ISO 27001 Informe de Gestión de Incidentes de Cumplimiento - Vulnerabilidad de Inyección SQL

Introduccion del reporte

Este informe detalla la identificación y explotación de una vulnerabilidad de inyección SQL en la aplicación web Damn Vulnerable (DVWA). La prueba se realizó en un entorno controlado, se llevó a cabo en una máquina de virtualbox con sistema operativo DEBIAN LINUX, para demostrar una vulnerabilidad común en base de datos SQL con nivel de seguridad bajo y su posible impacto en la seguridad de la aplicación.

Descripción del incidente

Durante la evaluación de seguridad de DVWA, se descubrió una vulnerabilidad de inyección SQL en el módulo "Inyección SQL". Esta vulnerabilidad permite a un atacante inyectar consultas SQL maliciosas a través de los campos de entrada de la aplicación web, comprometiendo así la integridad y confidencialidad de los datos almacenados en la base de datos y dándonos así toda la información que contenía la base de datos anteriormente dicha.

Método de inyección SQL utilizado

Para replicar y demostrar la vulnerabilidad, se utilizó la siguiente carga útil SQL en el campo "ID de usuario":

SQL

```
1' OR '1'='1
```

Esta carga útil explota la vulnerabilidad para modificar la consulta SQL original de tal manera que devuelva los nombres de usuario y las contraseñas almacenadas en la tabla de usuarios, Al introducir esta cadena, el atacante explota la vulnerabilidad de la aplicación para modificar la consulta SQL original. Normalmente, la consulta estaría diseñada para buscar datos de un usuario concreto utilizando un filtro. Aquí, la condición '1'='1' es siempre verdadera, lo que hace que la consulta devuelva todos los registros de la tabla de usuarios en lugar de limitarse a uno en particular. En la imagen se aprecia que, tras enviar el payload, la aplicación despliega una lista de usuarios (por ejemplo, admin, Gordon Brown, Hack Me, Pablo Picasso y Bob Smith), evidenciando que los datos de todos ellos han sido recuperados sin ningún tipo de autorización.

Impacto del incidente

Explotar esta vulnerabilidad podría permitir a un atacante:

● Acceder y extraer información confidencial de la base de datos, incluidas las credenciales de usuario.

● Modificar, eliminar o comprometer datos confidenciales almacenados en la aplicación.

Esto representa un riesgo significativo para la confidencialidad, integridad y disponibilidad de los datos y servicios proporcionados por DVWA, lo que representa el bajo nivel de seguridad que esta implementando este servidor.

Recomendaciones

Con base en los hallazgos de esta evaluación de seguridad, se recomiendan las siguientes medidas correctivas y preventivas:

1. Validación de entrada: Implementar validaciones de entrada estrictas para todos los datos proporcionados por el usuario, utilizando parámetros seguros en las consultas SQL para evitar la inyección de SQL.

2. Pruebas de penetración: Realizar auditorías de seguridad periódicas, incluyendo pruebas de penetración, para identificar y mitigar las vulnerabilidades de seguridad antes de que sean explotadas por atacantes.

3 - Usar consultas parametrizadas: En lugar de construir consultas SQL concatenando entradas de usuario, utiliza consultas preparadas o declaraciones parametrizadas. Esto evita que los datos ingresados sean interpretados como código SQL malicioso.

4 - Aplicar el principio de mínimo privilegio: Limita los permisos de las cuentas de la base de datos. Por ejemplo, una cuenta utilizada por la aplicación web no debería tener permisos para modificar la estructura de la base de datos.
5 - Usar un firewall de aplicaciones web (WAF): Un WAF puede detectar y bloquear intentos de inyección SQL antes de que lleguen a la base de datos.

Conclusiones

La identificación y explotación exitosa de la vulnerabilidad de inyección de SQL en DVWA subraya la importancia de la seguridad proactiva en el desarrollo y mantenimiento de aplicaciones web. Implementar controles de seguridad robustos y seguir las mejores prácticas de ciberseguridad es esencial para proteger los activos críticos y garantizar la continuidad del negocio, como así también evitar roba de identidad de los usuarios del servicio y mantener una imagen limpia de la organización.