

Seguridad en la Red Informática Mundial

Top-Ten Vulnerabilidades según OWASP

Semana 12 clases 23 y 24

Mtra. María Noemí Araiza Ramírez

Top-Ten Vulnerabilidades según OWASP



Objetivos:

¿Qué es OWASP Top 10?

¿Qué objetivos y motivaciones tiene?

¿Qué ha cambiado de 2013 a 2017?

¿Qué son los Riesgos de Seguridad de Aplicaciones?

¿Conocer cómo me afectan o cuál es mi riesgo?

¿Conocer los 10 Riesgos más importantes?

OWASP Top 10 2013	±	OWASP Top 10 2017
A1 – Inyección	→	A1:2017 – Inyección
A2 – Pérdida de Autenticación y Gestión de Sesiones	→	A2:2017 – Pérdida de Autenticación y Gestión de Sesiones
A3 – Secuencia de Comandos en Sitios Cruzados (XSS)	↘	A3:2017 – Exposición de Datos Sensibles
A4 – Referencia Directa Insegura a Objetos [Unido+A7]	U	A4:2017 – Entidad Externa de XML (XXE) [NUEVO]
A5 – Configuración de Seguridad Incorrecta	↘	A5:2017 – Pérdida de Control de Acceso [Unido]
A6 – Exposición de Datos Sensibles	↗	A6:2017 – Configuración de Seguridad Incorrecta
A7 – Ausencia de Control de Acceso a las Funciones [Unido+A4]	U	A7:2017 – Secuencia de Comandos en Sitios Cruzados (XSS)
A8 – Falsificación de Peticiones en Sitios Cruzados (CSRF)	✗	A8:2017 – Deserialización Insegura [NUEVO, Comunidad]
A9 – Uso de Componentes con Vulnerabilidades Conocidas	→	A9:2017 – Uso de Componentes con Vulnerabilidades Conocidas
A10 – Redirecciones y reenvíos no validados	✗	A10:2017 – Registro y Monitoreo Insuficientes [NUEVO, Comunidad]

Top-Ten Vulnerabilidades según OWASP



¿Cuál es mi Riesgo?

Top 10 2013

Agente de Amenaza	Vectores de Ataque	Prevalencia de Debilidades	Detectabilidad de Debilidades	Impacto Técnico	Impacto al Negocio
Específico de la aplicación	Fácil	Difundido	Fácil	Severo	Específico de la aplicación /negocio
	Promedio	Común	Promedio	Moderado	
	Difícil	Poco Común	Difícil	Menor	

Agente de Amenaza	Explotabilidad	Prevalencia de Vulnerabilidad	Detección de Vulnerabilidad	Impacto Técnico	Impacto de Negocio
Específico de la Aplicación	Fácil 3	Difundido 3	Fácil 3	Severo 3	Específico del Negocio
	Promedio 2	Común 2	Promedio 2	Moderado 2	
	Difícil 1	Poco Común 1	Difícil 1	Mínimo 1	

Top 10 2017

A4

Referencia directa insegura a objetos

Recordemos que está ubicada en la tabla de amenazas de 2013, para 2017 esta amenaza pasa a ser la amenaza 5, pero recordemos también que se unieron la amenaza 4 con la 7 del 2013, quedando sólo una en 2017 que es la siguiente:

A5
:2017

Pérdida de Control de Acceso

Top-Ten Vulnerabilidades según OWASP



A4

Referencia directa insegura a objetos

Una referencia directa a objetos ocurre cuando un desarrollador expone una referencia a un objeto de implementación interno, tal como un fichero, directorio o base de datos.

Sin un chequeo de control de acceso u otra protección, los atacantes pueden manipular estas referencias para acceder datos no autorizados.

A4

Referencia directa insegura a objetos

Este es un problema de autorización, que resulta muy sencillo de explotar.


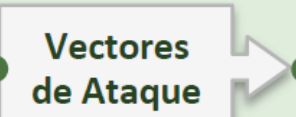
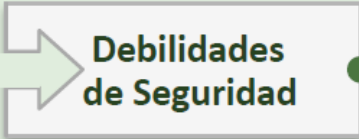
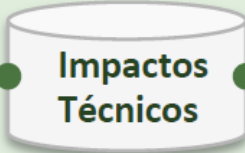

Como resultado, el usuario es capaz de acceder a un objeto del sistema al que no está autorizado, teniendo como impacto la exposición de datos privados hacia el resto de usuarios de la aplicación.

Ocurre cuando un desarrollador expone al exterior una referencia a un objeto de implementación interno, tal como un archivo, directorio, o base de datos.

Top-Ten Vulnerabilidades según OWASP

A4

Referencia directa insegura a objetos

 Agentes de Amenaza	 Vectores de Ataque	 Debilidades de Seguridad		 Impactos Técnicos	 Impactos al negocio
Específico de la Aplicación	Explotabilidad FÁCIL	Prevalencia COMÚN	Detección FÁCIL	Impacto MODERADO	Específico de la aplicación/negocio
Considere los tipos de usuarios en su sistema. ¿Existen usuarios que tengan únicamente acceso parcial a determinados tipos de datos del sistema?	Un atacante, como usuario autorizado en el sistema, simplemente modifica el valor de un parámetro que se refiere directamente a un objeto del sistema por otro objeto para el que el usuario no se encuentra autorizado. ¿Se concede el acceso?	Normalmente, las aplicaciones utilizan el nombre o clave actual de un objeto cuando se generan las páginas web. Las aplicaciones no siempre verifican que el usuario tiene autorización sobre el objetivo. Esto resulta en una vulnerabilidad de referencia de objetos directos inseguros. Los auditores pueden manipular fácilmente los valores del parámetro para detectar estas vulnerabilidades. Un análisis de código muestra rápidamente si la autorización se verifica correctamente.		Dichas vulnerabilidades pueden comprometer toda la información que pueda ser referida por parámetros. A menos que el espacio de nombres resulte escaso, para un atacante resulta sencillo acceder a todos los datos disponibles de ese tipo.	Considere el valor de negocio de los datos afectados o las funciones de la aplicación expuestas. También considere el impacto en el negocio de la exposición pública de la vulnerabilidad.

A4

Referencia directa insegura a objetos

¿Soy vulnerable?

La mejor manera de poder comprobar si una aplicación es vulnerable a esta amenaza, es verificar que todas las referencias a objetos tienen las protecciones apropiadas.

Para conseguir esto, debemos considerar:

1. Para referencias directas a recursos restringidos la aplicación necesitaría verificar si el usuario está autorizado a acceder al recurso en concreto que solicita

A4

Referencia directa insegura a objetos

¿Soy vulnerable?

2. Si la referencia es una referencia indirecta, la correspondencia con la referencia directa debe ser limitada a valores autorizados para el usuario en concreto.

Un análisis del código de la aplicación serviría para verificar rápidamente si dichas propuestas se implementan con seguridad.

A4

Referencia directa insegura a objetos

¿Soy vulnerable?

También es efectivo realizar comprobaciones para identificar referencias a objetos directos y si estos son seguros.

Normalmente las herramientas automáticas no detectan este tipo vulnerabilidades porque no son capaces de reconocer cuáles necesitan protección o cuáles son seguros e inseguros.

A4

Referencia directa insegura a objetos

¿Cómo se puede evitar?

Prevenir referencias inseguras a objetos directos requiere seleccionar una manera de proteger los objetos accesibles por cada usuario.

Dos alternativas:

La primera es utilizar referencias indirectas por usuario o sesión, esto evitaría que los atacantes accedieran directamente a recursos no Autorizados.

A4

Referencia directa insegura a objetos

¿Cómo se puede evitar?

Por ejemplo, en vez de utilizar la clave del recurso de base de datos, se podría utilizar una lista de 6 recursos que utilizase los números del 1 al 6 para indicar cuál es el valor elegido por el usuario.

La aplicación tendría que realizar la correlación entre la referencia indirecta con la clave de la base de datos correspondiente en el servidor ESAPI de OWASP incluye relaciones tanto secuenciales como aleatorias de referencias de acceso que los desarrolladores pueden utilizar para eliminar las referencias directas a objetos.

A4

Referencia directa insegura a objetos

¿Cómo se puede evitar?

La segunda alternativa es comprobar el acceso, cada uso de una referencia directa a un objeto de una fuente que no es de confianza, debe incluir una comprobación de control de acceso, para asegurar que el usuario está autorizado a acceder al objeto solicitado.

A4

Referencia directa insegura a objetos

Ejemplos de escenarios de ataques

La aplicación utiliza datos no verificados en una llamada SQL que accede a la información sobre la cuenta.

```
String query ="SELECT * FROM accts WHERE account=?";
```

```
PreparedStatement
```

```
pstmt connection.prepareStatement(query,...);
```

```
Pstmt.setString(1, request.getParameter ("acct"));
```

```
ResultSet results = pstmt.executeQuery();
```

A4

Referencia directa insegura a objetos

Ejemplos de escenarios de ataques

El atacante simplemente modificaría el parámetro acct en su navegador para enviar cualquier número de cuenta que quiera.

Si esta acción no se verifica, el atacante podría acceder a cualquier cuenta de usuario, en vez de a su cuenta de cliente correspondiente

`http://example.com/app/accountInfo?acct=notmyacct`

A4
:2017

Entidades Externas XML (XXE)

Top-Ten Vulnerabilidades según OWASP



A4
:2017

Entidades Externas XML (XXE)

Muchos procesadores XML antiguos o mal configurados evalúan referencias a entidades externas en documentos XML.

Las entidades externas pueden utilizarse para revelar archivos internos mediante la URI o archivos internos en servidores no actualizados, escanear puertos de la LAN, ejecutar código de forma remota y realizar ataques de denegación de servicio (DoS).

Top-Ten Vulnerabilidades según OWASP



A4
:2017

Entidades Externas XML (XXE)

Un ataque de entidad externa XML es un tipo de ataque contra una aplicación que analiza la entrada XML.

Este ataque ocurre cuando la entrada XML que contiene una referencia a una entidad externa es procesada por un analizador XML débilmente configurado.

Top-Ten Vulnerabilidades según OWASP



A4
:2017

Entidades Externas XML (XXE)

Este ataque puede llevar a la divulgación de datos confidenciales, denegación de servicio, falsificación de solicitudes del lado del servidor, escaneo de puertos desde la perspectiva de la máquina donde se encuentra el analizador y otros impactos del sistema.

El estándar XML 1.0 define la estructura de un documento XML, también define un concepto llamado entidad, que es una unidad de almacenamiento de algún tipo.

A4
:2017

Entidades Externas XML (XXE)

Hay algunos tipos diferentes de entidades, la entidad externa general / parámetro analizado a menudo se acorta a la entidad externa, que puede acceder a contenido local o remoto a través de un identificador de sistema declarado.

Se supone que el identificador del sistema es un URI que el procesador XML puede anular (acceder) al procesar la entidad, posteriormente el procesador XML reemplaza las ocurrencias de la entidad externa nombrada con los contenidos que el identificador del sistema hace referencia.

A4
:2017

Entidades Externas XML (XXE)

Si el identificador del sistema contiene datos contaminados y el procesador XML no hace referencia a estos datos contaminados, el procesador XML puede revelar información confidencial que normalmente no es accesible por la aplicación.

Los ataques pueden incluir la divulgación de archivos locales, que pueden contener datos confidenciales, como contraseñas o datos privados del usuario, mediante el uso de archivos: esquemas o rutas relativas en el identificador del sistema.

A4
:2017

Entidades Externas XML (XXE)

Dado que el ataque ocurre en relación con la aplicación que procesa el documento XML, el atacante puede usar esta aplicación confiable para pivotar a otros sistemas internos, posiblemente divulgando otro contenido interno a través de solicitudes http (s) o lanzando un CSRF (Cross-Site Request Forgery), crear un ataque a cualquier servicio interno desprotegido.

Se debe tomar en cuenta que la aplicación no necesita devolver explícitamente la respuesta al atacante para que sea vulnerable a las divulgaciones de información. Un atacante puede aprovechar la información del DNS para filtrar los datos a través de nombres de subdominios a un servidor DNS que controla.

Top-Ten Vulnerabilidades según OWASP

A4
:2017

Entidades Externas XML (XXE)

App. Específica	Explotabilidad: 2	Prevalencia: 2	Detectabilidad: 3	Técnico: 3	¿Negocio?
<p>Los atacantes pueden explotar procesadores XML vulnerables si cargan o incluyen contenido hostil en un documento XML, explotando código vulnerable, dependencias o integraciones.</p>		<p>De forma predeterminada, muchos procesadores XML antiguos permiten la especificación de una entidad externa, una URI que se referencia y evalúa durante el procesamiento XML. Las herramientas SAST pueden descubrir estos problemas inspeccionando las dependencias y la configuración. Las herramientas DAST requieren pasos manuales adicionales para detectar y explotar estos problemas. Los <i>testers</i> necesitan ser entrenados para hacer estas pruebas, ya que no eran realizadas antes de 2017.</p>		<p>Estos defectos se pueden utilizar para extraer datos, ejecutar una solicitud remota desde el servidor, escanear sistemas internos, realizar un ataque de denegación de servicio y ejecutar otro tipo de ataques.</p> <p>El impacto al negocio depende de las necesidades de la aplicación y de los datos.</p>	

A4
:2017

Entidades Externas XML (XXE)

Factores de riesgos

- La aplicación analiza documentos XML.
 - Los datos contaminados se permiten dentro de la parte del identificador del sistema de la entidad, dentro de la declaración de tipo de documento (DTD).
 - El procesador XML está configurado para validar y procesar la DTD.
 - El procesador XML está configurado para resolver entidades externas dentro de la DTD.
-

A4
:2017

Entidades Externas XML (XXE)

¿La aplicación es vulnerable?

Las aplicaciones y, en particular servicios web basados en XML, o integraciones que utilicen XML, pueden ser vulnerables a este ataque si:

- La aplicación acepta XML directamente, carga XML desde fuentes no confiables o inserta datos no confiables en documentos XML. Por último, estos datos son analizados sintácticamente por un procesador XML.

A4
:2017

Entidades Externas XML (XXE)

¿La aplicación es vulnerable?

- Cualquiera de los procesadores XML utilizados en la aplicación o los servicios web basados en SOAP, poseen habilitadas las definiciones de tipo de documento (DTDs). Dado que los mecanismos exactos para deshabilitar el procesamiento de DTDs varía para cada procesador, se recomienda consultar la hoja de trucos para prevención de XXE de OWASP.
- Ser vulnerable a ataques XXE significa que probablemente la aplicación también es vulnerable a ataques de denegación de servicio, incluyendo el ataque Billion Laughs.

A4
:2017

Entidades Externas XML (XXE)

¿La aplicación es vulnerable?

- La aplicación utiliza SAML para el procesamiento de identidades dentro de la seguridad federada o para propósitos de Single Sign-On (SSO). SAML utiliza XML para garantizar la identidad de los usuarios y puede ser vulnerable.
- La aplicación utiliza SOAP en una versión previa a la 1.2 y, si las entidades XML son pasadas a la infraestructura SOAP, probablemente sea susceptible a ataques XXE.

A4
:2017

Entidades Externas XML (XXE)

¿Cómo se previene?

El entrenamiento del desarrollador es esencial para identificar y mitigar defectos de XXE. Aparte de esto, prevenir XXE requiere:

- De ser posible, utilice formatos de datos menos complejos como JSON y evite la serialización de datos confidenciales.
- Actualice los procesadores y bibliotecas XML que utilice la aplicación o el sistema subyacente. Utilice validadores de dependencias. Actualice SOAP a la versión 1.2 o superior.

A4
:2017

Entidades Externas XML (XXE)

¿Cómo se previene?

- Deshabilite las entidades externas de XML y procesamiento DTD en todos los analizadores sintácticos XML en su aplicación, según se indica en la hoja de trucos para prevención de XXE de OWASP.
- Implemente validación de entrada positiva en el servidor (“lista blanca”), filtrado y sanitización para prevenir el ingreso de datos dañinos dentro de documentos, cabeceras y nodos XML.

A4
:2017

Entidades Externas XML (XXE)

¿Cómo se previene?

- Verifique que la funcionalidad de carga de archivos XML o XSL valide el XML entrante, usando validación XSD o similar.
- Las herramientas SAST pueden ayudar a detectar XXE en el código fuente, aunque la revisión manual de código es la mejor alternativa en aplicaciones grandes y complejas.
- Si estos controles no son posibles, considere usar parcheo virtual, gateways de seguridad de API, o Firewalls de Aplicaciones Web (WAFs) para detectar, monitorear y bloquear ataques XXE.

A4
:2017

Entidades Externas XML (XXE)

Ejemplos de escenarios de ataques

Han sido publicados numerosos XXE, incluyendo ataques a dispositivos embebidos. Los XXE ocurren en una gran cantidad de lugares inesperados, incluyendo dependencias profundamente anidadas. La manera más fácil es cargar un archivo XML malicioso, si es aceptado.

Escenario 1: el atacante intenta extraer datos del servidor:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE foo [
<!ELEMENT foo ANY>
<!ENTITY xxe SYSTEM "file:///etc/passwd">]>
<foo>&xxe;</foo>
```


A4
:2017

Entidades Externas XML (XXE)

Ejemplos de escenarios de ataques

Escenario 2: cambiando la línea ENTITY anterior, un atacante puede escanear la red privada del servidor:

```
<!ENTITY xxe SYSTEM "https://192.168.1.1/private">]>
```

Escenario #3: incluyendo un archivo potencialmente infinito, se intenta un ataque de denegación de servicio:

```
<!ENTITY xxe SYSTEM "file:///dev/random">]>
```

A5

Configuración de Seguridad Incorrecta

Recordemos que está ubicada en la tabla de amenazas de 2013, para 2017 esta amenaza pasa a ser la amenaza 6

A6
:2017

Configuración de Seguridad Incorrecta

12

Top-Ten Vulnerabilidades según OWASP



A5

Configuración de Seguridad Incorrecta

Una buena seguridad requiere tener definida e implementada una configuración segura para la aplicación, marcos de trabajo, servidor de aplicación, servidor web, base de datos y plataforma. Todas estas configuraciones deben ser definidas, implementadas, y mantenidas ya que por lo general no son seguras por defecto.

Esto incluye mantener todo el software actualizado, incluidas las librerías de código utilizadas por la aplicación.

Top-Ten Vulnerabilidades según OWASP



A5

Configuración de Seguridad Incorrecta


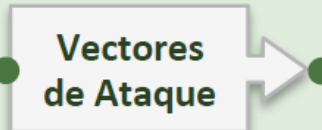

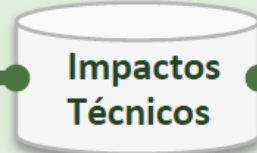

La configuración de seguridad incorrecta es un problema muy común y se debe en parte a establecer la configuración de forma manual, ad hoc o por omisión (o directamente por la falta de configuración).

Son ejemplos: S3 buckets abiertos, cabeceras HTTP mal configuradas, mensajes de error con contenido sensible, falta de parches y actualizaciones, frameworks, dependencias y componentes desactualizados, etc.

Top-Ten Vulnerabilidades según OWASP

A5

**Configuración de Seguridad
Incorrecta**

 Agentes de Amenaza	 Vectores de Ataque	 Debilidades de Seguridad		 Impactos Técnicos	 Impactos al negocio
Específico de la Aplicación	Explotabilidad FÁCIL	Prevalencia COMÚN	Detección FÁCIL	Impacto MODERADO	Específico de la aplicación / negocio
<p>Considere atacantes anónimos externos así como usuarios con sus propias cuentas que pueden intentar comprometer el sistema. También considere personal interno buscando enmascarar sus acciones.</p>	<p>Un atacante accede a cuentas por defecto, páginas sin uso, fallas sin parchear, archivos y directorios sin protección, etc. para obtener acceso no autorizado o conocimiento del sistema.</p>	<p>Las configuraciones de seguridad incorrectas pueden ocurrir a cualquier nivel de la aplicación, incluyendo la plataforma, servidor web, servidor de aplicación, base de datos, framework, y código personalizado. Los desarrolladores y administradores de sistema necesitan trabajar juntos para asegurar que las distintas capas están configuradas apropiadamente. Las herramientas de detección automatizadas son útiles para detectar parches omitidos, fallos de configuración, uso de cuentas por defecto, servicios innecesarios, etc.</p>		<p>Estas vulnerabilidades frecuentemente dan a los atacantes acceso no autorizado a algunas funcionalidades o datos del sistema. Ocasionalmente provocan que el sistema se comprometa totalmente.</p>	<p>El sistema podría ser completamente comprometido sin su conocimiento. Todos sus datos podrían ser robados o modificados lentamente en el tiempo.</p> <p>Los costes de recuperación podrían ser altos.</p>

Top-Ten Vulnerabilidades según OWASP

A5

Configuración de Seguridad
Incorrecta

A6
:2017

Configuración de Seguridad Incorrecta ¹²

App. Específica	Explotabilidad: 3	Prevalencia: 3	Detectabilidad: 3	Técnico: 2	¿Negocio?
<p>Los atacantes a menudo intentarán explotar vulnerabilidades sin parchear o acceder a cuentas por defecto, páginas no utilizadas, archivos y directorios desprotegidos, etc. para obtener acceso o conocimiento del sistema o del negocio.</p>		<p>Configuraciones incorrectas de seguridad pueden ocurrir en cualquier nivel del <i>stack</i> tecnológico, incluidos los servicios de red, la plataforma, el servidor web, el servidor de aplicaciones, la base de datos, <i>frameworks</i>, el código personalizado y máquinas virtuales preinstaladas, contenedores, etc. Los escáneres automatizados son útiles para detectar configuraciones erróneas, el uso de cuentas o configuraciones predeterminadas, servicios innecesarios, opciones heredadas, etc.</p>		<p>Los defectos frecuentemente dan a los atacantes acceso no autorizado a algunos datos o funciones del sistema. Ocasionalmente, estos errores resultan en un completo compromiso del sistema.</p> <p>El impacto al negocio depende de las necesidades de la aplicación y de los datos.</p>	

A5

**Configuración de Seguridad
Incorrecta**

¿Soy vulnerable?

¿Ha fortalecido la seguridad en todos los niveles de la pila de la aplicación?

1. ¿Tiene implementados procesos para mantener actualizado el software de su organización? Incluyendo el sistema operativo, los servidores web/aplicación, sistemas DBMS, aplicaciones y todas las bibliotecas de código

A5

**Configuración de Seguridad
Incorrecta**

¿Soy vulnerable?

2. ¿Todo lo innecesario ha sido deshabilitado eliminado o desinstalado?
 3. ¿Ha cambiado o deshabilitado las contraseñas de las cuentas predeterminadas?
 4. ¿Ha configurado el sistema de gestión de errores para prevenir que se acceda de forma no autorizada a los mensajes de error?
 5. ¿Se han comprendido y configurado de forma adecuada las características de seguridad de las bibliotecas y ambientes de desarrollo?
-

A5

**Configuración de Seguridad
Incorrecta**

¿La aplicación es vulnerable? (Top 10 2017)

- Falta hardening adecuado en cualquier parte del stack tecnológico, o permisos mal configurados en los servicios de la nube.
 - Se encuentran instaladas o habilitadas características innecesarias (ej. puertos, servicios, páginas, cuentas o permisos).
 - Las cuentas predeterminadas y sus contraseñas siguen activas y sin cambios.
 - El manejo de errores revela a los usuarios trazas de la aplicación u otros mensajes demasiado informativos.
-

A5

**Configuración de Seguridad
Incorrecta**

¿La aplicación es vulnerable? (Top 10 2017)

- Para los sistemas actualizados, las nuevas funciones de seguridad se encuentran desactivadas o no se encuentran configuradas de forma adecuada o segura.
 - Las configuraciones de seguridad en el servidor de aplicaciones, en el framework de aplicación (ej., Struts, Spring, ASP.NET), bibliotecas o bases de datos no se encuentran especificados con valores seguros.
 - El servidor no envía directrices o cabeceras de seguridad a los clientes o se encuentran configurados con valores inseguros.
-

A5

**Configuración de Seguridad
Incorrecta**

¿La aplicación es vulnerable? (Top 10 2017)

- El software se encuentra desactualizado o posee vulnerabilidades.

Sin un proceso de configuración de seguridad de aplicación concertado y repetible, los sistemas corren un mayor riesgo.

A5

**Configuración de Seguridad
Incorrecta**

¿Cómo se puede evitar?

Las principales recomendaciones se enfocan en establecer lo siguiente:

Cuatro alternativas

1. Un proceso repetible que permita configurar rápida y fácilmente entornos asegurados Los entornos de desarrollo, pruebas y producción deben estar configurados de la misma forma Este proceso debe ser automatizado para minimizar el esfuerzo requerido en la configuración de un nuevo entorno
-

A5

**Configuración de Seguridad
Incorrecta**

¿Cómo se puede evitar?

2. Un proceso para mantener y desplegar todas actualizaciones y parches de software de manera oportuna Este proceso debe seguirse en cada uno de los ambientes de trabajo Es necesario que se incluya las actualizaciones de todas las bibliotecas de código.
3. Una arquitectura robusta de la aplicación que provea una buena separación y seguridad entre los componentes.
4. Considerar la realización periódica de exploraciones y auditorías para ayudar a detectar fallos en la configuración o parches faltantes.

A5

**Configuración de Seguridad
Incorrecta**

¿Cómo se previene? (Top 2017)

Deben implementarse procesos seguros de instalación, incluyendo:

- Proceso de fortalecimiento reproducible que agilice y facilite la implementación de otro entorno asegurado. Los entornos de desarrollo, de control de calidad (QA) y de Producción deben configurarse de manera idéntica y con diferentes credenciales para cada entorno. Este proceso puede automatizarse para minimizar el esfuerzo requerido para configurar cada nuevo entorno seguro.

A5

**Configuración de Seguridad
Incorrecta**

¿Cómo se previene? (Top 10 2017)

- Use una plataforma minimalista sin funcionalidades innecesarias, componentes, documentación o ejemplos. Elimine o no instale frameworks y funcionalidades no utilizadas.
 - Siga un proceso para revisar y actualizar las configuraciones apropiadas de acuerdo a las advertencias de seguridad y siga un proceso de gestión de parches. En particular, revise los permisos de almacenamiento en la nube (por ejemplo, los permisos de buckets S3).
-

A5

**Configuración de Seguridad
Incorrecta**

¿Cómo se previene? (Top 10 2017)

- La aplicación debe tener una arquitectura segmentada que proporcione una separación efectiva y segura entre componentes y acceso a terceros, contenedores o grupos de seguridad en la nube (ACLs).
 - Envíe directivas de seguridad a los clientes (por ej. Cabeceras de seguridad).
 - Utilice un proceso automatizado para verificar la efectividad de los ajustes y configuraciones en todos los ambientes.
-

A5

**Configuración de Seguridad
Incorrecta**

Ejemplos de escenarios de ataques

Escenario 1:

La consola de administración del servidor de aplicaciones está instalada y no ha sido removida.

Las cuentas predeterminadas no han sido cambiadas.

Un atacante descubre las páginas por defecto de administración que están en su servidor, se conecta con las contraseñas por defecto y lo toma

A5

**Configuración de Seguridad
Incorrecta**

Ejemplos de escenarios de ataques

Escenario 2:

El listado del contenido de los directorios no está deshabilitado en el servidor. Los atacantes descubren que pueden encontrar cualquier archivo simplemente consultando el listado de los directorios.

Los atacantes encuentran y descargan las clases java compiladas, estas son desensambladas por ingeniería reversa para obtener su código, a partir de un análisis del código se pueden detectar defectos en el control de acceso de la aplicación.

A5

**Configuración de Seguridad
Incorrecta**

Ejemplos de escenarios de ataques

Escenario 3:

La configuración del servidor de aplicaciones permite que los mensajes de la pila sean retornados a los usuarios.

Eso potencialmente expone defectos en la aplicación.

A los atacantes les encanta que les proporcionen información extra con los mensajes de errores.

A5

**Configuración de Seguridad
Incorrecta**

Ejemplos de escenarios de ataques

Escenario 4:

El servidor de aplicaciones viene con aplicaciones de ejemplo que no se eliminaron del servidor de producción.

Las aplicaciones de ejemplo pueden poseer fallos de seguridad bien conocidos que los atacantes pueden utilizar para comprometer el servidor.

A5

**Configuración de Seguridad
Incorrecta**

Ejemplos de escenarios de ataques (Top 10 2017)

Escenario 1:

Un proveedor de servicios en la nube (CSP) por defecto permite a otros usuarios del CSP acceder a sus archivos desde Internet.

Esto permite el acceso a datos sensibles almacenados en la nube.

Referencias

<https://www.owasp.org>