# 1. Determinación del nivel de seguridad requerido por aplicaciones.

## Caso práctico


Pixabay (Dominio público)

Julián está en un proyecto desarrollo de software en el que uno de sus componentes es una aplicación web. Julián nunca había estado en un proyecto de estas características y tiene ciertas preocupaciones con software expuesto a internet, ya que sabe que un fallo no contemplado en el código puede provocar un impacto importante (indisponibilidad, ejecución de código remota, compromiso de datos personales...).

Además las librerías y el código que va a usar está público en Github, lo cual es bueno puesto que si hay alguna vulnerabilidad hay una gran comunidad de usuarios que la detectará, la reportará y ayudará a mitigar.

Sabe además que hay una fundación (OWASP) que se encarga de evaluar los

riesgos que pueden sufrir estas aplicaciones expuestas a internet por lo que se apoya en estos frameworks para saber a qué se enfrenta y cómo mitigar cualquier futuro problema.

El desarrollo de software ha evolucionado mucho durante los últimos años, han surgido fundaciones como OWASP cuyo objetivo es la mejora en la seguridad del software mediante la identificación de las principales vulnerabilidades existentes y como poder afrontarlas.

También ha evolucionado mucho el desarrollo de software a nivel de exposición, ya que a día de hoy muchas de las líneas de código de las aplicaciones que vemos en internet están disponibles en repositorios públicos, lo cual aporta grandes beneficios como veremos más adelante pero también añade una exposición adicional al código, que cómo veíamos en unidades anteriores hay una gran relación entre el grado de exposición y el número de vulnerabilidades.

# 1.1.- Fuentes abiertas para desarrollo seguro.

Cuando hablamos de software de fuentes abiertas nos referimos a aquel software que podemos encontrar de fácil acceso, publicado y por lo general libre (el código fuente se pone a disposición del público para ser usado y modificado conforme los usuarios o desarrolladores puedan aportar).

Durante los últimos años ha habido una evolución clara, ya que hasta ahora el software que usábamos tanto a nivel doméstico como profesional era software cerrado o propietario, es decir, aquel en el que no se permite el acceso al código fuente.

Estos últimos años se ha demostrado que el software de fuentes abiertas aporta una serie de ventajas tanto a usuarios finales como a empresas y corporaciones de forma clara:

Unsplash (Dominio público)

- **Estabilidad**: al ser el acceso libre, los usuarios y programadores expertos pueden revisar el código y ayudar a mejorarlo, mejorando por tanto la estabilidad del software. Si consideramos que hay sistemas operativos abiertos hace que estos sistemas tengan comunidades detrás que los mejoren día tras día. Mayor estabilidad es también software más seguro, que coteje mejor los datos con los que trabaja la aplicación, por tanto afecta tanto al usuario final como las empresas desarrolladoras.

- **Coste:** cuando hay una comunidad tan importante de usuarios detrás del código, los costes de desarrollo del mismo se reducen, ya que la comunidad de usuarios interactuando con el código reduce los costes de desarrollo, depuración, testeo, etc.

- **Favorece la innovación colectiva** ya que incentiva la acción de la comunidad de usuarios detrás del código, revisando, mejorándolo.

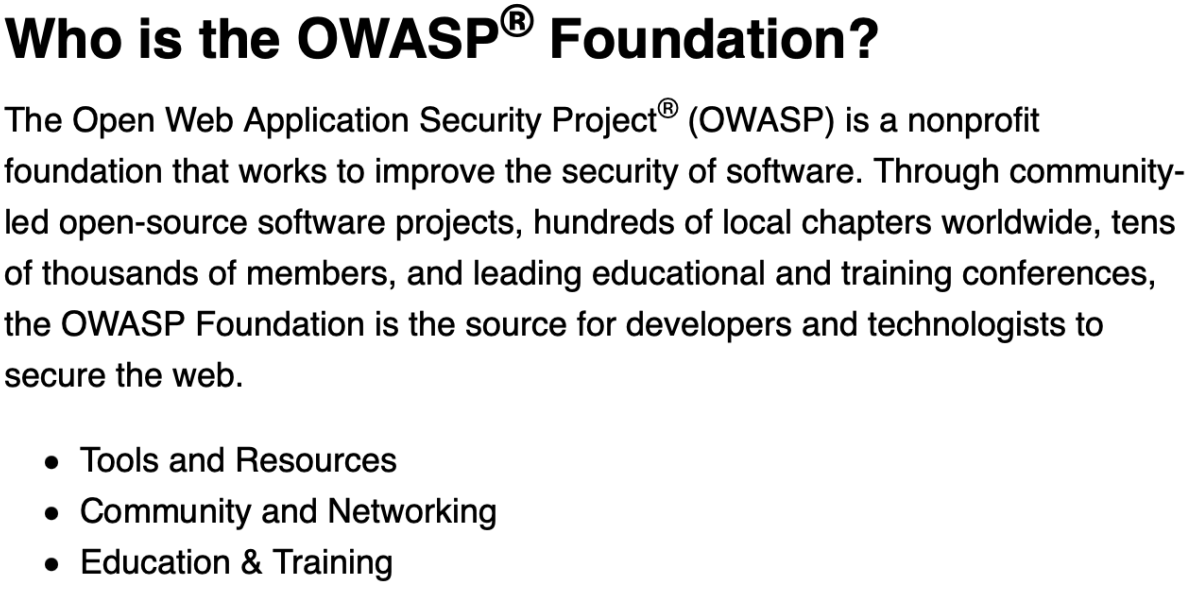
- **Mejoras:** muchos usuarios cuando interactúan con el código lo adaptan y modifican dándole nuevos usos y capacidades, por lo tanto al nivel el software original es depurado y potenciado.

La idea y el movimiento detrás del software de fuentes abiertas ha sido aceptada, aplicada, promovida y difundida por todo el mundo. Haciendo que muchas empresas de desarrollo de software lo hayan incorporado como parte de su ADN y por tanto el paradigma a nivel de desarrollo de software ha cambiado de forma radical. Como hemos comentado, al final este proceso no solamente beneficia a los usuarios finales y empresas sino también

## Para saber más

El incremento del uso de software abiertas ha sido un autentico cambio de paradigma, aquí en este video de Youtube de la prestigiosa cadena de televisión CNBC explican todas las causas de este fenómeno y las ventajas que este modelo ha aportado, convirtiéndose casi en el estándar actual para muchas aplicaciones, empresas y servicios.

# 1.2.- Listas de riesgos de seguridad habituales.




OWASP.org (Captura Pantalla)

La Fundación OWASP (Open Web Application Security Project®) es un organismo sin ánimo de lucro cuyo único objetivo es trabajar en mejorar la seguridad del software. A través de proyectos de software de código abierto (y fuentes abiertas) liderados por la comunidad, decenas de miles de miembros, conferencias educativas y cientos de cursos anuales de capacitación líderes, la Fundación OWASP es uno de los principales motores en la mejora del software en entornos web.

Esta organización publica de forma anual, desde el año 2003, el Top10 de riesgos a nivel software en entornos web. El proceso para confeccionar esta lista no es un proceso sencillo ya que se tienen en cuenta y se evalúan distintos parámetros cómo las vulnerabilidades, ataques conocidos, impactos, manejo de los datos, etc

1.- A1 - Pérdida Control Acceso
2.- A2 - Fallos Criptográficos
3.- A3 - Inyección de código
4.- A4 - Diseño Inseguro
5.- A5 - Problemas Configuración a nivel de Seguridad
6.- A6 - Componentes Vulnerables y Obsoletos
7.- A7 - Fallos de Autenticación e Identificación
8.- A8 - Fallos de Software y de Integridad de los Datos
9.- A9 - Fallos en el Logado y Monitorización de la Seguridad
10.- *A10 - Server-Side Request Forgery*

# Debes conocer

OWASP no solamente trabaja en identificar riesgos a nivel de web sino también elabora una lista para dispositivos móviles. Puedes encontrar mas información y la última versión de la lista en la web de OWASP.

# Autoevaluación

Identifica si las siguientes frases son verdaderas o falsas

OWASP es una organización que busca obtener beneficios económicos mejorando el software.

○ Verdadero ○ Falso

**Falso**

OWASP es una organización sin ánimo de lucro

El primer riesgo detectado por OWASP es un diseño inseguro

○ Verdadero ○ Falso

**Falso**

Si bien OWASP valora el diseño seguro dentro de su Top10, el primer lugar de la lista es el control del acceso.

OWASP valora vulnerabilidades y sobretodo el acceso a los datos que pudieran provocar.

○ Verdadero ○ Falso

**Verdadero**

Para la elaboración de la lista de OWASP intervienen miles de programadores,

empresas y expertos en seguridad y desarrollo.

○ Verdadero ○ Falso

**Verdadero**

# 1.3.- Comprobaciones de seguridad a nivel de aplicación.


Pixabay (Dominio público)

La fundación OWASP como hemos visto no solamente ayuda a disponer de un software más seguro mediante la identificación de los principales riesgos a los que estamos expuestos sino que también ha arrancado otro proyecto llamada ASVS.

El proyecto del Estándar de Verificación de Seguridad de Aplicaciones (*Application Security Verification Standard - ASVS*) de OWASP proporciona a los desarrolladores información para poder probar los controles técnicos de seguridad de las aplicaciones web y también proporciona a los desarrolladores una lista de requisitos para un desarrollo correcto y seguro, minimizando el impacto de los riesgos anteriormente detectados.

ASVS tiene dos objetivos principales:

- Ayudar a las organizaciones en el desarrollo y mantenimiento aplicaciones seguras.
- Permitir la alineación entre las necesidades y ofertas de los servicios de seguridad, proveedores de herramientas de seguridad y consumidores

Los requisitos se desarrollaron con los siguientes objetivos en mente:

- Usarlo como **métrica**: proporcione a los desarrolladores y propietarios de aplicaciones un criterio con el que evaluar el grado de confianza que se puede depositar en sus aplicaciones web,
- Utilizarlo como **guía**: proporcionar orientación a los desarrolladores de controles de

seguridad sobre qué incorporar en los controles de seguridad para satisfacer los requisitos de seguridad de las aplicaciones
- Usarlo durante la **compra de software** como **requisito**: proporciona una base para especificar los requisitos de verificación de seguridad de la aplicación en los contratos.

El ASVS define 3 niveles de Seguridad:

- ASVS nivel 1 se encuentra dirigido a todo tipo de software.
- ASVS nivel 2 es para aplicaciones que contienen datos sensibles, que requieren protección.
- ASVS nivel 3 es para las aplicaciones más críticas - aplicaciones que realizan transacciones económicas, contienen datos médicos confidenciales, o cualquier aplicación que requiera el más alto nivel de confianza.

## Debes conocer

Aunque la versión 5.0 ha sido anunciada, la última versión del estándar ASVS es la versión 4.0.3 que puedes encontrar en el siguiente enlace. Es importante que el alumno entienda no solamente los controles en sí, sino como según el tipo de aplicación que estemos desarrollando que nivel necesitaremos cumplir. Por otra parte el nivel 1 se considera el mínimo que debería de cumplir cualquier aplicación, por lo que es importante que el alumno se familiarice con este nivel.

# 1.4.- Requisitos de verificación necesarios asociados al nivel de seguridad establecido.

Pixabay (Dominio público)

El ASVS es un esfuerzo comunitario por establecer un marco de referencia para los requisitos de seguridad, controles funcionales y no funcionales necesarios al diseñar, desarrollar y testear aplicaciones web modernas.

El listado de controles generales en su última versión es el siguiente:

- V1. Arquitectura, diseño y modelado de amenazas
- V2. Autenticación
- V3. Gestión de sesiones
- V4. Control de acceso
- V5. Validación, sanitización y codificación
- V6. Criptografía en el almacenamiento
- V7. Gestión y registro de errores
- V8. Protección de datos
- V9. Comunicaciones

- V10. Código Malicioso
- V11. Lógica de negocio
- V12. Archivos y recursos
- V13. Servicios Web y API
- V14. Configuración

Dentro de cada uno de estos controles encontraremos subíndices con aspectos mas específicos dentro de ese control. Finalmente para cada nivel del ASVS tendremos un listado de que subíndices debemos de cumplir por cada control.

## Debes conocer

Es interesante que el alumno entienda la relación entre el dato y el riesgo, es decir cuando nuestra aplicación maneje datos sensibles, personales o transacciones económicas el nivel del ASVS subirá y por tanto los controles tambien lo harán.

Por tanto cuando un programador o empresa de software recibe el encargo de realizar una aplicación que vaya a manejar este tipo de datos es importante que entienda que deberá seguir unos controles rigurosos y por tanto los tiempos, costes y ciclos de desarrollo y testeo variarán.