

Curso: (30227) Seguridad Informática

Fernando Tricas García

Departamento de Informática e Ingeniería de Sistemas
Universidad de Zaragoza

<http://webdiis.unizar.es/~ftricas/>

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

ftricas@unizar.es

A modo de conclusiones

Fernando Tricas García

Departamento de Informática e Ingeniería de Sistemas
Universidad de Zaragoza

<http://webdiis.unizar.es/~ftricas/>

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

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

<http://technet.microsoft.com/en-us/library/cc722487.aspx>



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

<http://technet.microsoft.com/en-us/library/cc722488.aspx>



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

- ▶ A1 Injection
- ▶ A2 Broken Authentication and Session Management
- ▶ A3 Cross-Site Scripting (XSS)
- ▶ A4 Insecure Direct Object References
- ▶ A5 Security Misconfiguration
- ▶ A6 Sensitive Data Exposure
- ▶ A7 Missing Function Level Access Control
- ▶ A8 Cross-Site Request Forgery (CSRF)
- ▶ A9 Using Components with Known Vulnerabilities
- ▶ A10 Unvalidated Redirects and Forwards

https://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project
(2013 cambia cada tres años en principio)



2011 CWE/SANS 25 errores más peligrosos de programación

<http://cwe.mitre.org/top25/> (2011)

- ▶ Improper Neutralization of Special Elements used in an SQL Command ('SQL Injection')
- ▶ Improper Neutralization of Special Elements used in an OS Command ('OS Command Injection')
- ▶ Buffer Copy without Checking Size of Input ('Classic Buffer Overflow')
- ▶ Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting')
- ▶ Missing Authentication for Critical Function
- ▶ Missing Authorization
- ▶ Use of Hard-coded Credentials



2011 CWE/SANS 25 errores más peligrosos de programación

- ▶ Missing Encryption of Sensitive Data
- ▶ Unrestricted Upload of File with Dangerous Type
- ▶ Reliance on Untrusted Inputs in a Security Decision
- ▶ Execution with Unnecessary Privileges
- ▶ Cross-Site Request Forgery (CSRF)
- ▶ Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal')
- ▶ Download of Code Without Integrity Check
- ▶ Incorrect Authorization
- ▶ Inclusion of Functionality from Untrusted Control Sphere



2011 CWE/SANS 25 errores más peligrosos de programación

- ▶ Incorrect Permission Assignment for Critical Resource
- ▶ Use of Potentially Dangerous Function
- ▶ Use of a Broken or Risky Cryptographic Algorithm
- ▶ Incorrect Calculation of Buffer Size
- ▶ Improper Restriction of Excessive Authentication Attempts
- ▶ URL Redirection to Untrusted Site ('Open Redirect')
- ▶ Uncontrolled Format String
- ▶ Integer Overflow or Wraparound
- ▶ Use of a One-Way Hash without a Salt



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 In
'Seven Pernicious Kingdoms: A Taxonomy of Software Security Errors'

https://buildsecurityin.us-cert.gov/sites/default/files/bsi11-taxonomy_0.pdf



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 ...

