VirtualBox on your laptop

Rough Timeline

- Weeks 1-4: Introduction to PHP; Intro to SQL
- Week 5: Intro Multi Tiered Apps
- Week 6: Securing MySQL
- Week 7: Midterm/Practical/SRS
- Week 8: Encrypting Transport
- Week 9: User Auth
- Week 10: SQLi and XSS Defense
- Week 11: PCI and HIPAA
- Week 12: Logging and Debugging
- Week 13/14: Group Project



Books

- Essential PHP Security

ISBN 978-0596006563

Chris Shiflett

Oreilly Media



University of Colorado
Boulder

Attendance

- Attendance is mandatory.
- This is grad school, this is your job so you should probably show up.
- Labs are given in class and no makeup is available.

Attendance

- Name tags on your desks please.
- Introductions.
 - Name
 - Program
 - Hobby
 - If you had to program, what language would you choose.

Distance

- We have a lot of distance students this summer.
- I usually teach this very hands on.
- We can do a Google hangout or Skype once a week to do QA and someone will present their project/homework.

Cheating

- Zero Tolerance.
- If you cheat you get a 0/100.
- You will be reported.
- You are in grad school, if you feel the need to cheat you should rethink if grad school is right for you.



Connect to learning environment

- Can be connected to from anywhere in the CU network. You will need to VPN to CU to work from home.
- You have a web server and database server.
- For the next few weeks you only need the web server.



Your machines

- You have a web server and a database server.
- You will need to log in to both and change the root password.
- `passwd`

Your Virtual Machine

- For this week we will use your local Virtual Machine.
- <https://getfedora.org/en/workstation/>

Install Web Server and PHP

- `yum install php httpd php-mysql`
- `service httpd restart`
- `chkconfig httpd on`



Hello World!

- `cd /var/www/html`
- `vi index.php`

```
<?php
```

```
echo "Hello World";
```

```
?>
```



Web Forms

```
<html>
<body>
<form method=post action=index.php>
<input type=text name="someWord" value="">
<input type =hidden name="step" value="2">
<input type=submit name="submit" value="submit">
</form>
</body>
</html>
```


PHP Web Forms

Quotes must be escaped

```
<html>
```

```
<body>
```

```
<form method=post action=index.php>
```

```
<input type=text name=\"someWord\" value=\"\">
```

```
<input type=hidden name=\"step\" value=\"2\">
```

```
<input type=submit name=\"submit\" value=\"submit\">
```

```
</form>
```

```
</body>
```

```
</html>
```

Post Variables

- In the previous example we set variables. PHP needs to know to look for those variables.

```
isset ( $_REQUEST['i'] ) ? $i = $_REQUEST['i'] : $i = "";
```



```
<?php
isset ( $_REQUEST['i'] ) ? $i = $_REQUEST['i'] : $i = "";
echo "
<html>
  <head> <title> PHP Form Example </title> </head>
  <body>
    ";

    if ( $i == Null ) {
        echo "
        <form method=post action=index.php>
        Enter Text:
        <input type=\"text\" id=\"i\" name=\"i\">
        <input type=\"submit\" value =\"Submit\" /></p>
        </form>";
    } else {
        echo "You entered the text: $i";
    }
    echo "</body> </html>";
?>
```



Random Number

- Create a Random Number
- `rand(min, max)`
- `$x=rand(0,256);`



if else

- Comparing numbers is easy.

```
if ( variable == something ) {
```

```
    ...
```

```
} else {
```

```
    ...
```

```
}
```



Comparators

- == : Equals
- != : Not Equals
- > Greater than
- < Less Than
- >=
- <=



Assignment Due Tuesday @ Noon

- Create a page called hw1.php that asks the user for a number.
- Create a random number between 0-20.
- Display the random number and the number entered.
- State whether the number from the user is higher or lower than the random number.
- Print an error if the user enters a number outside the range of 0-20
- If they are not correct, prompt them for another number.