Mason Competitive Cyber

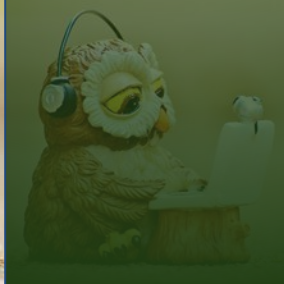
Web App Sec That's Not PHP

Announcements

- patriotCTF
  - GMU waitlist
- CCDC moving along
- CYSE discussions

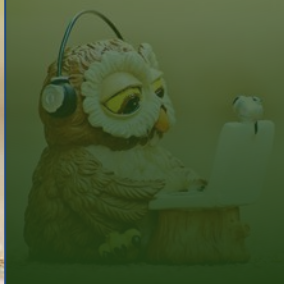
Disclosure

- To remain fair, some subject matter was restricted from this talk to ensure we aren’t covering a ton of patriotCTF problems before we have it



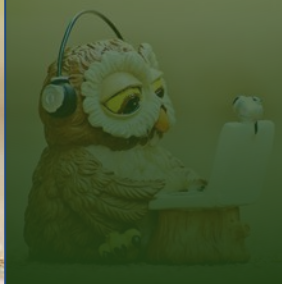
Agenda

- Refresh on old concepts
- Talk about new ones
- **Warning: This talk assumes a little on what you know in terms of *code***
- Juice Shop!!



Cross Site Scripting

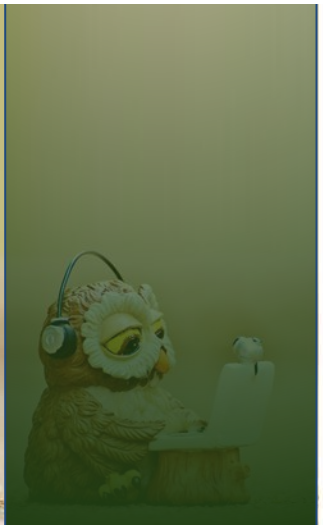
- Referred to as **XSS**
- Running your own Javascript on another user's browser
- Frequently used to impersonate users/"session hijack"
- Three kinds:
  - Stored
    - Stored on the server somewhere such as the database where it's retrieved at a later time
    - Much more dangerous, like if it's something like your First Name on a profile page
  - Reflected
    - Sent to the server and returned, such as in the URL
  - DOM-based
    - Leverages the "DOM", so basically existing Javascript or HTML



Code Example

```php
<?php

// Is there any input?
if( array_key_exists( "name", $_GET ) && $_GET[ 'name' ] != NULL ) {
    // Feedback for end user
    echo '<pre>Hello ' . $_GET[ 'name' ] . '</pre>';
}
```



SQL Injection

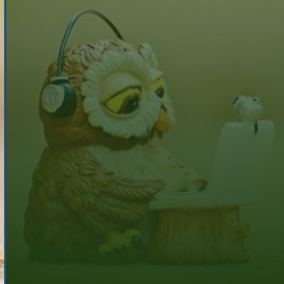
- Injecting your own code into a database statement, such as with a login
- Two major kinds:
  - Blind
    - You don't see output from what you've done
  - Normal
    - You do see your output

```
// Get input
$id = $_POST[ 'id' ];

// Check database
$query  = "SELECT first_name, last_name FROM users WHERE user_id = '$id';";
```

SQLMap

- **You don’t need to know SQL**
- Have to find vulnerable request, such as the first name or username or something
- In our VM and Kali by default
- **sqlmap --url http://yoururl.com/page?id=1 -p id --dbs**
  - **Enumerates DBs assuming id is unsafe**
- Can use it to get an OS shell, etc
- **Was used at VT Summit like last week**



Command Injection

- Similar to SQL injection but commands
- Appeared in CyberFusion

```
// Get input
$target = $_REQUEST[ 'ip' ];

$cmd = shell_exec( 'ping  ' . $target );
```

File Upload

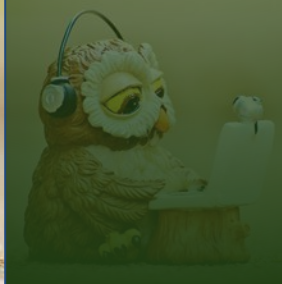
- When people don’t check their uploads and you can upload PHP
- Avoid using web shells like an idiot in competition
  - People don’t have to upload their own if they can browse to /shell.php and be done with it
- Common in defense competitions to remove
- **b374k, c99, etc** are common web shells
  - RARELY will actually get caught by AV

File Include

- When a programmer relies on user input to fetch files
- If you visit a page or an image is loaded and it's something like **page.php?file=dog.jpg**

```php
<?php

// The page we wish to display
$file = $_GET[ 'page' ];
```

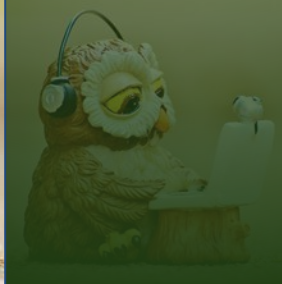


In this example, we'd need to display $file with something like **echo file_get_contents($file);**

Object Deserialization

```
app.get('/', function(req, res) {
 if (req.cookies.profile) {
   var str = new Buffer(req.cookies.profile, 'base64').toString();
   var obj = serialize.unserialize(str);
   if (obj.username) {
     res.send("Hello " + escape(obj.username));
   }
 } else {
     res.cookie('profile', "eyJ1c2VybmFtZSI6ImFqaW4iLCJjb3VudHJ5IjoiaW
       maxAge: 900000,
       httpOnly: true
     });
 }
 res.send("Hello World");
});
app.listen(3000);
```

```
var y = {
 rce : function(){
 require('child_process').exec('ls /', function(error, stdout, stderr)
 },
}
var serialize = require('node-serialize');
console.log("Serialized: \n" + serialize.serialize(y));
```



Broken Authentication

- Covers a lot
- Broken/bad sessions, low login requirements, etc
- Brute forcing, fuzzing should solve

Sensitive Data Exposure

- Is directory indexing on?
- Is data passed to the client that shouldn't be
- Is your privilege influenced by client side parameters such as user agent?
- IRL: Regulations?

XXE

- External Entities
- Note: Juice Shop is in fact in Docker, sorry
- Mileage varies massively
  - This is a framework matter

```
<?xml version="1.0" encoding="ISO-8859-1"?>
  <!DOCTYPE foo [
   <!ELEMENT foo ANY >
   <!ENTITY xxe SYSTEM "file:///dev/random" >]><foo>&xxe;</foo>
```

Juiceshop

- CTFd - localhost:8000
- App - localhost:3000