Seguridad I

Clase 16. Problemas de seguridad y soluciones

Carlos Ruiz Ruso · @micromante

Principios

Principios generales

- Muchas aplicaciones tienen fallos similares
- Los mas habituales suelen ser https://www.owasp.org/index.php/Top_10_2013-Top_10
- Ningun usuario es confiable
- Código sencillo y mantenible
- Ninguna entrada de datos es segura
- No existen fallos “sin importancia”
- Seguridad dividida por capas

Seguridad en capas en Frontend

- Frontend (buenas practicas)
  - Validación con HTML5
  - Validaciones JS
    - Formularios, peticiones, AJAX
- Frontend (problematica)
  - Pueden desactivar validación HTML5
  - Pueden “trucar” las validaciones JS y “entrar hasta la cocina”

Entonces, esta parte es importante pero no es suficiente.

Seguridad en capas en Backend

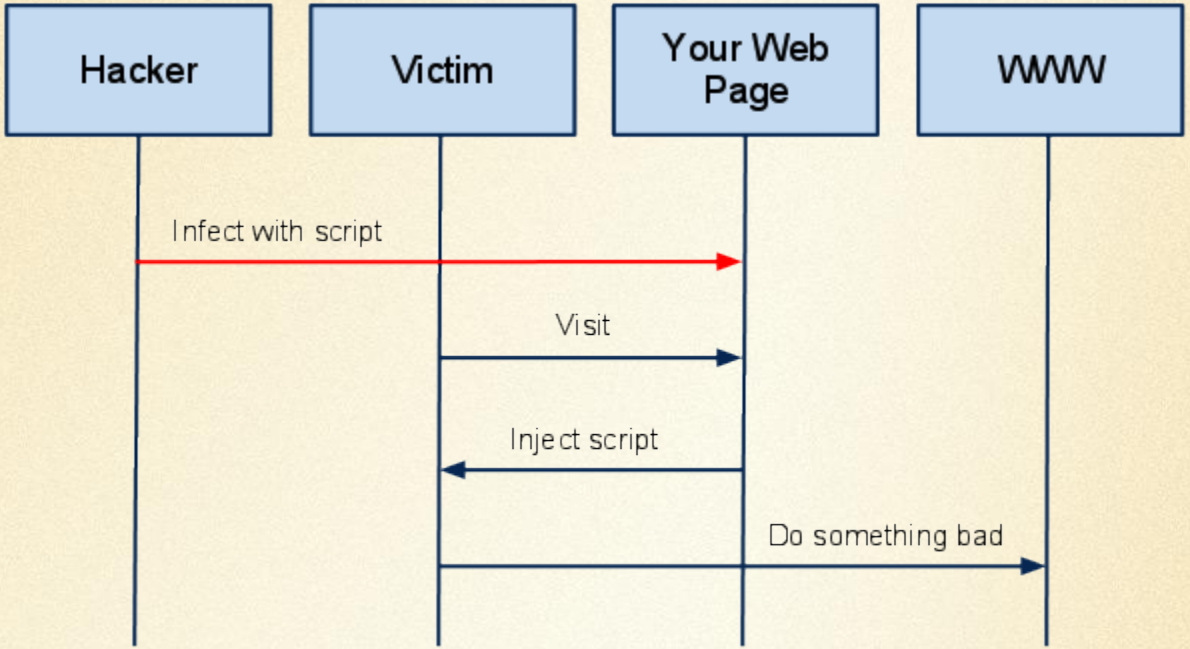
- Backend (buenas practicas)
  - Controlar entrada de datos
  - Validaciones post JS con “revalidaciones”
  - Limpiar los datos antes de insertarlos o pasarlos a una función que pueda comprometer la seguridad
- Backend (problematica a controlar y tener en cuenta)
  - Permisos de carpetas y ficheros
  - Datos BD, FTP, etc… con contraseñas seguras
  - Buena configuración del servidor por expertos
  - Control de excepciones y códigos de error que se muestran.

Seguridad en sistemas API

- ¿Que es API?
- API (buenas practicas)
  - Cifrado de datos + caducidad de sesiones
  - Proteccion CSRF (cross domain request)
  - Registro de peticiones ($_SERVER["HTTP_REFERER"])
  - TOKEN
  - Control operaciones de entrada de datos $_POST $_GET…
  - Validaciones internas

XSS Cross site Scripting

XSS Cross site Scripting



A High Level View of a typical XSS Attack

Problemas de seguridad con XSS

- Identificar la máquina o ladrón
- Acceso a información sensible
- Ganar acceso
- Espiar
- Difamación o ataque posicionamiento
- Romper seguridad y robar datos sensibles usuarios
- Ningún sistema es seguro al 100%

Problemas y soluciones

[DEMOS]

Inyecciones Javascript
Iframes en paginas (document.write… con src espia)
Formularios externos (inyectar formulario externo)
Códigos invisible al cliente
Reenvío a otras páginas con JS
Script en formulario search.php?search=<script….
Inserción de enlaces para manejar posicionamiento
Ataques externos al formulario con POSTMAN
....

Soluciones de seguridad con XSS

- Filtros PHP Validación
  - Expresiones regulares
  - Filter vars http://php.net/manual/es/filter.filters.php
- “Sanitizacion” con strip_tags + strlen
- Escape de la salida con htmlspecialchars…
- Veamos un ejemplo...

Ejemplo práctico validación, sanitización y escape

```php
<?php
// validate comment
$comment = trim($_POST["comment"]);
if (empty($comment)) {
    exit("must provide a comment");
}

// sanitize comment
$comment = strip_tags($comment);

// comment is now safe for storage
file_put_contents("comments.txt", $comment, FILE_APPEND);

// escape comments before display
$comments = file_get_contents("comments.txt");
echo htmlspecialchars($comments);
```

Filter Vars de PHP con Email

```php
$email = "clifton@example"; //Note the .com missing
echo "PHP Version: ".phpversion().'<br>';
if(filter_var($email, FILTER_VALIDATE_EMAIL)){
    echo $email.'<br>';
    var_dump(filter_var($email, FILTER_VALIDATE_EMAIL));
}else{
    var_dump(filter_var($email, FILTER_VALIDATE_EMAIL));
}
?>
```

Quiero HTML o no...

- Ejemplo práctico

```
$entrada = htmlentities(trim(strip_tags(stripslahes($entrada))), ENT_NOQUOTES, “UTF-8”);
```

¿ENT_NOQUOTES? http://www.w3schools.com/php/func_string_htmlentities.asp
¿strip_tags? https://php.net/strip_tags
¿trim? http://php.net/trim
¿htmlentities? http://php.net/htmlentities

MUCHAS GRACIAS A TODOS!

Podeis seguirme en la redes sociales como @micromante o Carlos Ruiz Ruso

www.micromante.com