SAD - U4.2. Seguridad en aplicaciones

Descargar estos apuntes

Índice

1. Seguridad en aplicaciones

La **seguridad en aplicaciones** es una disciplina que se encarga de proteger las aplicaciones informáticas de posibles amenazas. La seguridad en aplicaciones abarca diferentes aspectos, como la seguridad del **código**, la seguridad de los **datos**, la seguridad de las **comunicaciones**, la seguridad de los **usuarios**, etc. La seguridad en aplicaciones es fundamental para garantizar la confidencialidad, la integridad y la disponibilidad de las aplicaciones informáticas.

Las principales, aunque nos las únicas, aplicaciones que tienen que tener en cuenta la seguridad son las aplicaciones web, ya que están expuestas a Internet y son un objetivo común de los ciberdelincuentes. Las aplicaciones web pueden ser vulnerables a diferentes tipos de ataques, como la inyección de código, la fuga de información, la suplantación de identidad, la denegación de servicio, etc.

Existe una entidad que se encarga de la seguridad en aplicaciones web, la **OWASP** (Open Web Application Security Project), que proporciona una serie de guías y herramientas para mejorar la seguridad de las aplicaciones web.

La OWASP publica una lista de las 10 vulnerabilidades más críticas en las aplicaciones web, como la inyección de código, la fuga de información, la autenticación y autorización débiles, etc.

1.1. Seguridad del código

La **seguridad del código** es una disciplina que se encarga de proteger el código de las aplicaciones informáticas de posibles vulnerabilidades. La seguridad del código abarca diferentes aspectos, como la seguridad de las entradas, la seguridad de las salidas, la seguridad de las variables, la seguridad de las funciones, etc. La seguridad del código es fundamental para garantizar la confidencialidad, la integridad y la disponibilidad de las aplicaciones informáticas.

Algunas de las vulnerabilidades más comunes en el código de las aplicaciones informáticas son:

- **Inyección de código**: las vulnerabilidades de inyección de código permiten a un atacante ejecutar código malicioso en una aplicación, como SQL injection, XSS, CSRF, etc.
- **Fuga de información**: las vulnerabilidades de fuga de información permiten a un atacante acceder a información sensible de una aplicación, como contraseñas, datos personales, etc.
- **Desbordamiento de búfer**: las vulnerabilidades de desbordamiento de búfer permiten a un atacante sobrescribir la memoria de una aplicación y ejecutar código malicioso.

- **Autenticación y autorización débiles**: las vulnerabilidades de autenticación y autorización débiles permiten a un atacante acceder a una aplicación sin autorización, como contraseñas débiles, permisos incorrectos, etc.
- **Vulnerabilidades de configuración**: las vulnerabilidades de configuración permiten a un atacante acceder a una aplicación debido a una configuración incorrecta, como puertos abiertos, servicios activos, etc.

> **Info**
>
> La seguridad del código es fundamental para garantizar la confidencialidad, la integridad y la disponibilidad de las aplicaciones informáticas. Algunas de las prácticas recomendadas para mejorar la seguridad del código son:
>
> - Validar las entradas de los usuarios para prevenir la inyección de código.
> - Escapar las salidas de los usuarios para prevenir la fuga de información.
> - Limitar los privilegios de los usuarios para prevenir el desbordamiento de búfer.
> - Utilizar contraseñas seguras y cifradas para prevenir la autenticación y autorización débiles.
> - Mantener actualizado el software y la configuración de la aplicación para prevenir las vulnerabilidades de configuración.

1.2. Seguridad de los datos

La **seguridad de los datos** es una disciplina que se encarga de proteger los datos de las aplicaciones informáticas de posibles amenazas. La seguridad de los datos abarca diferentes aspectos, como la seguridad de los datos en reposo, la seguridad de los datos en tránsito, la seguridad de los datos en uso, la seguridad de los datos en copia de seguridad, etc. La seguridad de los datos es fundamental para garantizar la confidencialidad, la integridad y la disponibilidad de los datos de las aplicaciones informáticas.

Algunas de las amenazas más comunes a la seguridad de los datos de las aplicaciones informáticas son:

- **Fuga de información**: las fugas de información permiten a un atacante acceder a datos sensibles de una aplicación, como contraseñas, datos personales, etc.
- **Robo de información**: el robo de información permite a un atacante robar datos sensibles de una aplicación, como contraseñas, datos personales, etc.
- **Pérdida de información**: la pérdida de información permite a un atacante eliminar datos sensibles de una aplicación, como contraseñas, datos personales, etc.
- **Modificación de información**: la modificación de información permite a un atacante modificar datos sensibles de una aplicación, como contraseñas, datos personales, etc.

> **Info**
>
> La seguridad de los datos es fundamental para garantizar la confidencialidad, la integridad y la disponibilidad de los datos de las aplicaciones informáticas. Algunas de las prácticas recomendadas para mejorar la seguridad de los datos son:
>
> - Cifrar los datos en reposo para prevenir la fuga de información.
> - Cifrar los datos en tránsito para prevenir el robo de información.
> - Cifrar los datos en uso para prevenir la pérdida de información.
> - Realizar copias de seguridad de los datos para prevenir la pérdida de información.
> - Verificar la integridad de los datos para prevenir la modificación de información.

1.3. Seguridad de las comunicaciones

La **seguridad de las comunicaciones** es una disciplina que se encarga de proteger las comunicaciones de las aplicaciones informáticas de posibles amenazas. La seguridad de las comunicaciones abarca diferentes aspectos, como la seguridad de las conexiones, la seguridad de los protocolos, la seguridad de los certificados, la seguridad de los cifrados, etc. La seguridad de las comunicaciones es fundamental para garantizar la confidencialidad, la integridad y la disponibilidad de las comunicaciones de las aplicaciones informáticas.

Algunas de las amenazas más comunes a la seguridad de las comunicaciones de las aplicaciones informáticas son:

- **Intercepción de comunicaciones**: la intercepción de comunicaciones permite a un atacante acceder a las comunicaciones de una aplicación, como contraseñas, datos personales, etc.
- **Modificación de comunicaciones**: la modificación de comunicaciones permite a un atacante modificar las comunicaciones de una aplicación, como contraseñas, datos personales, etc.
- **Suplantación de identidad**: la suplantación de identidad permite a un atacante hacerse pasar por otro usuario en las comunicaciones de una aplicación, como contraseñas, datos personales, etc.
- **Denegación de servicio**: la denegación de servicio permite a un atacante interrumpir las comunicaciones de una aplicación, como contraseñas, datos personales, etc.

> **Info**
>
> La seguridad de las comunicaciones es fundamental para garantizar la confidencialidad, la integridad y la disponibilidad de las comunicaciones de las aplicaciones informáticas. Algunas de las prácticas recomendadas para mejorar la seguridad de las comunicaciones son:
>
> - Utilizar conexiones seguras, como HTTPS, para prevenir la intercepción de comunicaciones.
> - Utilizar protocolos seguros, como TLS, para prevenir la modificación de comunicaciones.
> - Utilizar certificados seguros, como SSL, para prevenir la suplantación de identidad.
> - Utilizar cifrados seguros, como AES, para prevenir la denegación de servicio.

La **seguridad en redes** es una disciplina que se encarga de proteger la información y los sistemas informáticos de posibles amenazas. La seguridad en redes abarca diferentes aspectos, como la seguridad física, la seguridad lógica, la seguridad de la información y la seguridad de los sistemas. La seguridad en redes es fundamental para garantizar la confidencialidad, la integridad y la disponibilidad de la información y los sistemas informáticos.

2.1. Inyección de código

La **inyección de código** es una vulnerabilidad de seguridad que permite a un atacante ejecutar código malicioso en una aplicación.

La inyección de código puede ocurrir en diferentes partes de una aplicación, como en las **entradas de los usuarios**, en las salidas de los usuarios, en las variables de la aplicación, en las funciones de la aplicación, etc. La inyección de código es una de las vulnerabilidades más comunes en las aplicaciones informáticas y puede tener graves consecuencias, como la ejecución de código malicioso, la fuga de información, la pérdida de información, etc.

Algunos ejemplos de inyección de código son:

- **SQL injection**: la inyección de código SQL permite a un atacante ejecutar código SQL malicioso en una aplicación para acceder a la base de datos, modificar los datos, eliminar los datos, etc.

- **XSS (Cross-Site Scripting)**: la inyección de código XSS permite a un atacante ejecutar código JavaScript malicioso en una aplicación para robar información, modificar la página, redirigir a otra página, etc.
- **CSRF (Cross-Site Request Forgery)**: la inyección de código CSRF permite a un atacante ejecutar código malicioso en una aplicación para realizar acciones no autorizadas, como enviar correos electrónicos, realizar transferencias bancarias, etc.
- **Shell injection**: la inyección de código shell permite a un atacante ejecutar comandos del sistema operativo en una aplicación para obtener acceso al sistema, robar información, dañar los sistemas, etc.

2.1.1. SQL injection

La **inyección de código SQL** es una vulnerabilidad de seguridad que permite a un atacante ejecutar código SQL malicioso en una aplicación para acceder a la base de datos, modificar los datos, eliminar los datos, etc.

La inyección de código SQL puede ocurrir en diferentes partes de una aplicación, como en los formularios de entrada, en las consultas de la base de datos, en las salidas de la aplicación, etc.

Ejemplo de inyección de código SQL:

```
SELECT * FROM users WHERE username = 'admin' AND password = 'password' OR '1'='1';
```

En este ejemplo, el atacante ha inyectado el código `' OR '1'='1` en el campo de la contraseña para acceder a la cuenta de administrador sin conocer la contraseña.

> ⚠ **Conocimiento del SGBD que se quiere atacar**
>
> Para realizar una inyección de código SQL es necesario conocer el sistema de gestión de bases de datos (SGBD) que se quiere atacar, como MySQL, PostgreSQL, SQL Server, Oracle, etc.
>
> En la web de PenTestMonkey podemos encontrar una lista de consultas básicas para posibles inyecciones de código SQL para diferentes SGBD en SQL Injection Cheat Sheet
>
> Además, se puede consultar la referencia básica de los comandos SQL para otros SGBD que no aparezcan en el enlace anterior

En Kali Linux tenemos herramientas como `sqlmap` que nos permiten realizar una inyección de código SQL en una aplicación para acceder a la base de datos, modificar los datos, eliminar los datos, etc.

Podemos ver ejemplos de uso de `sqlmap` para realizar una inyección de código SQL en una aplicación en la web SQL Injection.

2.1.2. XSS (Cross-Site Scripting)

Un ataque informático es un intento de acceder, dañar o robar información de un sistema informático sin autorización. Los ataques informáticos pueden ser realizados por personas malintencionadas, como hackers, crackers o ciberdelincuentes, con el objetivo de obtener beneficios económicos, políticos o personales. Los ataques informáticos pueden ser realizados a través de diferentes métodos, como la ingeniería social, el phishing, el malware, el ransomware o la explotación de vulnerabilidades.

Los ataques informáticos suelen seguir una serie de fases, que pueden variar en función del tipo de ataque y del objetivo del atacante. Algunas de las fases más comunes de un ataque informático son:

- **Reconocimiento** y **escaneo**: el atacante recopila información sobre el sistema objetivo, como direcciones IP, nombres de dominio, usuarios, contraseñas, puertos abiertos, servicios activos, etc.
  Una vez conocidos algunos objetivos, el atacante identifica los puntos débiles del sistema objetivo, como vulnerabilidades, puertos abiertos, servicios activos, etc.
- **Prepación**: el atacante prepara los recursos necesarios para llevar a cabo el ataque, como herramientas, scripts, malware, etc.
- **Distribución**: el atacante distribuye el malware o el ataque a través de diferentes medios, como correos electrónicos, mensajes, enlaces, etc.
- **Explotación**: el atacante explota las vulnerabilidades del sistema objetivo para obtener acceso no autorizado, robar información o dañar los sistemas.
- **Instalación**: el atacante instala el malware o el ataque en el sistema objetivo para mantener el acceso y control del sistema.
- **Comando y control**: el atacante envía comandos al sistema objetivo para controlarlo, robar información, dañar los sistemas, etc.
- **Acciones sobre el objetivo**: el atacante realiza acciones sobre el sistema objetivo, como robar información, dañar los sistemas, bloquear el acceso, etc.