CSCI-3403: Cyber Security
Spring 2020

Adapted from Abigail Fernandes and Biljith

Department of Computer Science

University of Colorado Boulder

Week 3

> PHP

> PHP Demo

Motivation!

> I can't even say what's *wrong* with PHP, because— okay. Imagine you have uh, a toolbox. A set of tools. Looks okay, standard stuff in there.
>
> You pull out a screwdriver, and you see it's one of those weird tri-headed things. Okay, well, that's not very useful to you, but you guess it comes in handy sometimes.
>
> You pull out the hammer, but to your dismay, it has the claw part on *both* sides. Still serviceable though, I mean, you can hit nails with the middle of the head holding it sideways.
>
> You pull out the pliers, but they don't have those serrated surfaces; it's flat and smooth. That's less useful, but it still turns bolts well enough, so whatever.
>
> And on you go. Everything in the box is kind of weird and quirky, but maybe not enough to make it *completely* worthless. And there's no clear problem with the set as a whole; it still has all the tools.
>
> Now imagine you meet millions of carpenters using this toolbox who tell you "well hey what's the problem with these tools? They're all I've ever used and they work fine!" And the carpenters show you the houses they've built, where every room is a pentagon and the roof is upside-down. And you knock on the front door and it just collapses inwards and they all yell at you for breaking their door.
>
> That's what's wrong with PHP.

PHP Security Vulnerabilities

- **SQL Injection:** Hackers get access to databases by inserting malicious code
- **XSS (Cross Site Scripting):** Injects JavaScript code mixed with submitted content, so when a user visits a Web page with the submitted content, the malicious script gets downloaded automatically in their browser and gets executed.
- **Session Hijacking:** Session hijacking is when an attacker steals and use someone else's session ID
- **Buffer Overflows:** Buffer overflows may cause the PHP engine to execute arbitrary code that can perform security exploits

Motivated enough?

What is PHP?

1. PHP is an acronym for "PHP: Hypertext Preprocessor"
2. It is a server-side scripting language.
3. PHP files can contain text, HTML, CSS, JavaScript, and PHP code
4. PHP code is executed on the server, and the result is returned to the browser as plain HTML
5. PHP files have extension ".php"

```
<!DOCTYPE html>
<html>
<body>

<h1>My first PHP page</h1>

<?php
echo "Hello World!";
?>

</body>
</html>
```

What does a PHP file look like?

- PHP code can be embedded into an HTML file.
- We use `<?php`
- `?>` tags to separate PHP from HTML.
- Everything inside that tag is PHP code.

Installation

We'll be using the LAMP stack. **L**inux, **A**pache, **M**ySQL and **P**HP

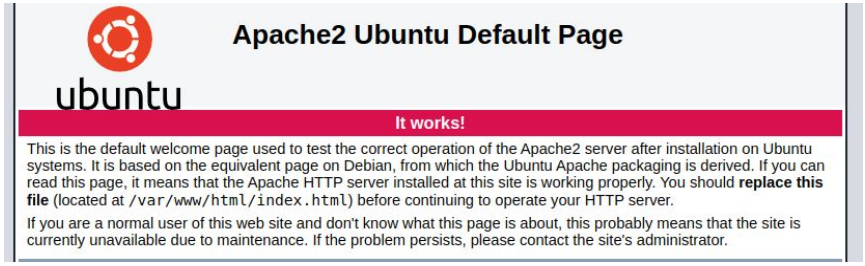
1. Install apache. Apache is the Webserver that will serve our PHP pages

```
$ sudo apt update
$ sudo apt install apache2
```

2. Install PHP

```
$ sudo apt install php libapache2-mod-php php-mysql
```

3. Visit localhost on your browser



4. Where is the document root?
   /var/www/html/

PHP Hello, World!

```
<?php $user = "John"; ?>
<html>
<head></head>
<body>
Hello <?php echo $user; ?>!
</body>
</html>
```

```php
$x = 1;
$y = 2;
$sum = $x + $y;
echo $sum;
```

```php
$name = "Jake";
echo "Your name is $name";
```

Variables

```php
$odd_numbers = [1,3,5,7,9];
for ($i = 0; $i < count($odd_numbers); $i=$i+1) {
    $odd_number = $odd_numbers[$i];
    echo $odd_number . "\n";
}
```

```php
$odd_numbers = [1,3,5,7,9];
foreach ($odd_numbers as $odd_number) {
  echo $odd_number . "\n";
}
```

Loops

Define where the data gets sent

Define how the data gets sent

```
<html>
<body>

<form action="welcome.php" method="post">
Name: <input type="text" name="name"><br>
E-mail: <input type="text" name="email"><br>
<input type="submit">
</form>

</body>
</html>
```

welcome.php

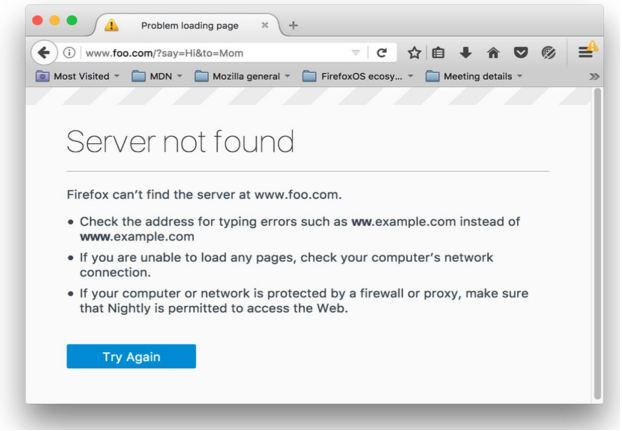
```
<html>
<body>

Welcome <?php echo $_POST["name"]; ?><br>
Your email address is: <?php echo $_POST["email"]; ?>

</body>
</html>
```

Handling form data

FORM Methods

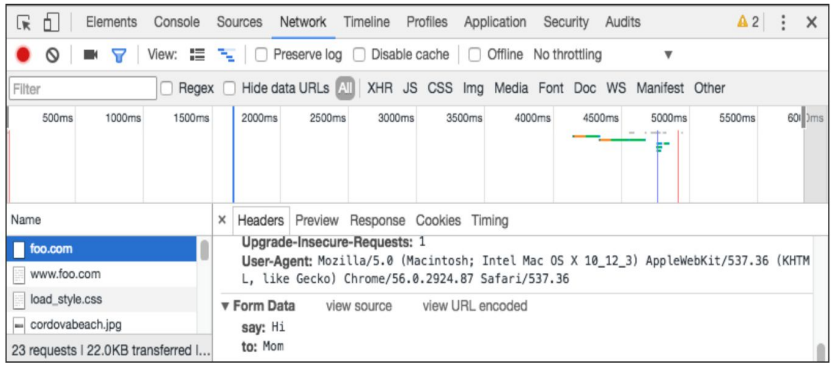
GET

- Appends form-data into the URL in name/value pairs
- The length of a URL is limited (about 3000 characters)



POST

- Appends form-data inside the body of the HTTP request
- Has no size limitations



Sessions

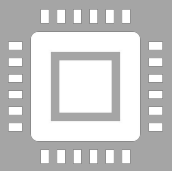
HTTP is a stateless protocol

This means that subsequent requests will not be associated with each other.

How do we allow users to login or add things to a cart?

Cookies

Cookies is one way to manage state



It usually contains a unique identifier to allow the server to recognize requests from the same server

PHP abstracts this concept. The programmer doesn't have to worry about creating or setting cookies.

```php
<?php
// Start the session
session_start();
?>
<!DOCTYPE html>
<html>
<body>

<?php
// Set session variables
$_SESSION["favcolor"] = "green";
$_SESSION["favanimal"] = "cat";
echo "Session variables are set.";
?>

</body>
</html>
```

```php
<?php
session_start();
?>
<!DOCTYPE html>
<html>
<body>

<?php
// Echo session variables that were set on previous page
echo "Favorite color is " . $_SESSION["favcolor"] . ".<br>";
echo "Favorite animal is " . $_SESSION["favanimal"] . ".";
?>

</body>
</html>
```

Sessions

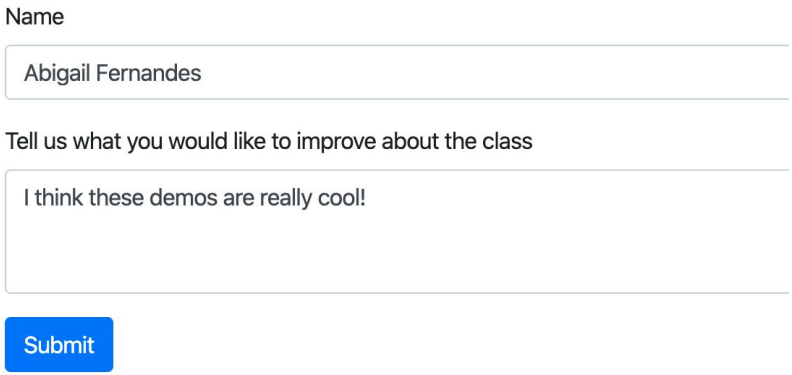
Week 3

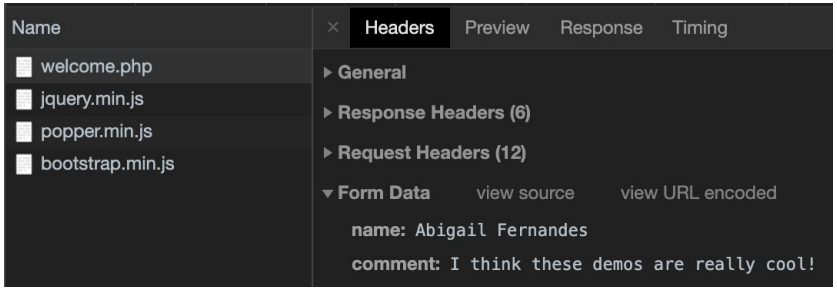
> Assignment 3 Discussion

> PHP

> **Demo** (http://csci3403.c1.biz/)

```
<form action="welcome.php" method="post">
  <div class="form-group">
    <label for="name">Name</label>
    <input type="text" class="form-control" placeholder="Enter your full name" name="name" required>
  </div>
  <div class="form-group">
    <label for="name">Tell us what you would like to improve about the class</label>
    <textarea class="form-control" placeholder="Enter some feedback..." name="comment" rows="3" required></textarea>
  </div>
    <button type="submit" class="btn btn-primary">Submit</button>
  </form>
```





Form

```php
<?php include("top.html");?>
  <ul class="media-list">

      <?php
          $fn = fopen("data.txt","r");
          while(! feof($fn))  {
                $line = fgets($fn);
                if ($line == "") continue;
                $line_array = explode("::", $line);
                echo "<li class=\"media\">
                          <div class=\"media-body\">
                            <strong class=\"text-success\">
                                    @$line_array[0]
                            </strong>
                            <p>$line_array[1]</p>
                          </div>
                      </li>";
          }

          fclose($fn);
       ?>
```

@Abigail Fernandes

Free cookies for everyone before class

@Biljith Thadichi

No Midterm and finals

@Matthew

Make it tougher!

Feedback Page

top.html

```
<html>

<head>
 <meta charset="utf-8">
 <meta name="viewport" content="width=device-width, initial-scale=1">
 <title>HTML Intro</title>
 <link href="style.css" rel="stylesheet" type="text/css" />
 <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.4.1/css/bootstrap.min.css">
 <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.1/jquery.min.js"></script>
 <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.16.0/umd/popper.min.js"></script>
 <script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.4.1/js/bootstrap.min.js"></script>
</head>
<body>

  <nav class="navbar navbar-expand-sm bg-dark navbar-dark">
    <ul class="navbar-nav">
      <li class="nav-item">
        <a class="nav-link" href="index.html">Feedback Form</a>
      </li>
      <li class="nav-item">
        <a class="nav-link" href="comments.php">View Submissions</a>
      </li>
    </ul>
  </nav>

  <div class="jumbotron">
    <h1> PHP Tutorial </h1>
    <p>PHP is a popular general-purpose scripting language that is especially suited to web development.
    languages out there (Read Node.js, GoLang)! </p>
  </div>
  <div class="container">
```

index.html

```
<?php include("top.html");?>

    <form action="welcome.php" method="post">
      <div class="form-group">
        <label for="name">Name</label>
        <input type="text" class="form-control" placeholder="Enter y
      </div>
      <div class="form-group">
        <label for="name">Tell us what you would like to improve abou
        <textarea class="form-control" placeholder="Enter some feedba
      </div>
        <button type="submit" class="btn btn-primary">Submit</button>
      </form>
    </div>

  </body>
  </html>
```

comments.php

```
<?php include("top.html");?>
  <ul class="media-list">

      <?php
          $fn = fopen("data.txt","r");
          while(! feof($fn))  {
                $line = fgets($fn);
                if ($line == "") continue;
                $line_array = explode("::", $line);
                echo "<li class=\"media\">
                          <div class=\"media-body\">
                            <strong class=\"text-success\">@$line_array[0]</strong>
                            <p>$line_array[1]</p>
                          </div>
                      </li>";
          }

          fclose($fn);
      ?>
  </ul>
 </body>
</html>
```

Templates

PHP Resources

- https://www.php.net/manual/en/tutorial.requirements.php
- Host Website (PHP, MySQL): https://www.biz.nf/