

# Curso: (62612) Diseño de aplicaciones seguras

Fernando Tricas García

Departamento de Informática e Ingeniería de Sistemas  
Universidad de Zaragoza

<http://www.cps.unizar.es/~ftricas/>

<http://moodle.unizar.es/>

[ftricas@unizar.es](mailto:ftricas@unizar.es)

# A modo de conclusiones

Fernando Tricas García

Departamento de Informática e Ingeniería de Sistemas  
Universidad de Zaragoza

<http://www.cps.unizar.es/~ftricas/>

<http://moodle.unizar.es/>

[ftricas@unizar.es](mailto:ftricas@unizar.es)

# Buenas costumbres en general

- ▶ Sobre privacidad
  - ▶ Crear una declaración formal
  - ▶ Informar antes de recolectar información
  - ▶ Pedir permiso expresamente
  - ▶ No coleccionar información innecesaria
  - ▶ Dar acceso fácil a la información recolectada
  - ▶ Proteger los datos privados
  - ▶ Los niños son especiales
  - ▶ Ser cuidadoso

# Mas buenas costumbres

- ▶ Comprobar y re-comprobar, sobre todo en los errores
- ▶ Comentarios sobre seguridad en el código
- ▶ Autenticación, autorización, cifrado: mejor el SO
- ▶ No confiar en el buen juicio de los usuarios
- ▶ Los ejemplos son patrones (esqueletos)
- ▶ Nosotros igual que los usuarios!
- ▶ Si hacen falta privilegios elevados, todavía más cuidado

# Las 10 leyes inmutables de la seguridad

- ▶ Si alguien te convence para ejecutar su código en tu máquina, ya no es tu máquina
- ▶ Si alguien puede modificar el sistema operativo en tu máquina, ya no es tu máquina
- ▶ Si alguien tiene acceso físico a tu máquina, ya no es tu máquina
- ▶ Si alguien puede 'subir' programas a nuestra máquina, ya no es tu máquina

# Las 10 leyes inmutables de la seguridad

- ▶ Claves débiles estropean la mejor seguridad
- ▶ Una máquina es tan segura como confiable su administrador
- ▶ Los datos cifrados son tan seguros como la clave de descifrado
- ▶ Un anti-virus no actualizado sólo es marginalmente mejor que no tener nada
- ▶ El anonimato absoluto no es práctico, ni en la vida real ni en la web
- ▶ La tecnología no es la panacea



# Las 10 leyes inmutables de la administración de seguridad

- ▶ Nadie cree que le pueda pasar algo malo, hasta que le pasa
- ▶ La seguridad sólo funciona cuando el camino seguro es, además, el fácil
- ▶ Si no estás al tanto de las actualizaciones, tu red no seguirá siendo tuya por mucho tiempo
- ▶ No vale de nada asegurar algo que no empezó siendo seguro
- ▶ El precio de la seguridad es la constante vigilancia



# Las 10 leyes inmutables de la administración de seguridad

- ▶ Hay alguien por ahí afuera tratando de adivinar tus claves
- ▶ La red mas segura es una bien administrada
- ▶ La dificultad para defender una red es directamente proporcional a su complejidad
- ▶ La seguridad no consiste en evitar los riesgos, si no en gestionarlos
- ▶ La tecnología no es la panacea



# Excusas tontas

- ▶ Nadie lo hará!
- ▶ Por qué alguien podría hacer eso?
- ▶ Nunca hemos sufrido ataques
- ▶ Es seguro, usamos criptografía
- ▶ Es seguro, usamos ACLs
- ▶ Es seguro, tenemos cortafuegos

# Excusas tontas (mas)

- ▶ Auditamos el código, no hay fallos de seguridad
- ▶ Es el comportamiento por defecto, pero el administrador puede quitarlo
- ▶ Si no corre como administrador, no va

# OWASP Top Ten

1. Entrada sin validar
2. Control de acceso incorrecto
3. Gestión de sesiones o de autenticación incorrecto
4. Fallos de 'Cross Site Scripting'
5. Desbordamientos de memoria
6. Fallos de inyección
7. Manejo incorrecto de errores
8. Almacenamiento inseguro
9. Denegación de servicio
10. Gestión de configuraciones inseguro

Versión 2004 [http://www.owasp.org/index.php/Top\\_10\\_2004](http://www.owasp.org/index.php/Top_10_2004)

# OWASP Top Ten

1. A1 Cross Site Scripting (XSS)
2. A2 - Injection Flaws
3. A3 - Malicious File Execution
4. A4 - Insecure Direct Object Reference
5. A5 - Cross Site Request Forgery (CSRF)
6. A6 - Information Leakage and Improper Error Handling
7. A7 - Broken Authentication and Session Management
8. A8 - Insecure Cryptographic Storage
9. A9 - Insecure Communications
10. A10 - Failure to Restrict URL Access

[http://www.owasp.org/index.php/Top\\_10\\_2007](http://www.owasp.org/index.php/Top_10_2007)

[https://www.owasp.org/images/a/ae/OWASP\\_Top\\_10\\_2007\\_Spanish.pdf](https://www.owasp.org/images/a/ae/OWASP_Top_10_2007_Spanish.pdf)



# 2009 CWE/SANS 25 errores más peligrosos de programación

## Insecure Interaction Between Components

- ▶ CWE-20: Improper Input Validation
- ▶ CWE-116: Improper Encoding or Escaping of Output
- ▶ CWE-89: Failure to Preserve SQL Query Structure (aka 'SQL Injection')
- ▶ CWE-79: Failure to Preserve Web Page Structure (aka 'Cross-site Scripting')
- ▶ CWE-78: Failure to Preserve OS Command Structure (aka 'OS Command Injection')
- ▶ CWE-319: Cleartext Transmission of Sensitive Information
- ▶ CWE-352: Cross-Site Request Forgery (CSRF)
- ▶ CWE-362: Race Condition
- ▶ CWE-209: Error Message Information Leak



# 2009 CWE/SANS 25 errores más peligrosos de programación

## Risky Resource Management

- ▶ CWE-119: Failure to Constrain Operations within the Bounds of a Memory Buffer
- ▶ CWE-642: External Control of Critical State Data
- ▶ CWE-73: External Control of File Name or Path
- ▶ CWE-426: Untrusted Search Path
- ▶ CWE-94: Failure to Control Generation of Code (aka 'Code Injection')
- ▶ CWE-494: Download of Code Without Integrity Check
- ▶ CWE-404: Improper Resource Shutdown or Release
- ▶ CWE-665: Improper Initialization
- ▶ CWE-682: Incorrect Calculation



# 2009 CWE/SANS 25 errores más peligrosos de programación

## Porous Defenses

- ▶ CWE-285: Improper Access Control (Authorization)
- ▶ CWE-327: Use of a Broken or Risky Cryptographic Algorithm
- ▶ CWE-259: Hard-Coded Password
- ▶ CWE-732: Insecure Permission Assignment for Critical Resource
- ▶ CWE-330: Use of Insufficiently Random Values
- ▶ CWE-250: Execution with Unnecessary Privileges
- ▶ CWE-602: Client-Side Enforcement of Server-Side Security

<http://cwe.mitre.org/top25/>

# Los 7 reinos + 1

- ▶ Input validation and representation
- ▶ API abuse
- ▶ Security features
- ▶ Time and state
- ▶ Error handling
- ▶ Code quality
- ▶ Encapsulation
- ▶ Environment

Gary Mc Graw. Software Security: Building Security

# Pensamientos finales

- ▶ No hay sustituto para el comportamiento seguro por defecto
- ▶ No confiar en los administradores, ni en que los parches se aplicarán
- ▶ En los usuarios tampoco

# Y ahora empieza todo ...