RIESGOS Y VULNERABILIDADES EN LAS BASES DE DATOS RELACIONALES

CHRISTIAN ALEXANDER PINILLA RAMOS

UNIVERSIDAD NACIONAL ABIERTA Y A DISTANCIA – UNAD
ESCUELA DE CIENCIAS BÁSICAS, TECNOLOGÍA E INGENIERÍA - ECBTI
ESPECIALIZACIÓN EN SEGURIDAD INFORMÁTICA
BOGOTÁ D.C
2022

RIESGOS Y VULNERABILIDADES EN LOS SISTEMAS INFORMÁTICOS

CHRISTIAN ALEXANDER PINILLA RAMOS

Monografía para optar por el título de
ESPECIALISTA EN SEGURIDAD INFORMÁTICA

DANIEL FELIPE PALOMO
Director de grado

UNIVERSIDAD NACIONAL ABIERTA Y A DISTANCIA – UNAD
ESCUELA DE CIENCIAS BÁSICAS, TECNOLOGÍA E INGENIERÍA - ECBTI
ESPECIALIZACIÓN EN SEGURIDAD INFORMÁTICA
BOGOTA DC
2022

NOTA DE ACEPTACIÓN

Firma del Presidente de Jurado

Firma del Jurado

Firma del Jurado

Bogotá D.C., Fecha sustentación

DEDICATORIA

Con amor dedico este trabajo a mi esposa, que con su apoyo y comprensión me acompañó en el proceso familiar, colaborándome de manera indirecta, minimizando mis preocupaciones de padre, también lo dedico a mi mamá que con su apoyo, dedicación y sacrificio minimizo mis preocupaciones en el hogar.

AGRADECIMIENTOS

Agradezco a las directivas de la Universidad Nacional Abierta y a Distancia UNAD, quienes con su trabajo continúo nos brindan la oportunidad de estudiar, por otro lado, a cada uno de los tutores y asesores que me acompañaron en el proceso les reconozco que sin su apoyo, conocimiento y colaboración a este anhelado logro no hubiese sido posible de alcanzarlo.

CONTENIDO

LISTA DE FIGURAS

pág.

GLOSARIO

ACTIVO: cosa que tiene algún valor para un sujeto, empresa, entidad u compañía.

ACTIVO VIRTUAL: denominación de un activo en el espacio virtual.

AMENAZA: Fuente latente de un daño o pérdida de un activo de informático.

ATAQUE: intención de dañar, extraer, modificar, destruir, substraer u obtener acceso no autorizado en relación a un activo informático.

BASE DE DATOS: recopilación constituida de información organizada, que tiene como finalidad almacenar datos.

BASE DE DATOS RELACIONAL: es una recopilación de datos relacionados entre sí, donde se utiliza principalmente las tablas para el ingreso de la información.

BASE DE DATOS NO RELACIONAL: sistema de almacenamiento de información, el cual tiene como principal característica el no uso de lenguaje SQL.

BLUE TEAM: grupo especialista que evalúa la capacidad real de seguridad informática que tiene una empresa, con el fin de proteger sus activos informáticos.

CAUSA: motivo por la cual la eventualidad se materializa.

CIBERCRIMEN: proceso criminal, el cual encierra a los servicios informáticos o software que se contemplen objetivo en el ciberespacio.

CIBERESPACIO: entorno que agrupa a las personas, software y servicios en internet a través de equipos informáticos y redes de tecnología acopladas a este.

CIBERSEGURIDAD: su objetivo es minimizar problemas en el entorno físico, social financiero, digital o de otro tipo que se reflejan del daño, error, incidente, problema o cualquier otro evento que se considere no apropiado en el ciberespacio.

CIBERPROTECCIÓN: acción de salvaguardia del ciberespacio con la misión de proteger la integridad, confidencialidad y disponibilidad de la información en el ciberespacio.

CONTROLES: mecanismos implementados para examinar procedimientos que son identificados como dudosos, los cuales pueden afectar de algún modo los activos informáticos.

DISPONIBILIDAD: es la razón de que la información se mantenga asequible y útil, por criterio de la entidad que la ostente.

HOST: computadora o dispositivo conectado a la red, con una dirección IP y dentro de un dominio.

INTEGRIDAD: responde la disponibilidad y la autenticidad de los datos, salvaguardando la información de acciones no autorizadas con respecto a su modificación, eliminación, destrucción, resaltando que estos procedimientos en muchas ocasiones no se encuentran aprobados.

RED TEAM: equipo de atacantes informáticos, que pone a prueba a una compañía con el fin de optimizar su seguridad informática.

RIESGO: mezcla de probabilidades de que suceda algún suceso y las consecuencias de estas sean dañinas.

VULNERABILIDAD: básicamente es un fallo o debilidad dentro de un sistema informático, el cual puede ser explotado por algún tipo de atacante.

RESUMEN

El presente documento relacionado con la detección de vulnerabilidades pretende dar a conocer una información sobre los procedimientos y herramientas utilizadas para la identificación de vulnerabilidades en las bases de datos; teniendo en cuenta, factores cualitativos y cuantitativos sobre resultados de investigaciones anteriores, relacionadas con la gestión del riesgo informático. Se tomará como referencia procedimientos y técnicas de detección de vulnerabilidades. La finalidad principal de la presente monografía estará asociada a los siguientes pasos alineados a los objetivos específicos: Exploración de los riesgos y amenazas que estén asociados a las vulnerabilidades en las bases de datos. Identificación de herramientas que proporcionen el análisis de vulnerabilidades. Verificación del funcionamiento de herramientas utilizables para la detección de vulnerabilidades en las bases de datos relacionales y por último sugerir alternativas de solución para las vulnerabilidades encontradas en las bases de datos.

Palabras claves: Riesgo, vulnerabilidad, identificación, bases de datos, relacionales.

ABSTRACT

This document related to the detection of vulnerabilities, aims to present an information on the procedures and tools used to identify vulnerabilities in databases; taking into account qualitative and quantitative factors on the results of previous research, related to IT risk management. Procedures and techniques for detecting vulnerabilities will be taken as a reference. The main purpose of this monograph will be associated with the following steps aligned to the specific objectives: Investigation of risks and threats that are associated with vulnerabilities in databases. Identification of tools that provide vulnerability analysis. Verification of the operation of usable tools for the detection of vulnerabilities in relational databases and, finally suggest alternative solutions for the vulnerabilities found in the databases.

Keywords: Risk, vulnerability, identification, databases, relational.

INTRODUCCIÓN

En su gran mayoría las organizaciones públicas o privadas, al igual que las personas, dependen de alguna manera de la tecnología de la información, como una herramienta esencial para lograr sus objetivos de negocio o para poder desarrollar actividades en su vida cotidiana; al mismo tiempo, todos tienen que enfrentarse con una amplia gama de amenazas y vulnerabilidades asociadas a los entornos informáticos de hoy. En la actualidad la seguridad de la información es más que, un problema de seguridad de datos en los computadores; esta básicamente orientada a proteger la propiedad intelectual y la información importante de las organizaciones y de las personas.

Dentro de las organizaciones las bases de datos son los activos más sensibles, en relación con el tráfico, pérdida y/o daño de información; se deben seguir el correcto uso de los principios básicos de la seguridad de la información, es decir, mantener la confidencialidad, integridad y disponibilidad de los datos. Teniendo en cuenta la sensibilidad de este tipo de activos de información, es necesario adquirir conocimientos con respecto a la detección temprana de vulnerabilidades. Ya que, de esta forma, se podría mitigar la explotación de este tipo de riesgos, más específicamente los relacionadas con los motores de bases de datos *Mysql*.

Todo esto con la finalidad de proteger el activo más valioso de una empresa, la información. Lo que se busca en este documento es analizar los riesgos y amenazas que estén asociados a las vulnerabilidades en las bases de datos relacionales. Así como, establecer herramientas que proporcionen la detección de vulnerabilidades y faciliten la identificación de riesgos; al igual que, probar su funcionamiento y finalmente recomendar alternativas de solución a estos problemas de seguridad informática.

1. DEFINICIÓN DEL PROBLEMA

El problema surge por la necesidad generada en la dificultad de la materialización de vulnerabilidades, directamente asociadas a las bases de datos relacionales; por lo general cuando se produce la explotación de estos riesgos; estos pueden ocasionar la pérdida de información clasificada y daños en los sistemas informáticos. Y de esta forma, generar pérdidas económicas considerables dentro de las empresas.

1.1 ANTECEDENTES DEL PROBLEMA

La tecnología ha facilitado algunos procesos cotidianos del hombre a través de las telecomunicaciones, uno de estos procesos es él de la realización de transacciones bancarias; otro de ellos es el acceso remoto a *host* empresariales mediante diversos dispositivos móviles. También es cierto que esto, expone a los usuarios a una probabilidad cada vez más alta a ataques informáticos, comúnmente conocidos como 'ciberataques' y que pueden tener consecuencias para el equipo o la información que él mismo aloja.

En el año 2020, en cada segundo ocurrieron alrededor de 12 ataques informáticos en América Latina, según cifras estimadas en la Cumbre de Analistas de Seguridad de *Kaspersky Lab*; por lo que, "en total, se contabilizaron un millón de violaciones a la seguridad de los internautas en esta región. Más allá de las voluminosas inversiones que se realizan para combatir estos ataques informáticos, buscando salvaguardar la infraestructura pública y privada, se hace necesario también un cambio cultural en el manejo y el almacenamiento de la información que realizan los usuarios mismos, pues eso da lugar a gran cantidad de amenazas vía e-mail o a la extracción de información confidencial mediante engaños como obtención de premios, recompensas y similares" [1].

De acuerdo con las cifras de ataques informáticos conferidas por la cumbre de analistas de *Kaspersky Lab*, se puede señalar que las vulnerabilidades en los sistemas informáticos en muchos de los casos son explotadas por los ciberdelincuentes y por lo general ayudan a lograr fines criminales [2]. Ignorar este tipo de problemáticas en el campo de la seguridad, y principalmente en las bases de datos, ha generado cuantiosas pérdidas económicas dentro de las empresas teniendo en cuenta que los atacantes al

[1] AVILA FORERO, Raúl. Amenazas cibernéticas y la vulnerabilidad de nuestro negocio. [En línea]. Bogotá: Semana, 2016., Disponible en: https://www.semana.com/opinion/columnistas/articulo/amenazas-ciberneticas-y-la-vulnerabilidad-de-nuestro-negocio-por-raul-avila/231682/.

[2] SALDANA, Gustavo. Kaspersky Lab registra un alza de 60% en ataques cibernéticos en América Latina. [En línea]. Ciudad de México: ciberamenazas, 2018., Disponible en: https://latam.kaspersky.com/blog/kaspersky-lab-registra-un-alza-de-60-en-ataques-ciberneticos-en-america-latina/13266/.

explotar estas vulnerabilidades posiblemente tendrán algún objetivo estratégico o económico.

> "Un cibercriminal se infiltró en 22.900 bases de datos poco seguras, eliminó su contenido y dejó una nota de rescate indicando que exige un pago en *bitcoins* a cambio de los datos. Si el rescate no se paga dentro de dos días, el atacante amenaza con notificar a las autoridades a cargo de hacer cumplir el Reglamento General de Protección de Datos (GDPR, por sus siglas en inglés) de la Unión Europea. Según *ZDNet*, medio que dio a conocer la historia, el cibercriminal detrás de esta operación está utilizando *scripts* automáticos para buscar en Internet las instalaciones de bases de datos expuestas a Internet sin protección por contraseña, eliminando su contenido y solicitando el pago de 0.015 en *bitcoins* (equivalente unos US$140) para devolver los datos" [3].

De esta manera, evitar los ataques informáticos a las bases de datos, mediante medidas de prevención y relacionados minimiza el riesgo de que se puedan producir perdidas de información, daño o alteración de las mismas, y además puede evitar detrimentos financieros considerables dentro de las organizaciones que surgen como consecuencia de las posibles insuficiencias en seguridad informática.

1.2 FORMULACIÓN DEL PROBLEMA

¿Cómo se podrán minimizar riesgos y vulnerabilidades en las bases de datos relacionales?

[3] PECERT. Alerta integrada de seguridad digital: *CreUsuSinAut.* Julio, 2020. Vol. 90, no. 090-2020, P. 5.

2. JUSTIFICACIÓN

La presente monografía se realiza por la necesidad de documentar sobre la afectación a las bases de datos en especial a las relacionales. Se busca identificar diferentes procedimientos y herramientas que faciliten el entendimiento de detección de vulnerabilidades en este tipo de base de datos. Es decir, que se pretende proponer mecanismos que faciliten la gestión del riesgo, buscando proteger la disponibilidad, integridad y confidencialidad de la información.

La protección de los sistemas de información dentro de una empresa es extremadamente importante y por lo tanto se debe garantizar la protección del activo más valioso (información). Por lo tanto, se deben establecer políticas de seguridad informática y aprender a utilizar herramientas técnicas que faciliten el monitoreo continuo del entorno de las Tecnologías de la Información TI, evitando la explotación de vulnerabilidades por parte de los atacantes.

Desde otra perspectiva y a través de la exploración de masas documentales, esta monografía ayudará adquirir habilidades y destrezas de manera práctica, buscando dar solución al desconocimiento de procedimientos para la ejecución de herramientas de detección de vulnerabilidades y así mismo, sugerir alternativas de solución, para evitar su materialización. Todo esto, con la finalidad de proteger la información digital de las organizaciones y de esta forma, evitar problemas de seguridad de la información.

3. OBJETIVOS

3.1 OBJETIVO GENERAL

Analizar los procedimientos aplicables para la detección de riesgos y vulnerabilidades dentro de las bases de datos relacionales, para plantear alternativas de solución.

3.2 OBJETIVOS ESPECÍFICOS

- Explorar los riesgos y amenazas que estén asociados a las vulnerabilidades en las bases de datos relacionales.
- Establecer herramientas que proporcionen el análisis de vulnerabilidades y faciliten la identificación de riesgos en las bases de datos relacionales.
- Probar el funcionamiento de herramientas utilizadas para la detección de vulnerabilidades en las bases de datos relacionales.
- Recomendar alternativas de solución para mitigar vulnerabilidades en las bases de datos relacionales.

4. MARCO REFERENCIAL

4.1. MARCO TEÓRICO

4.1.1 Bases de datos. De acuerdo con *Oracle* Colombia [4], una base de datos es una colección organizada de información estructurada, o datos, típicamente almacenados electrónicamente en un sistema de computadora. Una base de datos es comúnmente inspeccionada por un sistema de gestión de base de datos (DBMS). En conjunto, los datos y el DBMS, y las aplicaciones que están asociados con ellos, se conocen como un sistema de base de datos, que a menudo reduce el término a solo base de datos.

Según lo referenciado por *Oracle*, los datos presentados intrínsecamente dentro de los tipos más comunes de bases de datos en funcionamiento hoy en día se modelan típicamente en filas y columnas en una serie de tablas para que el tratamiento y la consulta de datos sean eficientes. Luego, se puede administrar fácilmente los datos. La mayoría de las bases de datos utilizan lenguaje de consulta estructurado *(SQL*) para escribir y consultar datos.

Además, la compañía *Oracle*, manifiesto que *SQL* es un lenguaje de programación usado por casi todas las bases de datos relacionales para administrar datos, y para proporcionar control de acceso. *SQL* se desarrolló por primera vez en la compañía *International Business Machines Corporation* IBM en la década del 70´ con Oracle como uno de los principales colaboradores, lo que llevó a la implementación del estándar del *American National Standards Institute ANSI de SQL.* Así, *SQL* ha generado muchas extensiones por parte de compañías como *IBM, Oracle y Microsoft.* Aunque *SQL* todavía se usa ampliamente en la actualidad, por la confianza que genera en sus usuarios y su presencia tradicional, comienzan a aparecer nuevos lenguajes de programación.

> Oracle sostiene que “las bases de datos han evolucionado dramáticamente desde su inicio a principios de los años sesenta. Las bases de datos de navegación, como la base de datos jerárquica (que se basaba en un modelo similar a un árbol y solo permitía una relación de uno a muchos), y la base de datos de red (un modelo más flexible que permitía múltiples relaciones), eran los sistemas originales utilizados para almacenar y manipular los datos. Aunque simples, estos primeros sistemas eran inflexibles. En la década de 1980, las bases de datos relacionales se hicieron populares, seguido de bases de datos orientadas a objetos en los años noventa. Más recientemente, surgieron las bases de datos NoSQL como respuesta al crecimiento de internet y la necesidad de una mayor velocidad y procesamiento de

[4] ORACLE Colombia. Bases de datos. [En línea]. Bogotá: ORACLE, 2019., p. 1. Disponible en: https://www.oracle.com/co/database/what-is-database/

datos no estructurados. Hoy, las bases de datos en la nube y las bases de datos independientes están abriendo nuevos caminos en cuanto a cómo se recopilan, almacenan, administran y utilizan los datos" [5].

4.1.2 Tipos de bases de datos. Según *Oracle*[6], hay muchos tipos de bases de datos. La mejor base de datos para una empresa específica depende de cómo la organización pretende utilizar los datos.

- **Bases de datos relacionales.** Las bases de datos relacionales se popularizaron en los años ochenta. Los elementos de una base de datos relacional se organizan como un conjunto de tablas con columnas y filas. La tecnología de base de datos relacional proporciona la manera más eficiente y flexible de acceder a información estructurada.
- **Bases de datos orientadas a objetos.** Este modelo agrupa paquetes relacionados entre sí, generalmente los datos de cada registro se combinan en un solo objeto, con todos sus atributos. De esta forma, toda la información se mantiene de forma agrupada y no en distintas tablas, facilitando la interacción de la información.
- **Bases de datos distribuidas.** Una base de datos distribuida consta de dos o más archivos ubicados en diferentes sitios. La base de datos puede almacenarse en múltiples computadoras, ubicadas en la misma ubicación física o dispersas en diferentes redes.
- **Almacenes de datos.** Un almacén de datos es un tipo de base de datos diseñada específicamente para consultas y análisis rápidos, y funciona como un depósito central de datos.
- **Bases de datos *NoSQL*.** Una *NoSQL*, o una base de datos no relacional, permite que los datos no estructurados y semiestructurados se almacenen y manipulen, a diferencia de una base de datos relacional, que define cómo deben componerse todos los datos insertados en la base de datos. Las bases de datos *NoSQL* se hicieron populares a medida que las aplicaciones web se hacían más comunes y más complejas.
- **Bases de datos orientadas a grafos.** Una base de datos orientada a grafos almacena datos en términos de entidades y las relaciones entre entidades.
- **Bases de datos *OLTP*.** Una base de datos *OLTP* es una base de datos analítica y rápida diseñada para un gran número de transacciones realizadas por múltiples usuarios.

[5] ORACLE Colombia. Bases de datos. [En línea]. Bogotá: ORACLE, 2019., p. 3. Disponible en: https://www.oracle.com/co/database/what-is-database/

[6] Ibid, p.5

4.1.3 Sistema de administración de base de datos. Una base de datos ordinariamente demanda un programa completo de software de base de datos, que se conoce como sistema de administración de bases de datos *(DBMS)*[7]. Un *DBMS* este funciona de manera práctica como una interfaz entre la base de datos y sus interesados o programas finales, lo que permite a los usuarios administrar cómo se organiza y optimiza la información.

Algunos ejemplos de software de bases de datos o DBMS populares incluyen MySQL, *Microsoft Access, Microsoft* SQL *Server, FileMaker Pro, Oracle Database* y dBASE.

4.1.4 **Base de datos *MySQL*.** Es un sistema de gestión de bases de datos relacionales de código abierto basado en el lenguaje *SQL*. Básicamente, creado y orientado a aplicaciones web y además de esto, puede ejecutarse en múltiples plataformas[8]. Bases de datos como *MySQL* han evolucionado a medida que lo ha requerido la Internet, dándole lugar como una de las más utilizadas y recomendada para las aplicaciones asentadas en la web, calificación otorgada por diseñadores web. Ya que, está diseñada para procesar millones de consultas y miles de transacciones, *MySQL* es una elección para las empresas de comercio electrónico que necesitan gestionar negocios económicos. La flexibilidad bajo demanda es la característica principal de *MySQL.*

De acuerdo con el análisis de distintos proyectos sobre seguridad en bases de datos, se toman como referencia los más influyentes en términos académicos para esta investigación, así:

- Carolina Bonilla, en el año 2018, para obtener el título de magíster en gestión de bases de datos, en su proyecto de investigación; presenta el siguiente informe de mitigación de vulnerabilidades, el cual se toma como referencia para identificar alternativas de solución a posibles amenazas de Ciberseguridad. Informe de mitigación que se visualiza en la Figura 1.

[7] RUIZ GARCÍA, ezquiel. Administración de bases de datos. [En línea]. Lima: Universidad peruana de los Andes, 2016., p. 10. Disponible en: https://profesorezequielruizgarcia.files.wordpress.com/2016/06/administracion-de-base-de-datos.pdf

[8] GÓMEZ RODRÍGUEZ, Juan Felipe. Implementación de aplicación web con acceso a base de datos para manejo de inventario de la empresa *Orange Business Services* Colombia S.A. Trabajo de grado para el título de Ingeniero Electrónico. Bogotá D.C. Facultad De Ingeniería Electrónica. 2017.

Figura 1. Informe de mitigación de vulnerabilidades

INFORME DE MITIGACIÓN DE VULNERABILIDADES DE BASE DE DATOS	
DATOS TÉCNICOS	**Host y puerto afectado:** 192.168.1.4:1521 Ejecutar Oracle TNS Listener de la Base de datos Oracle 10.2.0.4 (versión vulnerable)
TIPO DE IMPACTO	**CRÍTICO:** Las versiones obsoletas de Oracle pueden ser vulnerables.
DESCRIPCIÓN	La Base de Datos de Oracle podría ser vulnerables a los ataques de desbordamiento de búfer, inyección SQL, ataques TNS Listener, ataques de cross-site scripting, y los ataques de recorrido de directorio (directory traversal).
REFERENCIA	http://www.oracle.com/support/library/brochure/lifetime-support-technology.pdf#page=6 http://www.oracle.com/us/support/lifetime-support/index.html
MITIGACIÓN	Es muy recomendable que actualice un parche o una instalación de Base de Datos Oracle con las últimas actualizaciones disponibles.
DATOS TÉCNICOS	Vulnerabilidad fue encontrada en Oracle Database 10.2.0.3/10.2.0.4/10.2.0.5/11.1.0.7/11.2.0.1 Función **mdsys.reset_inprog_index() SQL INJECTION**
TIPO DE IMPACTO	**CRÍTICO**
DESCRIPCIÓN	La Base de Datos de Oracle podría ser vulnerables a los ataques de desbordamiento de búfer, inyección SQL, ataques TNS Listener, ataques de cross-site scripting, y los ataques de recorrido de directorio (directory traversal).
REFERENCIA	https://vuldb.com/es/?id.4247 SecurityFocus (BID 45855), X-Force (64760), Secunia (SA42895), SecurityTracker (ID 1024972) y Vulnerability Center (SBV-29225)
MITIGACIÓN	Una actualización elimina esta vulnerabilidad. Aplicando un parche es posible eliminar el problema. El parche puede ser descargado de oracle.com. El mejor modo sugerido para mitigar el problema es actualizar a la última versión.
DATOS TÉCNICOS	Oracle Database 10.2.0.3/10.2.0.4/10.2.0.5/11.1.0.7/11.2.0.2 **Enterprise Manager Base Platform vulnerabilidad desconocida.**
TIPO DE IMPACTO	**CRÍTICO:** Las versiones obsoletas de Oracle pueden ser vulnerables.
DESCRIPCIÓN	La vulnerabilidad es identificada como CVE-2012-0520. El ataque se puede hacer desde la red. La explotación no necesita ninguna autentificación específica. No se conoce los detalles técnicos ni hay ningún exploit disponible. Permite a atacantes remotos afectar la integridad a través de vectores desconocidos relacionados con el Framework.
REFERENCIA	https://vuldb.com/es/?id.5093 SecurityFocus (BID 53081), Secunia (SA48855) y SecurityTracker (ID 1026929).

MITIGACIÓN	Aplicando un parche es posible eliminar el problema. El parche puede ser descargado de oracle.com. Una solución posible ha sido publicada inmediatamente después de la publicación de la vulnerabilidad.
DATOS TÉCNICOS	Base de Datos Oracle 10.2.0.3/10.2.0.4/10.2.0.5/11.1.0.7 **Enterprise Manager Base Platform /em/console/logon/logon autenticación débil**
TIPO DE IMPACTO	**CRÍTICO**
DESCRIPCIÓN	Una función desconocida del archivo */em/console/logon/logon* del componente *Enterprise Manager Base Platform* es afectada por esta vulnerabilidad. Por la manipulación de un input desconocido se causa una vulnerabilidad de clase autenticación débil. Esto tiene repercusión sobre la confidencialidad e integridad. La vulnerabilidad es identificada como CVE-2012-0528. El ataque se puede hacer desde la red. La explotación no necesita ninguna autentificación específica. Los detalles técnicos así como un exploit privado son conocidos.
REFERENCIA	https://vuldb.com/es/?id.5105 Secunia (SA48855) y SecurityTracker (ID 1026929).
MITIGACIÓN	Aplicando un parche es posible eliminar el problema. El parche puede ser descargado de oracle.com. Una solución posible ha sido publicada inmediatamente después de la publicación de la vulnerabilidad.
DATOS TÉCNICOS	**Oracle Database 10.2.0.4/11.1.0.7 OCIPasswordChange API escalada de privilegios**
TIPO DE IMPACTO	**CRÍTICO**
DESCRIPCIÓN	Una función desconocida del componente *OCIPasswordChange API* es afectada por esta vulnerabilidad. Por la manipulación de un input desconocido se causa una vulnerabilidad de clase escalada de privilegios. Esto tiene repercusión sobre la confidencialidad e integridad. La vulnerabilidad es identificada como CVE-2012-0511. Se considera fácil de explotar. El ataque puede ser iniciado desde la red. La explotación no requiere ninguna forma de autentificación. No son conocidos los detalles técnicos, pero hay un exploit privado disponible.
REFERENCIA	https://vuldb.com/es/?id.5084 Secunia (SA48855) y SecurityTracker (ID 1026929).
MITIGACIÓN	Aplicando un parche es posible eliminar el problema. El parche puede ser descargado de oracle.com. Una solución posible ha sido publicada inmediatamente después de la publicación de la vulnerabilidad.

Fuente: BONILLA BACA, Carolina Anabel. Elaboración de una metodología de detección y mitigación de vulnerabilidades de base de datos y su incidencia en la seguridad de la información de la empresa Automekano cía. Ltda., de la ciudad de Ambato. Trabajo de Investigación, previo a la obtención del Grado Académico de Magíster en Gestión de Bases de datos. Ambato. Universidad técnica de Ambato. Facultad de ingeniería en sistemas, electrónica e industrial. 2017. Disponible en: https://repositorio.uta.edu.ec/bitstream/123456789/24534/1/Tesis_t1200mbd.pdf

- Así mismo, Rafael Molano en el año 2017, para obtener el título de especialista en auditoria de sistemas, en su proyecto de investigación; realizó entrevistas a cuatro líderes del área de las TI, sobre temas de seguridad en las bases de datos; obteniendo como resultado la siguiente información que se plasma en el siguiente mapa mental como lo muestra la Figura 2. Y de esta forma, se toma como referencia para documentar el desarrollo del documento.

Figura 2. Recomendaciones según metodología PHVA

Fuente: MOLANO ESPINEL, Rafael Antonio. Proyecto de investigación Estrategia para implementar un sistema de gestión de la Seguridad de la información basada en la norma ISO 27001 en el Área de ti para la empresa *Market MIX.* Trabajo de grado para obtener el título de Especialista en Auditoria de Sistemas. Bogotá D.C. Universidad Católica De Colombia. Facultad De Ingeniería. Programa De Especialización En Auditoria De Sistemas. 2017. Disponible en: https://repository.ucatolica.edu.co/bitstream/10983/15240/1/Esp%20Auditoria%20de%20sistemas.pdf

4.2 MARCO CONCEPTUAL

Un diagnóstico integral de tecnologías de la información y factores que afectan la ciberseguridad representa el punto de partida para diseñar herramientas que permitan administrar las necesidades detectadas en el mismo. De esta manera, haciendo una aproximación conceptual en la cual desde las herramientas técnicas de consulta se pueda abordar un aspecto que servirá como referencia al momento de realizar la investigación.

4.2.1 Herramientas de penetración y *testing.* son la forma más viable de medir la seguridad de sus sistemas de información, utilizando las mismas herramientas y/o procesos con los que administraría el sistema o un delincuente informático podría tener acceso a la organización u empresa, además de que si es un entorno controlado puede lograr la finalidad de identificar las fallas de seguridad.

4.2.1.1 *OWASP (Open Web Application Security Project* - Proyecto de seguridad de aplicaciones web abiertas). Según Pinzón[9] OWASP se constituye de 2 fases, la primera (pasiva) consiste en el levantamiento de datos, manejando distintas herramientas en busca de identificar de manera lógica el software; mientras que la segunda fase (activa) está basada en el uso de test de penetración en las que están comprendidas las siguientes categorías:

- Recopilación de datos.
- Verificación de la lógica del negocio.
- Pruebas de autenticación.
- Prueba de gestión de sesión.
- Prueba de validación de datos.
- Prueba de denegación de servicios.

[9] PINZÓN, Liliana. Pruebas de intrusión y metodologías abiertas. En: Revista Ciencia, Innovación y Tecnología (RCIYT). Enero-diciembre, 2013. Vol. 1. p. 27

4.2.1.2 *ISSAF (Information Systems Security Assessment Framework* - Marco de evaluación de seguridad de sistemas de información). De acuerdo con Pinzón, manifiesta que, “la metodología que consta de tres fases, en la primera se realiza la planificación, preparación y se establece el alcance y las pruebas que se deben realizar; la segunda fase consta de la evaluación, además se realizan pruebas de penetración a los activos de información; mientras que la tercer fase, se realizan los informes requeridos y la destrucción de la información que se levantó durante el proceso, en dicho informe se detallan los hallazgos y sus posibles soluciones” [10].

4.2.1.3 *OSSTMM 3* – Manual de metodología abierta para pruebas de seguridad. Esta metodología comprende sus objetivos en cuanto a 5 factores, a saber, físico, redes inalámbricas, humano, telecomunicaciones y redes de datos, además esta sub dividida por las siguientes 4 fases: Inducción (verificación de políticas, procedimientos y manuales, además de establecer el alcance de la prueba), Interacción (principalmente se basa en la verificación de los controles que se tienen establecidos), Investigación “Se verifican en detalle la calidad de los procesos y sus niveles de seguridad, además de la configuración de los equipos, capacitación, segregación de funciones, verificación de exposición del riesgo e Intervención (se realiza un mapeo y el impacto de los hallazgos), se realiza una revisión de la continuidad de las operaciones y sus diferentes alertas” [11].

4.3 ANTECEDENTES O ESTADO ACTUAL

4.3.1 Ataques informáticos realizados a bases de datos relacionales en la última década. La última década en la historia del hombre ha sido testigo del cambio de modelo en el que los hackers buscan aprovechar debilidades dentro de las organizaciones u empresas y las infraestructuras tecnológicas de las mismas. Con el fin de contrarrestar, es significativo conocer cuáles han sido los principales ataques llevados a cabo por ‘delincuentes’ cibernéticos, lo que permita trabajar sobre la lección aprendida y disminuir riesgos.

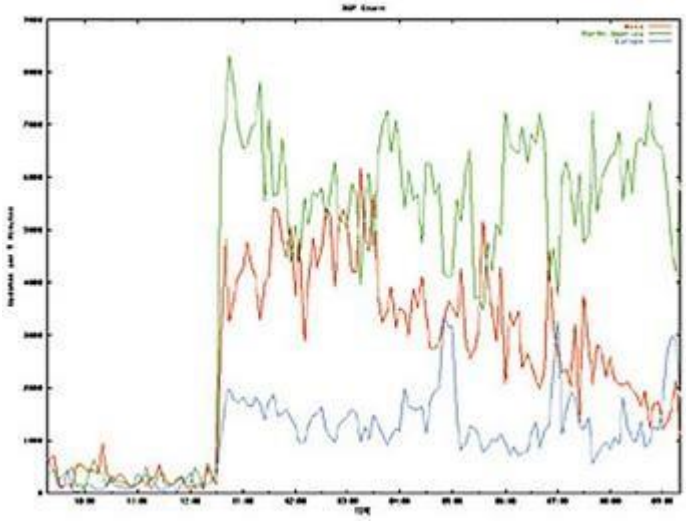
[10] Ibit, p. 28

[11] PINZÓN, Liliana. Pruebas de intrusión y metodologías abiertas. En: Revista Ciencia, Innovación y Tecnología (RCIYT). Enero-diciembre, 2013. Vol. 1. p. 29

4.3.1.1 EL Gusano Atacante (*Slammer*). El 25 de enero del 2003, un gusano conocido como *Slammer* infectó más de 75.000 máquinas en un lapso de 10 minutos[12], propagándose a una velocidad nuca antes vista hasta la fecha, generando denegación de servicio en algunos dominios y lentitud en el tráfico en general.

Este gusano explotaba una vulnerabilidad de **desbordamiento de *buffer***, en *Microsoft **SQL** Server* pública y con parche disponible desde 6 meses antes del ataque; representado de manera similar en la Figura 3. Muchos administradores no habían parcheado sus sistemas de forma adecuada, pero sobre todo la difusión tan masiva fue debido a que muchos usuarios no eran conscientes de tener instalado MS SQL Server Desktop *Engine* (*MSDE*), el cual dispone de un motor de MS SQL Server.

Figura 3. Aumento del tráfico BGP, indicativo de atascos en internet.



Fuente: DIÁZ SAÉZ, Vicente. Seguridad en bases de datos y aplicaciones web. [En línea]. Cataluña: Fundación para la Universidad Oberta de Catalunya, 2017., p. 8. Disponible en: https://www.exabyteinformatica.com/uoc/Informatica/Seguridad_en_bases_de_datos/Seguridad_en_bases_de_datos_(Modulo_1).pdf

Este es un claro ejemplo de una vulnerabilidad explotada exitosamente por parte de un atacante. En las bases de datos existen decenas de vulnerabilidades, con distintas dificultades de explotación y con distinto grado de divulgación, la evolución que han

[12] ARTEAGA, Sandra. Reaparece *SQL Slammer*, el gusano que colapsó Internet en 2003. [En línea]. Madrid: CH, 2017. Disponible en: https://computerhoy.com/noticias/software/reaparece-sql-slammer-gusano-que-colapso-internet-2003-58432

realizado estas vulnerabilidades y los ataques en bases de datos, son similares a la actualización de cualquier otro software según aumenta en popularidad y en complejidad.

Este ataque ilustra los siguientes aspectos:

- Es un buen ejemplo, de la gran difusión de las bases de datos, en este caso incluso sin el conocimiento de los usuarios de las mismas. El motor de base de datos estaba soportado en otro software.

- Una base de datos, sin desconocer los distintos conceptos que se le pueden definir, no deja de ser otro producto de software, susceptible a ataques específicamente orientados a obtener los datos que almacenan, o simplemente ataques genéricos que aprovechan una vulnerabilidad no corregida, como en cualquier otro software.

- Al estar conectadas al internet, las bases de datos son muy accesibles (vulnerables). Aunque la distribución del gusano *Slammer* fue indiscriminada, existen muchas técnicas para buscar bases de datos vulnerables a algún problema conocido, empleando herramientas comunes como lo puede ser Google.

- Hay que tener presente que la vulnerabilidad, no por ser conocida ni por disponer de un parque desde 6 meses antes del ataque, deja de ser fatal. Esto pone en tela de juicio el no contar con una adecuada política de actualizaciones y además de esto, tener en cuenta el factor humano, orientado a generar conciencia de Ciberseguridad.

4.3.1.2 El Caso *Marriott*. De acuerdo con lo expuesto por la BBC NEWS MUNDO. "El viernes 30 de noviembre *Marriott*, la cadena de hoteles más grande del mundo, sufrió un ataque informático sin precedentes. Y aunque no fue la peor violación de datos de la historia, figura en la lista de las más graves por número de afectados. El ataque afectó a una base de datos de reservas de 500 millones de clientes de su división *Starwood*, la cual cuenta con marcas internacionales como *Le Méridien* o *Sheraton*" [13].

Así mismo, la BBC NEWS MUNDO señala que, "en la base a donde accedió un hacker desconocido, había información 'combinada' que incluía detalles como nombres, direcciones, teléfonos, cuentas de email, números de pasaporte, horas de entrada y salida al hotel e incluso información cifrada de tarjetas de crédito (esto último solo en algunos casos). *Marriott* dice que los hackers 'copiaron y encriptaron información' de al

[13] *BBC NEWS MUNDO.* Cuáles fueron los peores hackeos informáticos de la historia y por qué el que sufrió *Marriott* es uno de los más graves. [En línea]. Londres: BBC, 2018. P. 1. Disponible en: https://www.bbc.com/mundo/noticias-46426990.

menos 327 millones de personas que figuraban en esa base de datos y que tuvieron acceso no autorizado a ella desde 2014”[14].

Para evitar este tipo de ataques se deben tener en cuenta los siguientes aspectos:

- Se debe contar con un plan de actualizaciones a nivel de sistema operativo y motor de base de datos, que permita contar con las últimas actualizaciones, teniendo especial cuidado de aplicarlas en un ambiente de pruebas, antes de llevarlas a producción.

- Utilizar cuentas de inicio de sesión, que correspondan a un dominio y utilizando la política del nivel de privilegios mínimo.

- No utilizar las cuentas propias del sistema operativo, para iniciar los servicios del motor de base de datos, ya que son cuentas con un nivel elevado de privilegios.

- No instalar el servidor web en el mismo lugar del servidor de base de datos.

- Cifrar la comunicación del canal de datos con el servidor de base de datos.

4.3.1.3 El Caso *Yahoo*. Según la *BBC NEWS MUNDO*[15], En agosto de 2013, se había producido el hasta ahora mayor hackeo corporativo del que se tiene constancia: el ataque masivo a *Yahoo*, que afectó a unos 3.000 millones de cuentas.

La compañía, que fue vendida a *Verizon* en julio de 2016, admitió en 2017 que el hackeo fue mayor de lo que pensaba, pues alcanzó a toda la plataforma digital y no sola a una parte de ella. En un comunicado oficial, *Oath*, la división de *Verizon* encargada de *Yahoo*, dijo en aquel momento que todas las cuentas de sus clientes se habían visto afectadas. Los *hackers* obtuvieron preguntas de seguridad y direcciones de email usadas para restablecer contraseñas. Pero poco más tarde, la empresa fue de nuevo víctima de los *hackers.* En diciembre de 2014, una filtración de datos a gran escala dejó expuestas al menos a 500 millones de cuentas (se supo públicamente en 2016). En esa ocasión, se comprometieron nombres, direcciones de *e-mail,* números de teléfono, contraseñas y, en algunos casos, preguntas de seguridad. El Departamento de Justicia de Estados Unidos culpó a cuatro individuos rusos[16].

[14]Ibit, p. 1.

[15]*BBC NEWS* MUNDO. Cuáles fueron los peores hackeos informáticos de la historia y por qué el que sufrió *Marriott* es uno de los más graves. [En línea]. Londres: BBC, 2018. P. 1. Disponible en: https://www.bbc.com/mundo/noticias-46426990.

[16]Ibit, p. 1.

Para evitar este tipo de ataques se deben tener en cuenta los siguientes aspectos:

- Contar con la política del principio de privilegios mínimo.

- Crear códigos *schemas* para clasificar y agrupar los objetos de la base de datos como tablas, funciones, procedimientos almacenados y vistas.

- Asignar permisos mediante el uso de roles en vez de asignarse directamente a los usuarios.

- Desactivar los usuarios por defecto del motor de base de datos.

- Utilizar las herramientas de auditoría, para realizar oportunamente los seguimientos en la base de datos.

- Monitorear los usuarios que tienen permisos de administrador sobre la base de datos.

4.3.1.4 ***Friend Finder y Equifax.*** En forma dada por la *BBC NEWS* MUNDO, "completan la lista de los mayores ataques informáticos conocidos hasta el momento el que afectó a la red de sitios web para adultos *Friend Finder* en 2016 y el hackeo a *Equifax* en 2017. En el caso del primero, se expusieron datos de al menos 416 millones de personas, según analistas de *Leaked Source*. El sitio había sido previamente hackeado en mayo de 2015, el ataque afectó a 146 millones de clientes" [17].

Para evitar este tipo de ataques se deben tener en cuenta los siguientes aspectos:

- Contar con las últimas actualizaciones del sistema operativo en el servidor y motor base de datos.

- Instalar y configurar correctamente herramientas firewall tanto físicas como lógicas con las debidas reglas de acceso.

- Restringir el acceso a la base de datos, para evitar conexiones utilizando internet, redes externas, conexiones inalámbricas.

- Realizar una correcta asignación de roles y/o permisos a los usuarios, con el fin de evitar otorgar permisos de súper usuario o administrador a usuarios que no deben contar con ese tipo de acceso.

[17] *BBC NEWS* MUNDO. Cuáles fueron los peores hackeos informáticos de la historia y por qué el que sufrió *Marriott* es uno de los más graves. [En línea]. Londres: BBC, 2018. P. 1. Disponible en: https://www.bbc.com/mundo/noticias-46426990.

4.4 MARCO LEGAL

4.4.1 Cumplimiento legal. En la república de Colombia, la ciberdelincuencia genera grandes pérdidas económicas y de información en las bases de datos, debido a que no aseguran a tiempo los sistemas informáticos y sus redes, por consiguiente, se da a conocer los lineamientos que tiene la normatividad vigente sobre la protección de la información y de los datos, con el fin de tenerlas en cuenta para ser aplicadas en casos de vulnerabilidad y amenazas en la información.

La Ley 1273 de 2009, por medio de la cual se modifica el Código Penal, se crea un nuevo bien jurídico tutelado - denominado "de la protección de la información y de los datos" y se preservan integralmente los sistemas que utilicen las tecnologías de la información y las comunicaciones, entre otras disposiciones[18].

EL CONGRESO DE COLOMBIA

DECRETA:

ARTÍCULO 1 ro. Adicionase el Código Penal con un Título VII BIS denominado “De la protección de la información y de los datos”, del siguiente contexto:

CAPITULO I

De los atentados contra la confidencialidad, la integridad y la disponibilidad de los datos y de los sistemas informáticos

Artículo 269A: Acceso abusivo a un sistema informático. El que, sin autorización o por fuera de lo acordado, acceda en todo o en parte a un sistema informático protegido o no con una medida de seguridad, o se mantenga dentro del mismo en contra de la voluntad de quien tenga el legítimo derecho a excluirlo, incurrirá en pena de prisión de cuarenta y ocho (48) a noventa y seis (96) meses y en multa de 100 a 1.000 salarios mínimos legales mensuales vigentes.
Artículo 269B: Obstaculización ilegítima de sistema informático o red de telecomunicación. El que, sin estar facultado para ello, impida u obstaculice el funcionamiento o el acceso normal a un sistema informático, a los datos informáticos allí contenidos, o a una red de telecomunicaciones, incurrirá en pena de prisión de cuarenta y ocho (48) a noventa y seis (96) meses y en multa de 100 a 1000 salarios mínimos legales mensuales vigentes, siempre que la conducta no constituya delito sancionado con una pena mayor.

[18] Congreso De La República. Presidencia de la República de Colombia. Ley 1273 (5, enero, 2009). De la protección de la información y de los datos. Bogotá D.C: El congreso, 2009. p. 1-3. http://www.secretariasenado.gov.co/senado/basedoc/ley_1273_2009.html

Artículo 269C: Interceptación de datos informáticos. El que, sin orden judicial previa intercepte datos informáticos en su origen, destino o en el interior de un sistema informático, o las emisiones electromagnéticas provenientes de un sistema informático que los transporte incurrirá en pena de prisión de treinta y seis (36) a setenta y dos (72) meses.

Artículo 269D: Daño Informático. El que, sin estar facultado para ello, destruya, dañe, borre, deteriore, altere o suprima datos informáticos, o un sistema de tratamiento de información o sus partes o componentes lógicos, incurrirá en pena de prisión de cuarenta y ocho (48) a noventa y seis (96) meses y en multa de 100 a 1.000 salarios mínimos legales mensuales vigentes.

Artículo 269E: Uso de software malicioso. El que, sin estar facultado para ello, produzca, trafique, adquiera, distribuya, venda, envíe, introduzca o extraiga del territorio nacional software malicioso u otros programas de computación de efectos dañinos, incurrirá en pena de prisión de cuarenta y ocho (48) a noventa y seis (96) meses y en multa de 100 a 1.000 salarios mínimos legales mensuales vigentes.

Artículo 269F: Violación de datos personales. El que, sin estar facultado para ello, con provecho propio o de un tercero, obtenga, compile, sustraiga, ofrezca, venda, intercambie, envíe, compre, intercepte, divulgue, modifique o emplee códigos personales, datos personales contenidos en ficheros, archivos, bases de datos o medios semejantes, incurrirá en pena de prisión de cuarenta y ocho (48) a noventa y seis (96) meses y en multa de 100 a 1000 salarios mínimos legales mensuales vigentes.

Artículo 269G: Suplantación de sitios web para capturar datos personales. El que con objeto ilícito y sin estar facultado para ello, diseñe, desarrolle, trafique, venda, ejecute, programe o envíe páginas electrónicas, enlaces o ventanas emergentes, incurrirá en pena de prisión de cuarenta y ocho (48) a noventa y seis (96) meses y en multa de 100 a 1.000 salarios mínimos legales mensuales vigentes, siempre que la conducta no constituya delito sancionado con pena más grave.

En la misma sanción incurrirá el que modifique el sistema de resolución de nombres de dominio, de tal manera que haga entrar al usuario a una IP diferente en la creencia de que acceda a su banco o a otro sitio personal o de confianza, siempre que la conducta no constituya delito sancionado con pena más grave.

La pena señalada en los dos incisos anteriores se agravará de una tercera parte a la mitad, si para consumarlo el agente ha reclutado víctimas en la cadena del delito.

Artículo 269H: Circunstancias de agravación punitiva: Las penas imponibles de acuerdo con los artículos descritos en este título, se aumentarán de la mitad a las tres cuartas partes si la conducta se cometiere:

1. Sobre redes o sistemas informáticos o de comunicaciones estatales u oficiales o del sector financiero, nacionales o extranjeros.

2. Por servidor público en ejercicio de sus funciones.

3. Aprovechando la confianza depositada por el poseedor de la información o por quien tuviere un vínculo contractual con este.

4. Revelando o dando a conocer el contenido de la información en perjuicio de otro.

5. Obteniendo provecho para sí o para un tercero.

6. Con fines terroristas o generando riesgo para la seguridad o defensa nacional.

7. Utilizando como instrumento a un tercero de buena fe.

8. Si quien incurre en estas conductas es el responsable de la administración, manejo o control de dicha información, además se le impondrá hasta por tres años, la pena de inhabilitación para el ejercicio de profesión relacionada con sistemas de información procesada con equipos computacionales.

5. DESARROLLO DE LOS OBJETIVOS

5.1. EXPLORAR LOS RIESGOS Y AMENAZAS QUE ESTÉN ASOCIADOS A LAS VULNERABILIDADES EN LAS BASES DE DATOS RELACIONALES

5.1.1. Según IMPERVA[19], las 10 principales amenazas en entornos de bases de datos relacionales son:

5.1.1.1. Privilegios excesivos. Se produce este evento innecesario, cuando a los usuarios se les otorgan permisos especiales para manipular las bases de datos, sin verificar previamente el perfil del usuario. El mecanismo de control de privilegios de roles laborales debe estar claramente definido o mantenido.

5.1.1.2. Abuso de privilegios. Esto sucede cuando los usuarios buscan abusar de los privilegios legítimos de la base de datos con fines fraudulentos o por beneficio propio, para el robo o alteración de información confidencial. Cuando el registro de información llega al cliente, los datos se muestran en varias categorías de infracción.

5.1.1.3. Ataques de inyección *SQL*. Al momento de hablar de ataques de inyección SQL, se hace referencia a un método que se aprovecha de errores que existen en aplicaciones web. Son básicamente vulnerabilidades que permiten a un posible intruso inyectar código malicioso para llevar a cabo sus ataques y comprometer la seguridad y privacidad de los usuarios.

Un ataque de inyección *SQL* podría estar orientado en comprometer páginas web o bases de datos. Un pirata informático podría manipular, robar o eliminar información y datos que hay en esas webs comprometidas o bases de datos.

Los ataques de inyección *SQL* se basan en vulnerabilidades existentes, es por esto, que es vital mantener siempre los sistemas, dispositivos y cualquier software que se utilice correctamente actualizado. De esta forma se podrán corregir errores de seguridad que puedan ser utilizados por terceros para llevar a cabo sus ataques. Se deben mantener siempre los últimos parches y actualizaciones. Por ejemplo, “un ciberdelincuente podría inyectar consultas *SQL* maliciosas, en el campo de entrada de una *web*. De esta

[19] *UNITED STATES GOVERNMENT*. Comunes, vulnerabilidades y exposiciones. [En línea]. *Gaithersburg:* CVE Mitre. 2019., p. 3. Disponible en: https://nvd.nist.gov/vuln/detail/CVE-2019-10749.

forma, podría engañar al sitio para que ejecute comandos y acceder a su base de datos. Lógicamente un ataque de este tipo, puede hacer que el funcionamiento de la web no sea el adecuado. Puede afectar al rendimiento. Sin embargo, también afecta a la privacidad al robar datos e información, modificarlo o incluso eliminarlo"[20].

5.1.1.4. ***Malware y spear phising*.** Es una técnica sofisticada utilizada por ciberdelincuentes, piratas informáticos o espías, para ingresar a una organización y robar datos confidenciales. El *spear phising* tiene como misión principal la estafa por correo electrónico o comunicaciones a través del mismo, está orientado a personas y/u organizaciones especificas; funciona por medio de señuelos falsos, utiliza la publicidad para atraer a las víctimas y finalmente conducirlas a un sitio web falso con gran diversidad de malware.

5.1.1.5. **Auditorías débiles**. No recopilar registros de auditoría detallados puede representar un riesgo muy grave para la organización en muchos niveles.

5.1.1.6. **Exposición de los medios de almacenamiento para *backup***. A menudo no estar protegido, genera como resultado muchas violaciones de seguridad informática. Teniendo en cuenta que, con la perdida de discos y copias de seguridad, los datos pueden verse comprometidos si los administradores no pueden controlar y monitorear el acceso de bajo nivel a la información confidencial.

5.1.1.7. **Explotación de vulnerabilidades y bases de datos mal configuradas**. Los atacantes informáticos, emplean herramientas con el fin de verificar el estado de seguridad de las bases de datos y estas a su vez en ocasiones se encuentran en mayor riesgo de vulnerabilidad debido a la falta de actualización o configuraciones por defecto.

5.1.1.8. **Datos sensibles mal gestionados**. Este tipo de información generará riesgos a las bases de datos ya que, al no efectuar controles y permisos especiales, los atacantes informáticos podrán explotar dicha vulnerabilidad.

[20] UNITED STATES GOVERNMENT. Comunes, vulnerabilidades y exposiciones. [En línea]. Gaithersburg: CVE Mitre. 2019., p. 3. Disponible en: https://nvd.nist.gov/vuln/detail/CVE-2019-10749.

5.1.1.9. Denegación de servicio *(DoS)*. Cuando se realiza este ataque, la misión es inhabilitar el uso ya sea, de un sistema, una aplicación o un equipo; con la finalidad de bloquear el servicio y el acceso a los datos de la red. Otro tipo de ataque muy parecido es el ***DDoS***, tiene como finalidad enviar varias solicitudes al sistema atacado, con la intención de desbordar su capacidad para administrar varias solicitudes y de evitar que este funcione correctamente.

5.1.1.10. Falta de conocimiento y experiencia en seguridad informática. En la actualidad muchas compañías, carecen de personal capacitado en seguridad informática, para efectuar la detección oportuna de vulnerabilidades y la mitigación de las mismas; la falta de este tipo de personal especializado se convierte en una vulnerabilidad dentro de la compañía.

5.1.2. Vulnerabilidades más comunes en las bases de datos. A continuación, se evidenciarán vulnerabilidades más habituales encontradas a la hora de trabajar con bases de datos[21].

5.1.2.1. Nombre de usuario/*password* en blanco o bien hacer uso de uno débil. En esta época los usuarios carecen de principios de seguridad, debido a la falta de orientación y capacitación; un claro ejemplo de esto son los tipos de usuarios y la utilización de (usuario/*password* del tipo *admi*n/12345) o similar. Esta información es la primera línea de defensa de entrada, a todo sistema informático, debido a esto se deben emplear el uso de caracteres especiales o algo más complejo y de esta forma, evitar un ataque de fuerza bruta.

5.1.2.2. Preferencia de exenciones de usuario por privilegios de grupo. El personal encargado de otorgar permisos especiales y de establecer los tipos de usuarios, debe contar con un filtro muy riguroso para realizar estas distinciones, ya que el mal otorgamiento de estos permisos, puede ocasionar un importante problema dentro de la empresa. Es recomendable controlar los privilegios concedidos a los usuarios que estarán en interacción con la información, con el fin de limitar las alteraciones no autorizadas.

[21] *UNITED STATES GOVERNMENT.* Comunes, vulnerabilidades y exposiciones. [En línea]. *Gaithersburg: CVE Mitre. 2019., p. 3. Disponible en: https://nvd.nist.gov/vuln/detail/CVE-2019-10749.*

5.1.2.3. Características de bases de datos innecesariamente habilitadas. Cuando se produce la instalación de las bases de datos, estas vienen con una cadena de paquetes o módulos anexos de distintas formas y tamaños, que en casi siempre no son utilizadas, esto se convierte en una potencial puerta de entrada para sufrir algún tipo de ataque informático; si en esos paquetes se identifican cualquier problema de seguridad; es recomendable que los interesados identifiquen esos paquetes que no se utilizan y se desactiven del servidor donde estén instalados. Esto no solo reduce los riesgos de ataques, sino que también simplifica la gestión de parches, ya que únicamente será de máxima urgencia actualizar aquellos que hagan referencia a un módulo que se esté manejando.

5.1.2.4. Desbordamiento de búfer. Es uno de las rutas preferidas manejadas por los piratas informáticos y se produce por el exceso de información enviada por medio del ingreso de datos, mediante el uso de formularios, es decir, se recibe mucha más información de lo que el sistema espera. Por poner un ejemplo, si se espera la entrada de una cuenta bancaria que puede ocupar unos 15 caracteres y se permite la entrada de muchos más caracteres desde ese campo, se podría dar este tipo problema informático.

5.1.2.5. Bases de datos sin actualizar. Esto ocurre con cualquier tipo de software que se encuentre instalado en un host, por tal motivo es necesario ir actualizando la versión de las bases de datos y tener todo el sistema debidamente licenciado; ya que, cuando se realiza esta buena práctica, se solucionan aquellos problemas de seguridad informática detectados y de esta forma, se aumentan los obstáculos de seguridad a los atacantes informáticos.

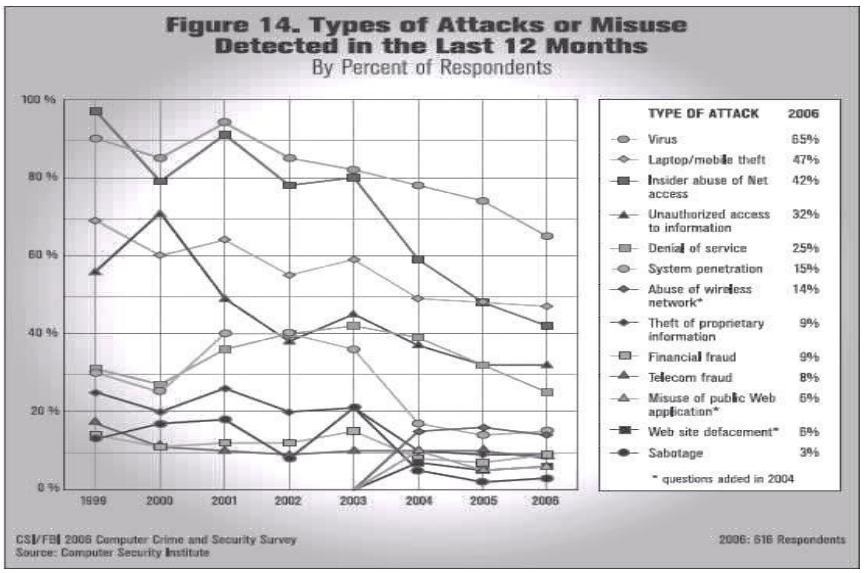
5.1.2.6. Datos sensibles sin cifrar. No todo el mundo cifra la información más importante almacenada en una base de datos, lo cual no es sorprendente. Esta es una excelente manera de dificultar que un atacante obtenga esta información en caso de un ataque. Para este tipo de problema de seguridad se debe implementar la criptografía y en ocasiones la estenografía, con el fin de proteger la información.

5.1.2.7. Inyecciones SQL. Esta ofensiva informática permite al atacante acceder a la base de datos sin ningún tipo de restricción y se produce a través de la inyección de código malicioso. La limpieza incorrecta de los datos de entrada puede provocar la ejecución inesperada de malware en los controles de la plataforma y la base de datos (la falla principal está ligada a la programación). La solución más recomendable para este tipo de ataque consta de un adecuado filtro en los *inputs* de los formularios web.

5.1.3 Ataques informáticos asociados a las bases de datos. Según los datos de la encuesta de seguridad anual del FBI, los virus informáticos continúan siendo una fuente importante de pérdidas financieras para las organizaciones, incluido el acceso no autorizado a los sistemas, el robo de información de posesión y el robo de información industrial y la pérdida de componentes de computadoras personales o móviles. Estas causas generan más del 74% del total de las pérdidas financieras. La inversión en tecnología para proteger la seguridad de la información aumenta día a día, pero, no obstante, se muestran las investigaciones y los resultados que se muestran en los siguientes gráficos. Las situaciones de Ciberseguridad o ansiedad cibernética surgen no solo de personas negligentes que divulgan información sensible, sino también de grandes corporaciones multinacionales que cuentan con suficientes departamentos técnicos y recursos para invertir en protección[22].

El FBI estableció una estadística de ataques y crimines cibernéticos, los cuales se muestran a continuación en la Figura 4.

Figura 4. Estadísticas de ataques



Fuente: ÁLVAREZ, Daniel Luz. Desafíos en la agenda regional de seguridad ciudadana y criminalidad transnacional organizada. En: Desafíos en la agenda regional de seguridad ciudadana y criminalidad transnacional organizada. Diciembre, 2014. Vol. 15., p. 33-54. Disponible en: https://revistas.uexternado.edu.co/index.php/derpen/article/view/9652006

[22] TARAZONA, Cesar. Amenazas informáticas y seguridad de la información. Derecho Penal y Criminología. En: Amenazas informáticas y seguridad de la información. Agosto, 2007. Vol 28., no. 84, p. 137-146.

Según *Security Absurdity* “un estudio reciente publicado por *Avante Garde*, donde se prueba un sistema configurado de fábrica con protecciones básicas que permanecía conectado a Internet”. El tiempo promedio que una computadora ‘exitosamente’ es de solo minutos. Lo que hacen personas en su nueva computadora es conectar su computadora a Internet, enviar correo electrónico, descargar archivos (música, video, etc.), navegar, jugar, ‘chatear’ y otras actividades. Pocos piensan en proteger las computadoras, instalar parches o herramientas de seguridad. (Cortafuegos personal, antivirus, *antispam*, e incluso copias de seguridad)”[23].

Las principales vulnerabilidades que permiten que los usuarios sean pirateados y víctimas de muchas amenazas contra nosotros, estás son a menudo el hecho de que la tecnología no se gestiona dentro de un marco integral de protección de la información e Internet. Los riesgos asociados con el uso de estas tecnologías y herramientas. Los esfuerzos para hacerlo pueden ser en vano o las metas pueden lograrse incorrectamente. La inversión en tecnología de seguridad como solución a los problemas que plantea debe realizarse dentro de un marco que se integre con otro conjunto de medios para formar lo que se denomina un "sistema de gestión de seguridad", la información[24].

[23] *EPPEL, Noam. Security Absurdity: The Complete, Unquestionable, And Total Failure of Information Security.* En: *securityabsurdity*. Noviembre, 2006. Vol. 1, p. 7-30.
[24] Ibid., p. 18-30.

5.2. ESTABLECER HERRAMIENTAS QUE PROPORCIONEN EL ANÁLISIS DE VULNERABILIDADES Y FACILITEN LA IDENTIFICACIÓN DE RIESGOS EN LAS BASES DE DATOS RELACIONALES

5.2.1. *DMitry*. Es una herramienta que utiliza líneas de comando *UNIX/GNU/LINUX*, en leguaje C, de código abierto, fácil de usar y muy utilizada especialmente para captar información, ayuda a descubrir, explorar y operar sistemas de información y bases de datos[25], su interface es sencilla visualmente como lo muestra la Figura 5. Actualmente soporta cantidad de servidores de bases de datos, entre los que se incluyen:

- *MySQL*
- *Oracle*
- *PostgreSQL*
- *Microsoft SQL Server*
- Servidor *Web Apache*
- *Firebird*
- *MongoDB*

Adicionalmente, se ha permitido explorar en sus capacidades y reconocer las características técnicas de esta herramienta con el fin de evaluar su operatividad y aporte al trabajo del programador, usuario o compañía, de la siguiente forma:

Ventajas: Algunas de las ventajas de la herramienta *DMitry*, son:

- Reconocimiento y escaneo automático con *NMAP, whataweb, nikto, Vulners, Hydra, SMBenum, dirbuster, sslyzer, webslayer.*
- Escaneo altamente personalizable para evasión de IPS tipo ninja.
- Detección automática de CPE (enumeración de plataforma común) y *CVE* (vulnerabilidades y exposiciones comunes).
- Guardado automático de resultados y tareas del proyecto en tiempo real.

Desventajas: Algunas de las desventajas de la herramienta *DMitry*, son:

- No se recomienda para personal novato en temas de *pentesting*, ya que la interface que utiliza, está basada en líneas de comandos.
- No es compatible con todos los sistemas operativos.

[25] *HOLEADER, Jack. Cara Scan Website Dengan DMitry di Kali Linux| Information Gathering. [En línea]. Estambul: Rozak, 2017., Disponible en: https://www.abdurrozak.my.id/2017/04/cara-scan-website-dengan-dmitry-di-kali.html*

Figura 5. Interface *DMitry*

Fuente: *HOLEADER, Jack. Cara Scan Website Dengan DMitry di Kali Linux*| *Information Gathering. [En línea]. Estambul: Rozak, 2017., Disponible en: https://www.abdurrozak.my.id/2017/04/cara-scan-website-dengan-dmitry-di-kali.html*

5.2.2. *Nmap*. Es un software que incorpora varias opciones para analizar redes y sistemas de bases de datos, como detección de dispositivos, servicios y sistemas operativos. Estas funciones se pueden programar y proporcionan detección avanzada, detección de vulnerabilidades y otras aplicaciones. Al mismo tiempo, es una herramienta muy flexible a la hora de analizar como lo muestra su interface en la Figura 6, adaptándose a las condiciones de la red como retrasos y congestión. Actualmente presenta las siguientes características de escaneo, entre las que se incluyen:

Ventajas: Algunas de las ventajas de la herramienta *Nmap*, son:

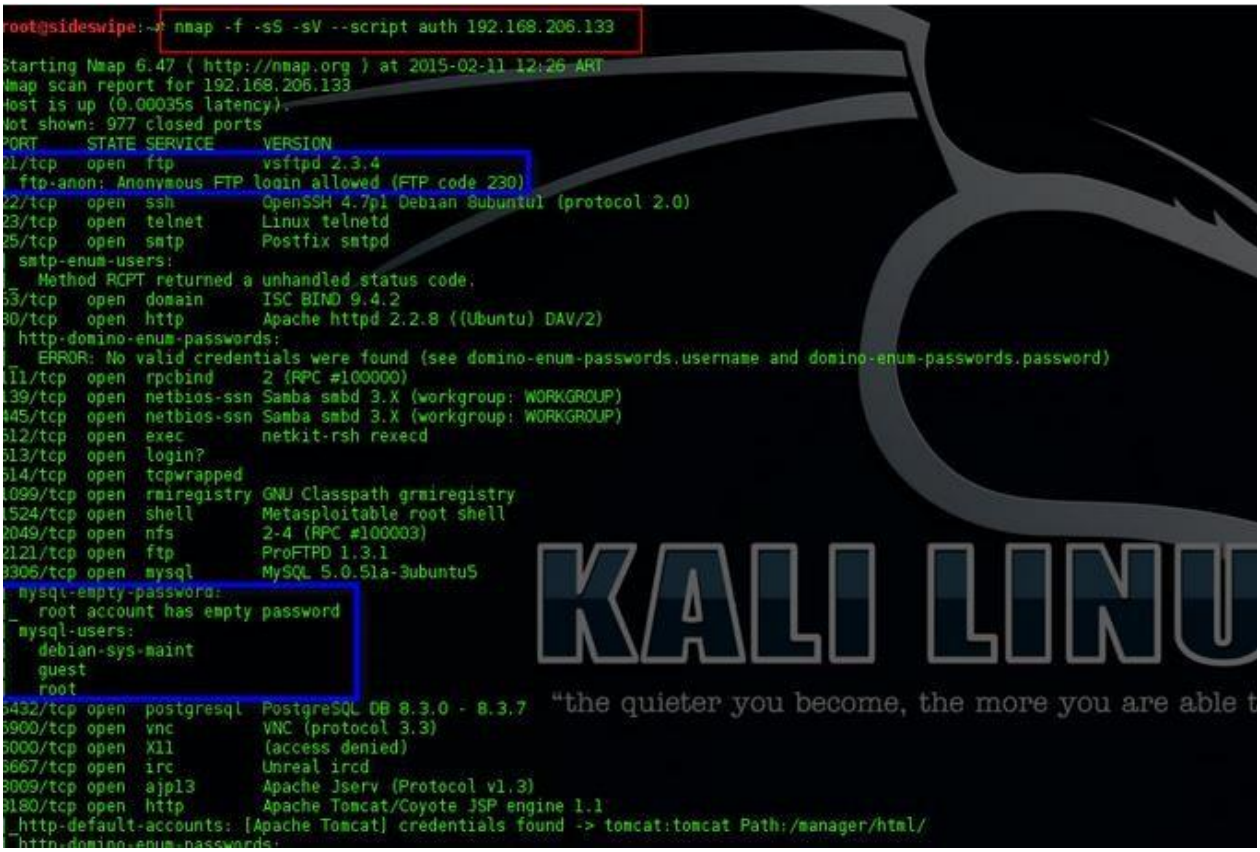
- Ejecución de Inyección *SQL*
- Inyección de código *JSP* / *ASP* / *PHP*
- Ejecución de comando
- Divulgación de archivos
- Scripting de sitios cruzados (XSS)
- Protocolos de red
- Soporta motor Mongo DB

Desventajas: Algunas de las desventajas de la herramienta *Nmap*, son:

- Su fuerte no está enfocado a las bases de datos.
- Nada recomendable para personal novato en temas de *pentesting*, ya que la interface utilizada por él, está basada en líneas de comandos.

- No brinda importaciones en formatos PDF o TXT sobre informe detallado del proceso de detección.
- No es compatible con todos los sistemas operativos.

Figura 6. Interface *Nmap*



Fuente: COSTES, Chris. Auditando con Nmap y sus scripts para escanear vulnerabilidades. [En línea]. Bratislava *Slovak Republic*, 2015., p. 1. Disponible en: https://www.welivesecurity.com/la-es/2015/02/12/auditando-nmap-scripts-escanear-vulnerabilidades/

5.2.3. *Nikto*. Es un escáner de vulnerabilidades de código abierto, escrito en el lenguaje Perl y lanzado por primera vez en 2011. Ofrece la capacidad de escanear servidores web en busca de vulnerabilidades, sus características visuales descritas en la Figura 7 y sus características de escaneo, incluyen:

Ventajas: Algunas de las ventajas de la herramienta *Nikto*, son:

- Verifica problemas de inyección de código JSP / ASP / PHP

- Divulgación de archivos
- Scripting de sitios cruzados (XSS)
- Protocolos de red
- Motor de base de datos Mongo DB y Mysql
- Verifica estado de servidores Web

Desventajas: Algunas de las desventajas de la herramienta *Nikto*, son:

- Su fuerte no está enfocado a las bases de datos.
- Todos estos hallazgos deben ser verificados manualmente, ya que podrían ser falsos positivos generadas por la herramienta.
- No es recomendable para personal novato en temas de pentesting, ya que la interface que utiliza, está basada en líneas de comandos.
- No brinda importaciones en formatos PDF o TXT sobre informe detallado del proceso de detección.
- No es compatible con todos los sistemas operativos.

Figura 7. Interface *Nikto*

Fuente: CABALLERO QUEZADA, Alonso Eduardo. Escanear un Servidor Web utilizando Nikto. [En línea]. Lima, 2018., p. 1. Disponible en: http://www.reydes.com/d/?q=Escanear_un_Servidor_Web_utilizando_Nikto

4..4 *Sqlmap.* "Es una herramienta de prueba de penetración a código abierto que automatiza el proceso de detección y explotación de errores de servidor de base de datos e inyección *SQL.* Posee un potente motor de detección, funciones completas para los principales probadores de penetración y una amplia gama de opciones como lo muestra la Figura 8, desde la toma de huellas digitales de la base de datos hasta la recuperación de información"[26]. Actualmente soporta cantidad de servidores de bases de datos, entre los que se incluyen:

- *MySQL*
- *Oracle*
- *Postgresql*
- *Microsoft SQL Server*
- *IBM DB2*
- *SQLLit*
- *Firebird*
- *Sybase*
- *SAP MaxDB*

Ventajas: Algunas de las ventajas de la herramienta *Sqlmap*, son:

- Enfocada a no permitir la explotación de fallas de inyección *SQL.*
- Posee una amplia gama de conmutadores direccionados a las bases de datos.
- Funciona a través de la ejecución de comandos en el sistema operativo por medio de conexiones de banda.
- Soporte para enumerar usuarios, hashes de contraseñas, privilegios, roles, bases de datos, tablas y columnas.
- Reconocimiento automático de formatos hash de contraseñas y soporte para descifrarlos usando un diccionario

Desventajas: Algunas de las desventajas de la herramienta *Sqlmap*, son:

- No es recomendable para personal novato en temas de *pentesting*, ya que la interface que utiliza, está basada en líneas de comandos.
- No brinda importaciones en formatos PDF o TXT sobre informe detallado del proceso de detección.
- No es compatible con todos los sistemas operativos.

[26] AVILA, Fredy. *Sqlmap,* herramienta automática para pruebas de penetración (Sql *Injection*. [En línea]. Bogotá D. C., 2018., p. 1. Disponible en: https://securityhacklabs.net/articulo/sqlmap-herramienta-automatica-para-pruebas-de-penetracion-sql

Figura 8. Interface Sqlmap

Fuente: AVILA, Fredy. Sqlmap, herramienta automática para pruebas de penetración (Sql Injection. [En línea]. Bogotá D. C., 2018., p. 1. Disponible en: https://securityhacklabs.net/articulo/sqlmap-herramienta-automatica-para-pruebas-de-penetracion-sql

5.2.4. *Metasploit framework.* Una vez que se han identificado el servicio y sus vulnerabilidades, el siguiente paso es explotarlas. Es decir, primero se debe verificar si las vulnerabilidades identificadas realmente pueden ser explotadas por un atacante. Se debe averiguar cuál es el daño, si se explota esta vulnerabilidad, en función de la información o los sistemas "detrás" de la vulnerabilidad.

Para este propósito, "*Metasploit* es una herramienta ideal para realizar estas pruebas. *Nessus* tiene una base de datos de vulnerabilidades, mientras que *Metasploit* tiene una base de datos de vulnerabilidades que pueden ser explotadas. Es decir, en lugar de buscar vulnerabilidades en un equipo remoto, el *exploit* se realiza directamente y si tiene éxito, se simulan los resultados posteriores"[27].

Sistemas soportados:

- *Website fingerprinting*
- *Mysql*
- *Metasploit*
- *Local/Remote File Inclusion*
- *Discovery of sensitive files*

[27] BORTNIK, Sebastián. Pruebas de penetración para principiantes: 5 Herramientas para empezar. [En línea]. Ciudad de México, 2020. Disponible en: https://revista.seguridad.unam.mx.

- URL
- *Oracle*
- Servidores Web

5.2.5. Vega. Es una herramienta de prueba de seguridad automatizada que rastrea un sitio web, analizando el contenido de la página para encontrar enlaces y parámetros de formulario. Entre sus objetivos está el de encontrar puntos de inyección, denominados nodos de estado de ruta, y ejecuta módulos escritos en *Javascript* para analizarlos. También ejecuta módulos de *Javascript* en todas las respuestas enviadas desde el servidor durante el escaneo.

Adicionalmente, Vega se ha permitido explorar en sus capacidades y reconocer las características técnicas de esta herramienta con el fin de evaluar su operatividad y aporte al trabajo del programador, usuario o compañía, de la siguiente forma:

Ventajas: Algunas de las ventajas de la herramienta Vega, son:

- La interface es agradable y muy fácil de utilizar, como se muestra en la Figura 9
- Incorpora una cantidad considerable de módulos de auditoría
- El consumo de recursos informáticos es bajo
- La herramienta es gratuita
- Es multi plataforma (Linux, Mac, Windows)

Desventajas: Algunas de las desventajas de la herramienta Vega, son:

- Existen algunas fallas en los módulos a la hora de localizar procesos relativamente obvios.
- No ofrece importaciones en formatos PDF o TXT sobre informe detallado del proceso de detección.

Figura 9. Interface Vega

Fuente: YAGO, Jesús. Vega, herramienta para auditar *Websites.* [En línea]. Bogotá D. C., 2013., p. 1. Disponible en: http://www.securitybydefault.com/2013/06/vega-herramienta-para-auditar-websites.html

5.2.6. ***Intruder.*** Es un escáner de vulnerabilidades automatizado idóneo para identificar debilidades de ciberseguridad en la infraestructura digital de una organización, evitando pérdidas económicas o la exposición de datos. Generalmente encuentra fallas como configuraciones incorrectas, parches faltantes, debilidades de cifrado y errores de aplicación (incluida la inyección de *SQL* y secuencias de comandos entre sitios) en áreas no autenticadas.

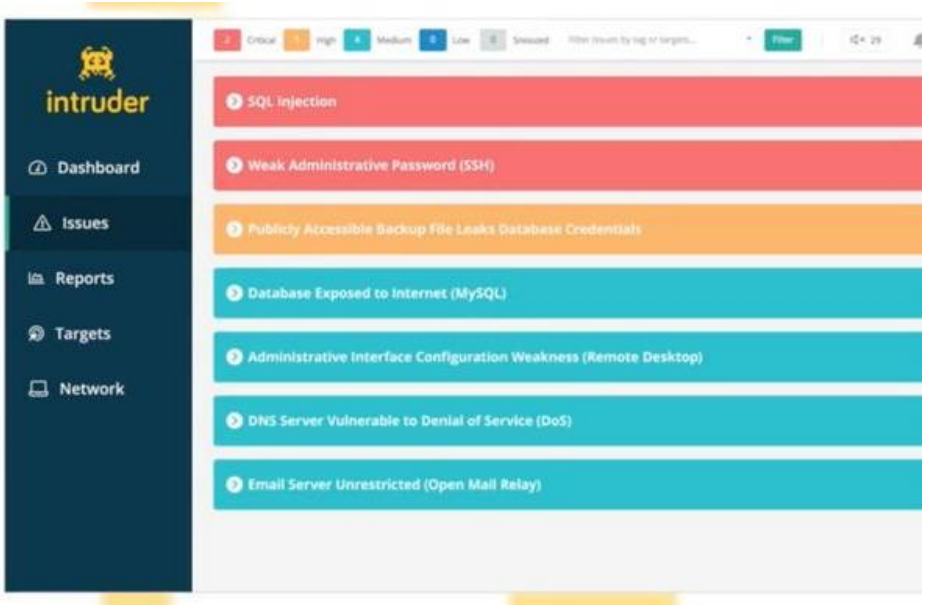
Ventajas: Algunas de las ventajas de la herramienta *Intruder*, son:

- La interface es sencilla y muy fácil de utilizar, puntualizada en la Figura 10.
- Ofrece la opción de determinar debilidades de la infraestructura, como la posibilidad de ejecución remota de código.
- Identifica errores de configuración de seguridad, como cifrado débil y servicios expuestos innecesariamente.

Desventajas: Algunas de las desventajas de la herramienta *Intruder*, son:

- Limita su utilización en relación al ancho de banda de la red utilizada.
- No ofrece importaciones en formatos PDF o TXT sobre informe detallado del proceso de detección.
- No es gratuita.

Figura 10. Interface *Intruder*



Fuente: *LAKSHMAN, Sharma*. 8 software de prueba de penetración *premium* para aplicaciones web. [En línea]. Madrid, 2015., p. 1. Disponible en: https://geekflare.com/es/premium-pen-testing-software/

5.2.7. *MySQL Enterprise Audit*. Este software suministra una solución de auditoría basada en políticas fácil de usar que ayuda a las organizaciones a implementar controles de seguridad más estrictos y satisfacer el cumplimiento normativo. A medida que se recopilan, almacenan y utilizan más datos confidenciales en línea, la auditoría de bases de datos se convierte en un componente esencial de cualquier estrategia de seguridad.

Ventajas: Algunas de las ventajas de la herramienta *MySQL Enterprise Audit,* son:

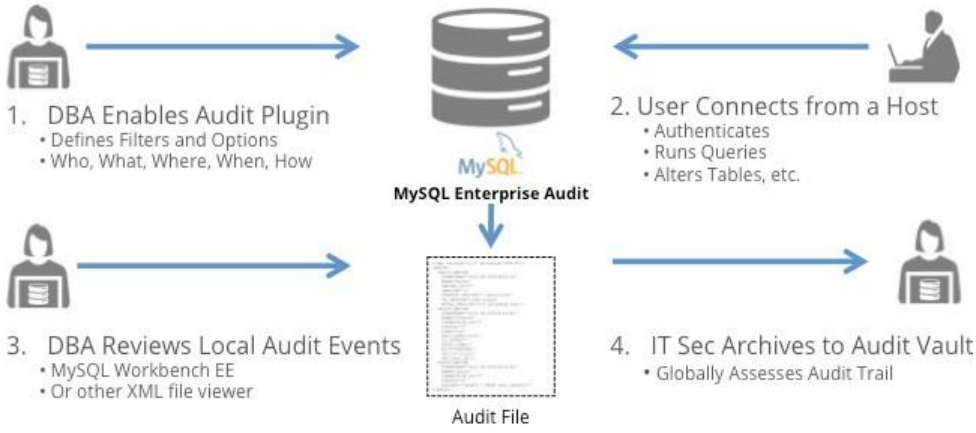
- Maneja eventos de registro de auditoria personalizados, los cuales pueden ser definidos por los usuarios.

- Con esta herramienta los archivos de auditorías resultantes, pueden ser cifrados utilizando el estándar AES-256. Estos archivos pueden compartirse y ser descifrados por aplicaciones externas que se les otorga la clave de cifrado.
- Ofrece la opción de compresión, la cual ayuda a reducir considerablemente el almacenamiento hasta 10 veces y además de esto, los archivos podrán ser descomprimidos por herramientas comunes.
- Sofisticado filtrado para proteger datos confidenciales, con la opción de definir previamente lo que se pretende auditar, utilizando plantillas o una elección de filtro *JSON* simple.
- Dinámico y fácil de administrar, con la opción de habilitar o deshabilitar, dinámicamente el flujo de auditoría, además se puede cambiar el filtrado sin tiempo de inactividad, como lo muestra la Figura 11.

Desventajas: Algunas de las desventajas de la herramienta *MySQL Enterprise Audit,* son:

- Limita su implementación, teniendo en cuenta que el proceso de auditoria se realiza en línea.
- No ofrece importaciones en formatos PDF o TXT sobre informe detallado del proceso de detección.
- No es gratuita.

Figura 11. Diagrama *MySQL Enterprise Audit*



Fuente: *ORACLE. MySQL Enterprise Audit.* [En línea]. Toronto, 2021., p. 1. Disponible en: https://www.mysql.com/products/enterprise/audit.html

5.2.8. Idera SQL Compliance Manager. Este aplicativo orientado a proteger las bases de datos, en las cuales se almacena el activo más importante de una empresa la información, utiliza solidas funciones de alerta y herramientas de informes de auditoría. Así mismo, brinda la opción de afrontar los estrictos requisitos de cumplimiento normativo de la industria y garantiza la compatibilidad entre servidor Sql con *HIPAA, GDPR.*

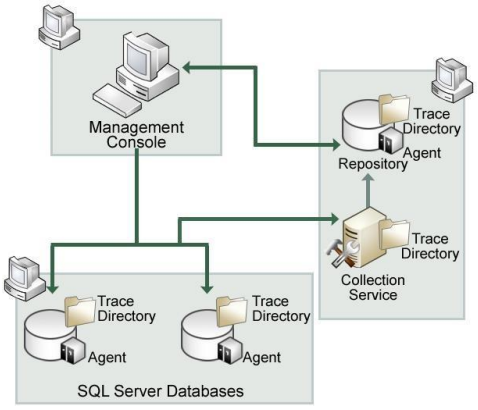
Ventajas: Algunas de las ventajas de la herramienta *Idera SQL Compliance Manager,* son:

- Por medio de esta herramienta se tiene la facilidad de monitorear alertar y registrar el acceso a bases de datos y servidores en tiempo real.
- Proporciona el escaneo a las bases de datos, identificado datos personales y otros datos sensibles de la empresa.
- Como resultado del proceso de escaneo, crea un registro de auditoría e informes confiables de eventos de seguridad.
- Valida que el repositorio de rastros de auditoria no haya sido alterado o manipulado.
- Utiliza un proceso que interactúa con el equipo, el y su herramientas de manera conjunta tal como lo muestra la Figura 12.

Desventajas: Algunas de las desventajas de la herramienta *Idera SQL Compliance Manager,* son:

- El entorno interactivo solo puede ser utilizado en lenguaje inglés.
- No ofrece importaciones en formatos PDF o TXT sobre informe detallado del proceso de detección.
- No es gratuita.

Figura 12. Diagrama *Idera SQL Compliance Manager*



Fuente: *IDERA. Product components and architecture*. [En línea]. 2013., p. 1. Disponible en: https://www.idera.com/help/sqlcm/4-3/web/content/architecture%20and%20components.htm

5.2.9. *Imperva Analytics.* Esta herramienta automatiza la detección de comportamientos de acceso a datos no compatibles. Además de esto, identifica riesgos o intenciones maliciosas en todas las bases de datos de las empresas que requieran de este servicio.

Ventajas: Algunas de las ventajas de la herramienta *Imperva Analytics,* son:

- Emplea algoritmos específicamente diseñados para reducir los falsos positivos.
- Alerta la generación amenazas críticas en un lenguaje sencillo.
- *Imperva* revela automáticamente el comportamiento de acceso a los datos, ya sea accidental, deficiente o determinadamente malicioso.
- Identifica el riesgo real en lugar de perder el tiempo en sospechas.

Desventajas: Algunas de las desventajas de la herramienta *Imperva Analytics,* son:

- La detección de vulnerabilidades es muy selectiva, limitando el abarcamiento de amenazas.
- No ofrece importaciones en formatos PDF o TXT sobre informe detallado del proceso de detección.
- No es gratuita.

Figura 13. Interface *Imperva*

Fuente: IMPERVA. Visibility into a broad range of risks**.** [En línea]. Texas, 2021., p. 1. Disponible en: https://www.imperva.com/products/data-user-behavior-analytics/

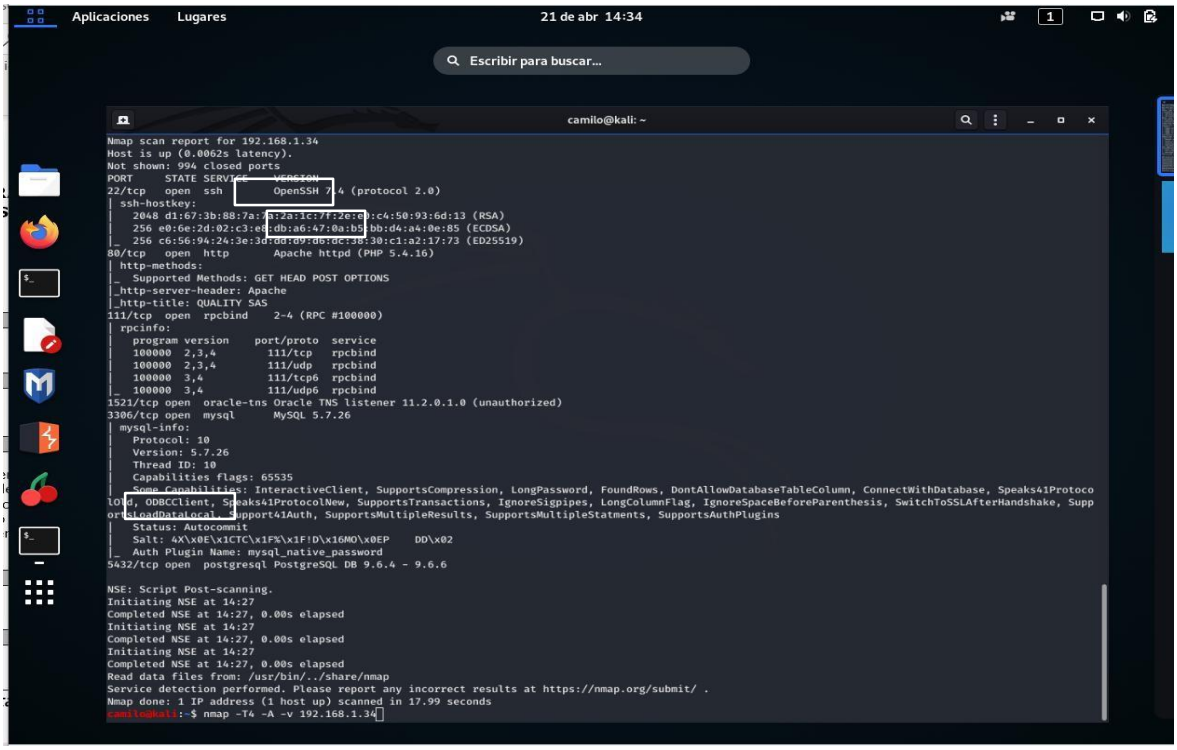
5.3. PROBAR EL FUNCIONAMIENTO DE HERRAMIENTAS UTILIZADAS PARA LA DETECCIÓN DE VULNERABILIDADES EN LAS BASES DE DATOS RELACIONALES

Para el desarrollo de este objetivo, se toma como referencia una base de datos relacional Mysql alojada en la dirección IP 192.168.1.34. Máquina virtual (objetivo) con sistema operativo Linux y una máquina virtual (atacante) soportada en un sistema operativo Kali Linux.

5.3.1. ***Nmap.*** Mediante el funcionamiento de la siguiente herramienta versión utilizada 7.92, se evidencia los puertos abiertos y cerrados que tiene la base de datos, de igual manera los servidores que se están manejando junto con las versiones.

En la Figura 14, se observa un escaneo básico de una base de datos, en donde se evidencia puertos *TCP* abiertos, las versiones que están instaladas en esta base de datos, *shh host*, para este caso, es el motor de búsqueda de *Postgresql* y se identifica las versiones 9.6.4 y 9.6.6.

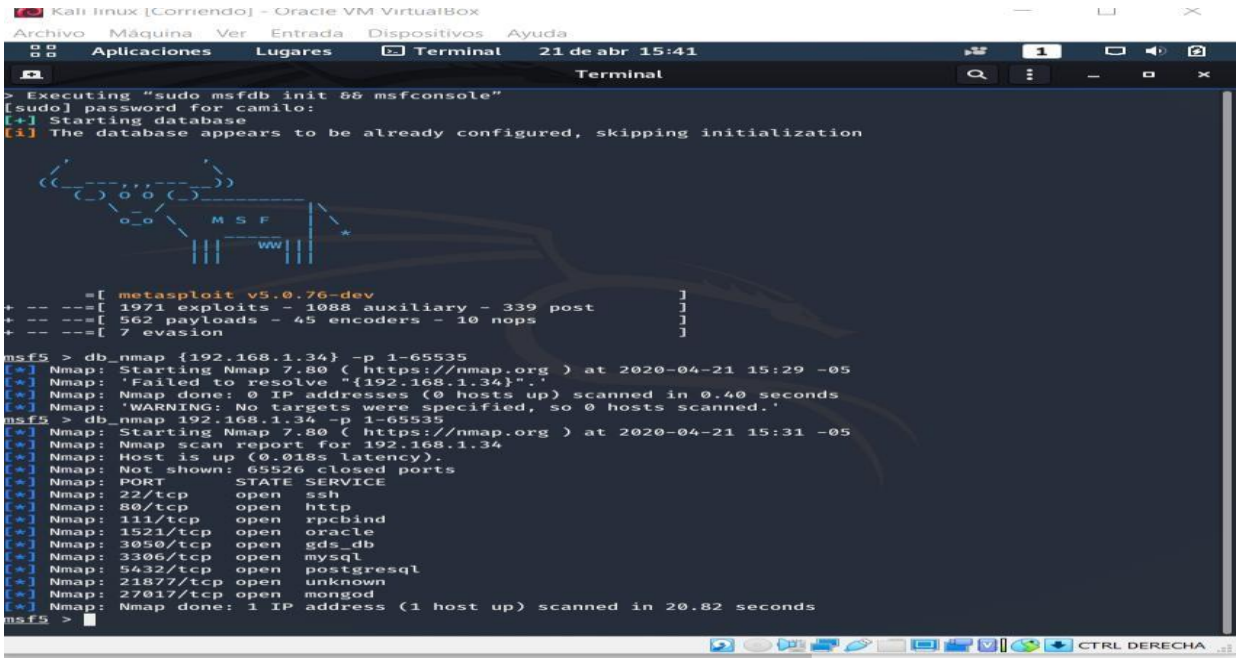
Figura 14. Muestra de la herramienta en funcionamiento de herramienta Nmap



Fuente: Elaboración propia.

5.3.2. ***Metasploit framework***. Mediante el funcionamiento de esta herramienta versión 6.0.24; se identificó la siguiente información como lo muestra la Figura 15, que *TCP* tiene abierto la base de datos y así mismo, todos los motores de base de datos que están instalados en el servidor.

Figura 15. Funcionamiento herramienta Metasploit framework



Fuente: Elaboración propia.

Se puede considerar como una vulnerabilidad el estado *open* del puerto 3306 *Mysql,* teniendo en cuenta que, si dentro de la empresa no se cuenta con las restricciones adecuadas dentro de los firewalls o no se cuenta con los mismos este puerto ofrece la conectividad de forma remota hacia la base de datos, facilitando de esta forma, el ingreso del personal externo y la oportunidad de realizar un ataque de inyección *Sql* si la aplicación no cuenta con las debidas validaciones.

5.3.3. ***Nikto.*** Mediante el funcionamiento de esta herramienta versión 2.1.6, se identificó información detallada del servidor donde se encuentra alojada la base de datos.

En la figura 16, se observa que tipo de servidor en este caso servidor apache, la *phpinfo* especifica por *Query* y además se evidencia 1 *host.* Aunque esta herramienta está enfocada a la búsqueda de vulnerabilidades en aplicaciones web, se puede destacar que

sirve de apoyo para enumerar versiones de software y a partir de esto, ubicar un posible *exploit* o ataque especifico.

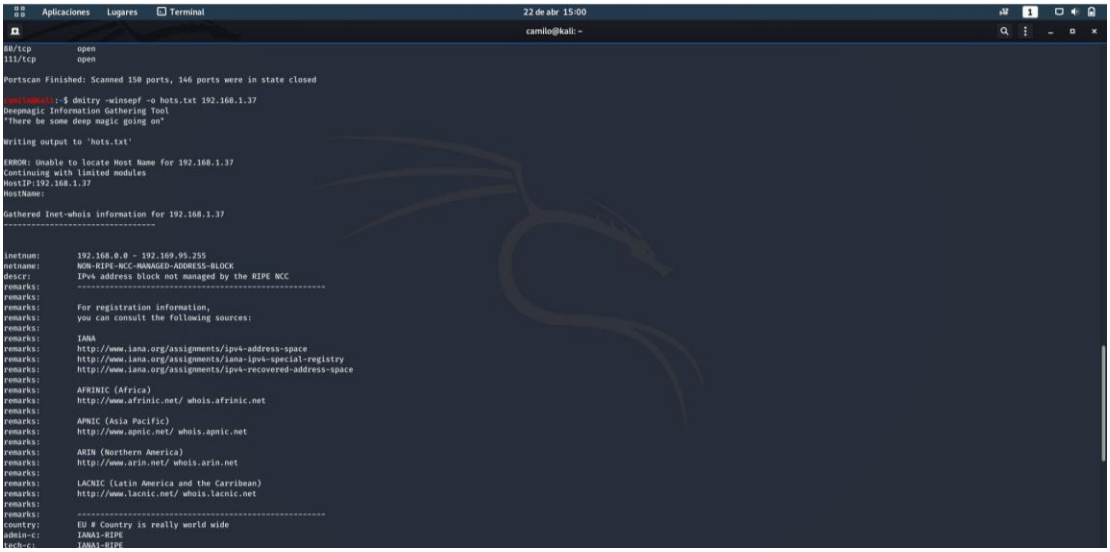
Figura 16. Funcionamiento de herramienta Nikto

Fuente: Elaboración propia.

5.3.4. *DMitry*. Mediante el funcionamiento de esta herramienta versión 2.4.1, se identifica información del *Host* y la *IP*, que se va a atacar, se evidencia subdominios, direcciones de correo electrónico, información de la actividad de la base de datos y puertos *TCP* que están abiertos.

En la figura 17, se puede identificar la siguiente información, puertos 22 – 88 y 111 /*TCP* en estado abierto, fecha de creación de la base de datos y su modificación. Adicional a ello, se puede observar la versión de *Query Service* la cual es la 1.97.

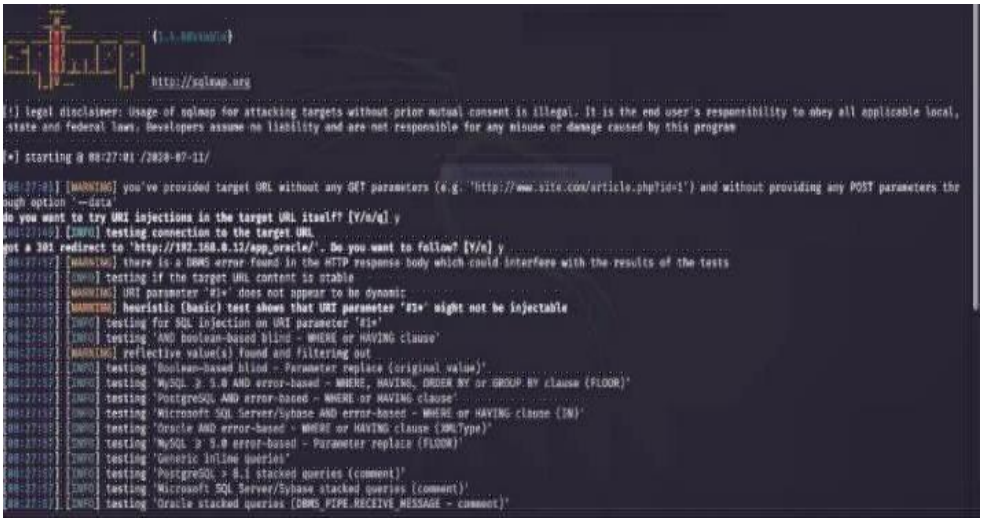
Figura 17. Funcionamiento de herramienta *DMitry*



Fuente: Elaboración propia.

5.3.5. *Sqlmap*. Mediante el funcionamiento de esta herramienta versión 7.92, se identificó vulnerabilidades de la base de datos Mysql, relacionadas con el reconocimiento automático de formatos hash de contraseñas, ofreciendo la facilidad de ser utilizada y/o explotada, para realizar un ataque basado en diccionario o ataque de fuerza bruta.

Figura 18. Funcionamiento de herramienta *Sqlmap*



Fuente: Elaboración propia.

5.4. RECOMENDAR ALTERNATIVAS DE SOLUCIÓN PARA MITIGAR VULNERABILIDADES EN LAS BASES DE DATOS RELACIONALES

Según *WHITEPAPER*, “los siguientes consejos se deben tener en cuenta para proteger una base de datos de posibles vulnerabilidades” [28].

5.4.1. Recomendaciones para proteger las bases de datos

- **Identificar su sensibilidad**

A continuación, se plasman algunos consejos a considerar para proteger las bases de datos de posibles vulnerabilidades. Hay que tener en cuenta que, el desconocimiento de estas vulnerabilidades, no los exime de la responsabilidad. Esto significa que, es importante conocer la seguridad informática de la empresa y el sistema de base de datos saber cómo funciona y de esta forma, mejorar la seguridad si es necesario. Para hacer esto, se pueden usar herramientas de detección de vulnerabilidades, con el fin de ayudar a encontrar vulnerabilidades que podrían ser atacadas.

La recomendación más indicada para identificar la sensibilidad, se encuentra ligada a realizar un estudio o autoanálisis, con el fin de identificar posibles vulnerabilidades y/o amenazas. Y de esta forma, evitar posibles ataques, buscando salvaguardar la información de las bases de datos. Se deben implementar estrategias o métodos adecuados de seguridad de la información, algo muy clave; es tener una persona experta en seguridad informática y con el profesionalismo apropiado dentro de la empresa.

Si bien es cierto, el realizar un autoanálisis, no garantiza no ser víctima de un ataque cibernético, pero sin embargo de una u otra forma, alerta y sensibiliza a las personas encargadas de la seguridad de la información dentro de la empresa.

- **Evaluación de las vulnerabilidades y la configuración**

Evaluar la configuración de la base de datos para descartar posibles vulnerabilidades de seguridad. Esto incluye comprobar el método de instalación y el sistema operativo. Por ejemplo, revisar los permisos de diferentes grupos de usuarios, en relación a modificar, leer y escribir acciones en la base de datos.

[28] TELEFÓNICA. *Whitepaper:* Bases de datos y sus vulnerabilidades más comunes. [En linea]. *Acens*, 2013., p. 4. Disponible en: https://www.acens.com/wp-content/images/2015/03/vulnerabilidades-bbdd-wp-acens.pdf

Al momento de identificar posibles vulnerabilidades dentro de las empresas, lo más recomendado es contrarrestarlas de inmediato; para ello se debe elaborar un plan de respuesta a incidentes el cual ayuda a las organizaciones a prepararse, detectar, responder y recuperarse de los incidentes de seguridad informática. Este plan debe ser elaborado por personas con el conocimiento necesario en temas de ciberseguridad.

Cuando se habla de configuración, en marca aspectos de instalación, ajuste y adecuación del entorno en sistema donde se encuentra alojada la base de datos relacional, algo tan sencillo, como otorgar permisos de configuración inadecuados a usuarios que no cumplan con el rol de administradores, puede ser fatal; teniendo en cuenta que estos podrían filtrar, modificar o extraer información o incluso borrarla.

Los usuarios administradores deben pasar por un estudio de seguridad personal, el cual se convierte en un filtro, para identificar personas que no cumplan con parámetros de confianza necesarios para el manejo de información sensible dentro de la empresa. Teniendo en cuenta que, el ser humano es un ser racional, sensible al factor de dinero; esto genera una vulnerabilidad que puede ser utilizada por personas ajenas a la empresa, quienes buscan penetrar al personal y obtener información sensible de la empresa.

- **Auditar**

Una vez que se haya creado una instalación, que se crea que es completamente segura, se debe realizar una verificación para asegurar de que no se haya desviado de su objetivo. Por ejemplo, se puede configurar un tipo de alerta para señalar posibles cambios en esa configuración.

Dentro de la empresa se debe realizar inspecciones internas a los procesos relacionados con las bases de datos, así como, generar el debido informe por cada uno de los inspectores, este tipo de auditoría interna se debe realizar de forma permanente, buscando mejorar los procesos internos y si es, el caso tomar acción de forma inmediata.

Otra forma de realizar una evaluación a los procesos internos dentro de la empresa, es invirtiendo en inspecciones por parte de personal externo, especialistas en temas de seguridad de la información y ciberseguridad; esto ayudará a identificar o desvirtuar posibles fallas de seguridad que se podrían convertir en amenazas fatales. Estas inspecciones se recomiendan efectuar de forma semestral.

Posterior a la realización de las auditorías, se debe generar por parte de control interno de la empresa un plan de mejoramiento, en el cual se deben plasmar los hallazgos encontrados, así como la línea de tiempo para subsanar dichas novedades y las acciones a realizar buscando contrarrestar las amenazas que se puedan presentar.

- **Monitorizar toda acción relacionada con la base de datos**

Al monitorear la actividad que ocurre en la base de datos, se puede obtener pistas si la base de datos está siendo mal utilizada o se pueden detectar intrusos.

Cuando se habla de monitoreo, se debe relacionar directamente con una verificación constante a todos los procesos realizados en la base de datos y en el entorno donde se encuentra alojada.

Se debe mantener un monitoreo de red en tiempo real con herramientas como wireshark y una auditoria a las bases de datos dentro de la empresa, mediante software de detección de vulnerabilidades a bases de datos relacionales como *sqlmap, DMitry* o *Imperva Analytics*. Y además de esto, emplear herramientas de prueba de intrusión como *Metasploit framework,* la cual podría generar alertas tempranas y evitar que la base de datos sea víctima de ataques cibernéticos.

Si es identificado algún tipo de acción sospechosa, dentro del monitoreo, se debe tomar acción de inmediato, teniendo en cuenta la orientación descrita en el plan de respuesta a incidentes. Así mismo, se debe contar con un plan de Backup (respaldo de información), en el cual se debe albergar la información importante en su totalidad (configuraciones, bases de datos, información del sistema), se recomienda realizar este plan de forma mensual, esto con la finalidad de evitar ser víctima de perdida, secuestro o alteración de la información.

- **Control de acceso y gestión de derechos**

No todos los datos son igual de significativos y no todos los usuarios son implantados igual. Para garantizar la integridad de la información, es necesario establecer una jerarquía que permita a cada tipo de usuario realizar solo las acciones permitidas en la base de datos. Para datos sensibles como contraseñas de todo tipo, se recomienda utilizar algún tipo de cifrado de datos para que la información no sea fácilmente legible a simple vista.

El control de acceso está ligado a proteger el ingreso a las bases de datos, teniendo en cuenta la selección de perfiles adecuados para el manejo de la información o administración de todo el sistema. *(No todos los usuarios pueden tener los mismos permisos o privilegios dentro de una empresa).* Se debe tener en cuenta, los tipos de usuarios de bases de datos como lo muestra la Figura 19.

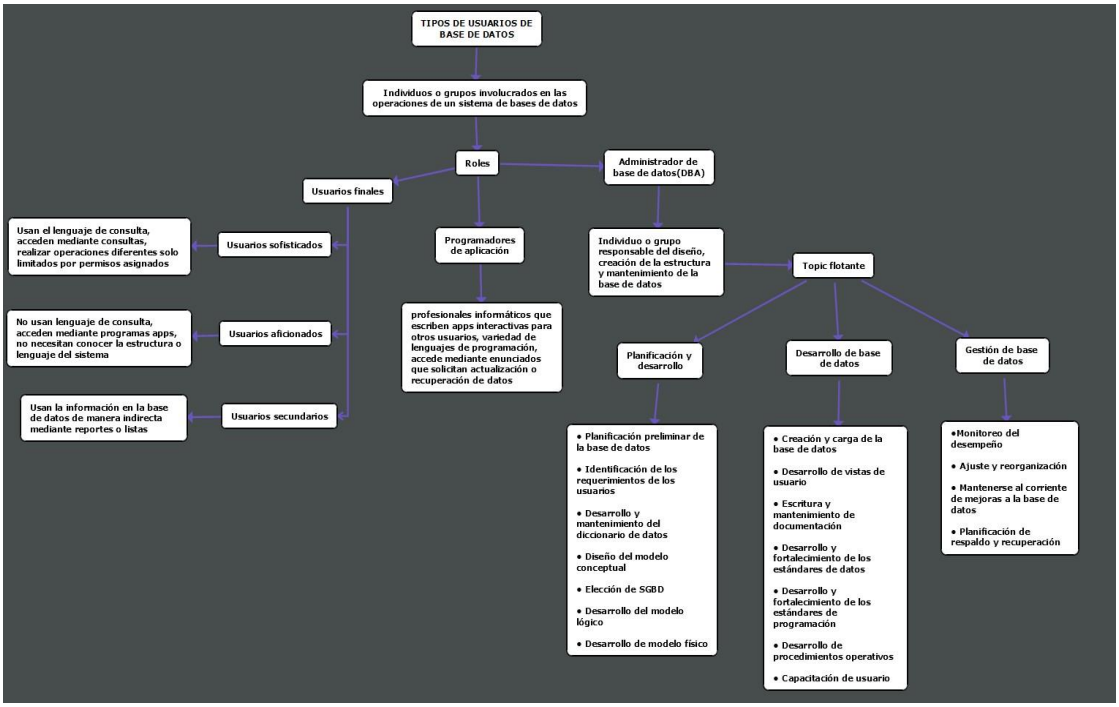
Al momento de crear algún tipo de contraseña, se debe tener en cuenta las siguientes recomendaciones:

- Longitud de contraseña: 8 caracteres
- Caracteres numéricos: mínimo 2

- Caracteres de símbolos: mínimo 1 especial
- Letras mayúsculas: como mínimo 2
- Letras minúsculas: mínimo 1

Una recomendación, es la de utilizar estrategias de cifrado o alternativas como la estenografía, con el fin de proteger la integridad, disponibilidad y confidencialidad de la información. Cuando se requiera salvaguardar algún tipo de información del alto valor estratégico, se necesita un alto grado de confidencialidad, para estos casos las contraseñas solo deben ser manejadas por la gerencia de la empresa.

Figura 19. Tipos de usuarios de bases de datos



Fuente: IMBAQUINGO TITUMAITA, Mayra Natalia. Tipos de usuarios de base de datos. [En línea]. Quito, 2021., Disponible en: https://www.mindomo.com/es/mindmap/tipos-de-usuarios-de-base-de-datos-e1b46a4ee3567d51c35b0505daa5d095

5.4.2. Alternativas de solución para vulnerabilidades en bases de datos relacionales

- **CVE-2018-1058**

Descripción. Se ha encontrado un error en la forma en la que *Postgresql,* permite que un usuario modifique el comportamiento de una consulta para otros usuarios. Un ataque

con una cuenta de usuario podría emplear este error para ejecutar código con permisos de ‘superusuario’ en la base de datos.

Impacto. El impacto que se puede generar, afecta parcialmente a la integridad, confidencialidad y disponibilidad del sistema.

Descripción del método de identificación de la vulnerabilidad. Esta vulnerabilidad se puede observar a través de red en versiones 9.3 hasta la 10, la fecha de creación de la entrada puede reflejar cuándo se asigna o reserva el ID de CVE y no indica necesariamente cuándo se descubre, se comparte con el proveedor afectado, se divulga públicamente o se actualiza esta vulnerabilidad en CVE.

Modo de subsanar la vulnerabilidad. Para resolver esta vulnerabilidad y evitar una posible explotación, se debe ejecutar en la terminal: *REVOKE CRETE ONS CHEMA public FROM PUBLIC*, con ello se evita que los superusuarios puedan crear objetos en el esquema público.

- **CVE-2019-10164**

Descripción. Un usuario autenticado podría llevar a cabo un desbordamiento de búfer cambiando su propia contraseña a través de un valor específicamente diseñado. Así mismo, tendría la capacidad de bloquear el servidor *PostgreSQL y* ejecutar código arbitrario en la condición de la cuenta del sistema de gestión de base de datos *PostgreSQL.*

Impacto. El impacto que puede generar es bloquear el usuario con el proceso de autenticación *SCRAM.*

Descripción del método de identificación de la vulnerabilidad. Esta vulnerabilidad requiere de una única autenticación y un ataque exitoso tendría un impacto completo, esto se detectó en el sistema de gestión de bases de datos *PostgreSQL*. La ejecución de código arbitrario se considera una vulnerabilidad de seguridad significativa; ya que hace, que el sistema ejecute código no confiable, debido a esto, es necesario solucionar el problema de la versión programada regularmente. La razón principal por la que CVE-2019-10164 afecta a *PostgreSQL* es la vulnerabilidad de cómo se almacena las contraseñas para la autenticación *Scram.*

Modo de subsanar la vulnerabilidad. Para resolver esta vulnerabilidad se deben aplicar últimos parches de seguridad, con el fin de evitar la exposición a ataques externos y la toma de control de los sistemas informáticos. Por otro lado, todas las actualizaciones de *PostgreSQL* son acumulativas. Por tanto, los usuarios no están obligados a volcar y volver a cargar la base de datos o usar *pg_upgrade;* para aplicarlos simplemente, el fabricante indica que se deberá cerrar *PostgreSQL* y actualizar los binarios.

- **CVE-2013-1903**

Descripción. PostgreSQL, 9.2.x antes 9.2.4, 9.1.x antes 9.1.9, 9.0.x antes 9.0.13, 8.4.x antes 8.4.17 y 8.3.x antes 8.3.23 genera archivos temporales inseguros con nombres de archivo predecibles, que tiene vectores de impacto y ataque no especificados relacionados con "instaladores gráficos para Linux y Mac *OS X*".

Impacto. El impacto que puede generar en un cierre total del recurso afectado. Este ataque puede dejar el recurso totalmente disponible, el nivel de complejidad de acceso es bajo y la autenticación no es requerida para poder ejecutar esta vulnerabilidad.

Descripción del método de identificación de la vulnerabilidad. Esta vulnerabilidad requiere de una divulgación total de información, lo que termina en la revelación de todos los archivos del sistema o base de datos.

Modo de subsanar la vulnerabilidad. Para resolver esta vulnerabilidad se deben aplicar los últimos parches de actualizaciones y paquetes del sistema disponibles.

- **CVE-2013-1899**

Descripción. Vulnerabilidad de inyección de argumentos en *PostgreSQL 9.2.*

Impacto. El impacto que puede generar es permitir a los ataques causar una denegación de servicio dentro del sistema (corrupción de archivos).

Descripción del método de identificación de la vulnerabilidad. Esta vulnerabilidad requiere de autenticaciones remotas y modificación de ajustes de configuración o ejecución de código arbitrario, por medio de conexión empleando un nombre de base de datos que comienza con un *-*.

Modo de subsanar la vulnerabilidad. Para resolver esta vulnerabilidad se debe aplicar la siguiente sentencia dentro de la terminal, *Postgres-upgrade-9_0_13.*

- **Vulnerabilidad puerto 3306**

Descripción. Conexión *TCP* por defecto abierta, es muy común dentro de la base de datos *Mysql*, que este tipo de puerto por defecto se encuentre abierto; facilitando la conexión en especial remota por parte de los atacantes.
A través de esta vulnerabilidad se podrá acceder a la base de datos y realizar alteraciones, extraer información de suma importancia o incluso instalar algún tipo de código malicioso.

Impacto. El impacto que se puede generar es de nivel socioeconómico, teniendo en cuenta que se puede realizar alteraciones, extracciones de información importante. Y así

mismo, realizar el secuestro extorsivo de información, afectando económicamente a la empresa u organización.

Descripción del método de identificación de la vulnerabilidad. Esta vulnerabilidad se podría observar, al momento de incorporar credenciales por defecto sin autenticación adecuada o en algunos casos esta opción deshabilitada. La forma adecuada de identificar esta vulnerabilidad es a través de herramientas de análisis de vulnerabilidades.

Modo de subsanar la vulnerabilidad. Para resolver esta vulnerabilidad y evitar una posible explotación de algún tipo de atacante, por medio de un ataque inyección *Sql*. Se sugiere que las bases de datos deben estar en un segmento de red o *VLAN* diferente a los usuarios. De igual forma, ofrecer mayor control mediante una herramienta de seguridad que podría ser un *Firewall*, solo se deberá permitir la comunicación desde el segmento de servidores de aplicación hacia la base de datos, no desde usuarios hacia base de datos y además de esto, tener en cuenta lo siguiente:

- Implementación de software de identificación y remediación de vulnerabilidades.
- Realizar el cierre del puerto a través de la sentencia adecuada. (Solo si es necesario).
- Realizar restricciones de seguridad en el firewall.
- No aceptar credenciales de confianza.

- **Vulnerabilidad puerto 5432 *postgresql***

Descripción. El puerto 5432 abierto (*Open)* Utilizado por *Postgresql*, es un sistema de gestión de bases de datos relacionales de código abierto orientado a objetos. Que utiliza este puerto para las conexiones.

Por medio de este puerto los atacantes pueden lograr el acceso a las bases de datos, mediante ataques de fuerza bruta, con el fin de obtener las credenciales de acceso y de esta forma, realizar algún tipo de ataque.

Impacto: El impacto que se puede generar es de nivel socioeconómico, teniendo en cuenta que se puede realizar alteraciones, extracciones y daño de información importante. Así mismo, realizar el secuestro extorsivo de información, afectando económicamente a la empresa u organización.

Descripción del método de identificación de la vulnerabilidad: Esta vulnerabilidad se podría observar al momento de incorporar credenciales por defecto, sin autenticación adecuada o en algunos casos esta opción deshabilitada. La forma adecuada de identificar esta vulnerabilidad es a través de herramientas de análisis de vulnerabilidades.

Modo de subsanar la vulnerabilidad: Para resolver esta vulnerabilidad y evitar una posible explotación de algún tipo de atacante, se debe realizar las instalaciones de parches y configuraciones adecuadas, si es necesario el cierre manual de este puerto y además tener en cuenta lo siguiente:

- Implementación de software de identificación y remediación de vulnerabilidades.
- Se recomienda que la base de datos debe estar en un segmento de red o *VLAN* diferente al de los usuarios.
- Realizar restricciones en los firewalls.
- No aceptar credenciales de confianza.

- **Vulnerabilidades credenciales por defecto**

Descripción. Al momento de realizar la implementación de una base de datos, se generan una serie de parámetros por defecto, en muchas ocasiones los usuarios administradores no verifican rigurosamente este tipo de configuraciones. Esto se convierte en una vulnerabilidad para la estructura interna de la base de datos, siendo muy atractivo para un atacante, que conoce por donde obtener algún tipo de acceso mediante las configuraciones por defecto.

Impacto. El impacto que se puede generar es de nivel socio-económico, teniendo en cuenta que se puede realizar alteraciones, extracciones de información, afectando económicamente a la empresa u organización y dañando la imagen institucional de la misma.

Descripción del método de identificación de la vulnerabilidad. Esta vulnerabilidad se podría observar al momento de incorporar configuraciones por defecto. La forma adecuada de identificar esta vulnerabilidad es a través de herramientas de análisis de vulnerabilidades y un control riguroso por parte del usuario administrador.

Modo de subsanar la vulnerabilidad. Para resolver esta vulnerabilidad y evitar una posible explotación de algún tipo de atacante, con el fin de ingresar y realizar de un ataque. Se debe realizar una evaluación de las configuraciones por defecto, más sensibles en el funcionamiento de las bases de datos, además tener en cuenta lo siguiente:

- Implementación de software de identificación y remediación de vulnerabilidades.
- Análisis manual de configuración por parte del administrador.
- Realizar restricciones en los firewalls.
- No aceptar credenciales de confianza.
- Realizar actualizaciones a las bases de datos.

6. CONCLUSIONES

- A partir de la revisión documental asociada a vulnerabilidades y riesgos en las bases de datos relacionales, se identificaron los principales riesgos; siendo la poca desactualización de las mismas y las credenciales por defecto, los problemas más comunes en la actualidad. Estos tipos de vulnerabilidades ofrecen a los atacantes informáticos la facilidad de ejecutar algún tipo de código no deseado o realizar un ataque inyección *Sql*, él cual es el más utilizado por los atacantes para extraer, modificar o en ocasiones secuestrar la información a través de *ransomware*. Es por esto, que las organizaciones deben ser conscientes, de la necesidad de proteger la información almacenada en las bases de datos y de esta forma, evitar la explotación de este tipo de vulnerabilidades.

- Una vez realizado el análisis de herramientas de detección de vulnerabilidades en bases de datos relacionales, se identificó a la herramienta *sqlmap* como la más popular para realizar test de penetración. Además de esto, esta herramienta ofrece la ventaja de ser utilizada en código abierto, automatiza el proceso de detectar y explotar las amenazas de inyección *SQL* y finalmente soporta un alto rango de motores de bases de datos en especial *Mysql*. Es por esto, que se recomienda utilizar para el escaneo en bases de datos relacionales.

- Posterior al análisis de las herramientas, se efectuó la verificación del funcionamiento de las mismas, en relación a la detección de vulnerabilidades en bases de datos relacionales: sqlmap, *nmap*, *nikto* y *DMitry*; estas herramientas trabajan con código abierto, pero, sin embargo, por sus características especiales de *test* de penetración la más recomendable, es la herramienta *Sqlmap*. Por otra parte, durante el proceso de funcionamiento de estas herramientas; entre las vulnerabilidades más detectadas, se encuentran los puertos de conexión abiertos (*5432- TCP)* y el *(3306 Mysql*). Él estado abierto de estos puertos, generan que las bases de datos, sean víctimas de diferentes ataques, entre ellos inyección *SQL* y de esta forma, afectan directa o indirectamente sobre la integridad, disponibilidad y confidencialidad de la información.

- Después del proceso de análisis y validación de las vulnerabilidades en las bases de datos relacionales; se realizaron acciones de prevención, como es la implementación de *software de detección,* estas herramientas permitieron la identificación y remediación oportuna de vulnerabilidades en la base de datos *Mysql*; cabe resaltar que, para lograr este tipo de acción, se debe concientizar a las organizaciones, con respecto a la necesidad de proteger el sistema, en razón a evitar una posible explotación de dichas amenazas, mediante inyección *Sql* y

finalmente se concluyó que la mejor forma de contrarrestar la desactualización en las bases de datos y la utilización de credenciales por defecto, es con la contratación de personal idóneo en temas de seguridad informática (Oficiales de seguridad); los cuales mediante, un plan de trabajo podrían contrarrestar todo tipo de vulnerabilidad en el sistema.

7. RECOMENDACIONES

- Entre las recomendaciones más relevantes se encuentra el proceso de concientización. Tiene como objetivo inspirar a las personas a informarse sobre los riesgos y amenazas a los que se enfrenta la información actualmente. Así mismo, se deben establecer una serie de lineamiento parámetros o normas con el fin, de que estas vulnerabilidades no se materialicen; teniendo en cuenta que, las personas son quienes crean, utilizan y custodian la información. Esta concientización debe estar apoyada de un plan de capacitación sobre temas de Ciberseguridad.

- Implementar dentro de las organizaciones herramientas, que realicen un continuo escaneo de *host* y de redes, esto con el fin, de evidenciar vulnerabilidades en tiempo real; como lo son puertos de comunicación en estado abierto o credenciales por defecto. De esta forma, se podrá minimizar ataques cibernéticos como inyección *Sql*, los cuales podrían afectar directa o indirectamente a las bases de datos relacionales de la empresa. Se recomienda para una mejor efectividad, prever que las herramientas estén licenciadas y debidamente actualizadas o en su defecto utilizar software de código abierto.

- Establecer políticas de seguridad informática dentro de las organizaciones y en estas plasmar todas las normas, conductas y directrices, que faciliten el control en la seguridad de la información, además de esto, estas políticas deben ser difundidas a todo nivel y se debe dejar constancia de la difusión de las mismas. Una política muy resaltante estaría directamente relacionada con la pérdida de información; se debe generar un plan de *Backup* de forma mensual, abarcando información sensible de todas las áreas de la empresa; esto con el fin de evitar la pérdida total de la información o en ocasiones secuestro de datos sensibles.

- Generar actas de manejo de información y realizar estudios de seguridad al personal que tenga acceso a las bases de datos. Teniendo en cuenta la sensibilidad de la información, se debe contar con personal altamente confiable; es por esto que, al momento de contratar personal, se debe realizar previamente un estudio de seguridad personal. Además de esto, este personal debe contar con un acta, donde se comprometa a salvaguardar la información y cumplir con las políticas de seguridad informática establecidas por la empresa.

BIBLIOGRAFÍA

AGUIRRE TOBAR, Ricardo Andrés, et al. Estudio para la implementación del sistema de gestión de seguridad de la información para la secretaria de educación departamental de Nariño basado en la norma ISO/IEC 27001. 2015.

ÁLVAREZ ISAZA, Zaida Milena. *ISO/IEC 2700: 2013-sistemas de gestión de seguridad de la información*. 2016. Tesis de Licenciatura. Universidad Piloto de Colombia.

AMAIQUEMA VERA, Julio Francisco; SÁNCHEZ PINCAY, Freddy Idario. Estudio de seguridad en las aplicaciones web desarrolladas por un servicio OUTSOURCING. 2018. Tesis Doctoral. Universidad de Guayaquil. Facultad de Ciencias Matemáticas y Físicas. Carrera de Ingeniería en Sistemas Computacionales.

AMAYA HERNÁNDEZ, Janeth Rocío. *El sistema financiero y la seguridad informática*. 2014. Tesis de Licenciatura. Universidad Piloto de Colombia.

AMUTIO GÓMEZ, M.; CANDAU, J. Metodología de Análisis y Gestión de Riesgos de los Sistemas de Información. 2012.

ARÉVALO-CORDOVILLA, Felipe Emiliano, et al. Importancia de la seguridad de los sistemas de información frente el abuso, error y hurto de información. *Dominio de las Ciencias*, 2020, vol. 6, no 2, p. 835-846.

ARIAS VALENCIA, M. M. (abril de 1999). La triangulación metodológica: sus principios, alcances y limitaciones. Obtenido de Udea.com: https://www.uv.mx/mie/files/2012/10/Triangulacionmetodologica.pdf

AVILA, Bibiana. La triangulación, una técnica de investigación. *Rescatado de http://triangulacion-tecnicadeinvest. Blogspot. Com/2010/10/la-triangulacion-unatecnica-de. html*, 2010.

AVILÉS, Gabriel Gallardo. Seguridad en bases de datos y aplicaciones web. IT Campus Academy, 2015.

ALEGRE RAMOS, María Del Pilar; GARCÍA-CERVIGÓN HURTADO, Alfonso. Seguridad informática. Editorial Paraninfo, 2011

BOGOTÁ, S. J. Proyecto de Acuerdo 8 de 2009 Concejo de Bogotá DC. recuperado de: http://www. alcaldiabogota. gov. co/sisjur/normas. *Norma1. jsp*, 2018.

BOCANEGRA QUINTERO, Yamilet, et al. Análisis y gestión de riesgos de los sistemas de información de la alcaldía municipal de Tuluá aplicando la metodología MAGERIT. 2015.

BOTERO VEGA, David Humberto, et al. Diseño del Sistema de Gestión de Seguridad Informática y de la Información (SGSI) para la Empresa Belisario Ltda. de la ciudad de Bogotá DC.

BORTNIK, Sebastián. Pruebas de penetración para principiantes: 5 herramientas para empezar. 2013.

BUENDÍA, José Fabián Roa. Seguridad informática. McGraw-Hill España, 2013.

CAMARGO RAMÍREZ, Juan David, et al. Diseño de un sistema de Gestión de la Seguridad de la Información (SGSI) en el área tecnológica de la Comisión Nacional del Servicio Civil-CNSC basado en la norma ISO27000 e iso27001. 2017.

Canal En VIVO - Universidad EAFIT. (2018). Administración de Riesgos Cibernéticos: Nuevos desafíos relacionados con la dependencia tecnológica [Video]. Recuperado de: https://www.youtube.com/watch?v=bb8gzgkJutk.

CANO, Jeimy J. Inseguridad informática: un concepto dual en seguridad informática. *Revista de Ingeniería*, 2004, no 19, p. 40-44.

CASTRO QUINDE, Carlos Oswaldo. Elaboración de un Sistema de Gestión de Seguridad de la Información (Sgsi) para la Empresa Radical Cia. Ltda. En la Ciudad de Quito para el año 2014. 2014. Tesis de Maestría. Quito: Universidad de las Américas, 2014.

CASTRO, Martha Irene Romero, et al. *Introducción a la seguridad informática y el análisis de vulnerabilidades*. 3Ciencias, 2018.

CÁRDENAS-SOLANO, Leidy-Johanna; MARTÍNEZ-ARDILA, Hugo; BECERRA-ARDILA, Luis-Eduardo. Gestión de seguridad de la información: revisión bibliográfica. *El profesional de la información (EPI)*, 2016, vol. 25, no 6, p. 931-948.

CARDONA TOVAR, Lorena Patricia, et al. Desarrollo de un Marco de Trabajo para la Gestión del SGSI en PYMES Desarrolladoras de Software en Bogotá Basado en la Metodología MGSM-PYME.

CHEN, Hsinchun; WANG, Fei-Yue; ZENG, Daniel. Intelligence and security informatics for homeland security: information, communication, and transportation. IEEE Transactions on Intelligent Transportation Systems, 2004, vol. 5, no 4, p. 329-341.

COLLMANN, Jeff; COOPER, Ted. Breaching the security of the Kaiser Permanente Internet patient portal: the organizational foundations of information security. Journal of the American Medical Informatics Association, 2007, vol. 14, no 2, p. 239-243.

COMUNES, VULNERABILIDADES Y EXPOSICIONES. 2019. CVE Mitre. *CVE Mitre.* [En línea] 29 de 10 de 2019. [Citado el: 16 de Junio de 2020.] https://nvd.nist.gov/vuln/detail/CVE-2019-10749.

Congreso De La República. Presidencia de la República de Colombia. Ley 1273 (5, enero, 2009). De la protección de la información y de los datos. Bogotá D.C: El congreso, 2009. p. 1-3. http://www.secretariasenado.gov.co/senado/basedoc/ley_1273_2009.html

DE RISQUES, Méthode Harmonisée d'Analyse. Club de la Sécurité de l'Information Français. 2010.

ESPINEL, Rafael Antonio. Proyecto de investigación estrategia para implementar un sistema de gestión de la seguridad de la información basada en la norma iso 27001 en el área de ti para la empresa market mix.

EPPEL, Noam. Security absurdity: the complete, unquestionable, and total failure of information security. 2005.

FIGUEROA-SUÁREZ, Juan A., et al. La seguridad informática y la seguridad de la información. Polo del conocimiento, 2018, vol. 2, no 12, p. 145-155.

FONTECHA ZABALETA, Joan Alexander. Python en la seguridad informática. 2017.

GEANA, Ionut Catalin. Security Solutions for Informatics Systems. Int'l J. Info. Sec. & Cybercrime, 2012, vol. 1, p. 17.

GORDON, L. A., Loeb, M. P., Lucyshyn, W., & Richardson, R. (2006). 2006 CSI/FBI computer crime and security survey. Computer Security Journal, 22(3), 1.

JARAMILLO, Carlos Arturo Blandón; SEPÚLVEDA, Alejandra María Benavides. Modelo sistema de gestión de seguridad de la información para instituciones educativas de nivel básico. *Scientia et technica*, 2018, vol. 23, no 1, p. 85-92.

LÓPEZ, Purificación Aguilera. Seguridad informática. Editex, 2010

LÓPEZ SIERRA, Pablo Andrés; VARGAS VARGAS, Jenny Alexandra. *Seguridad informática, importancia desde la perspectiva del recurso humano*. 2014. Tesis de Licenciatura. Universidad Piloto de Colombia.

MARRERO TRAVIESO, Yran. La Criptografía como elemento de la seguridad informática. *Acimed*, 2003, vol. 11, no 6, p. 0-0.

MARTÍNEZ, Cuauhtémoc Vélez. Seguridad informática. *Gaceta Instituto de Ingeniería, UNAM*, 2017, vol. 1, no 98, p. 21-21.

MARIÑO OBREGÓN, Alipio. Factores inhibidores en la implementación de sistemas de gestión de la seguridad de la información basado en la NTP-ISO/IEC 17799 en la administración pública. 2010.

MIHAI, Ioan-Cosmin; CIUCHI, Costel. Security Implementation on Informatics Systems. *Pub. Sec. Stud.*, 2012, vol. 1, p. 60.

MONROY SALAZAR, Juan Carlos, et al. Metodología para hacking ético en bases de datos.

MORENO ABAUNZA, César Alberto. Estudio de políticas de seguridad para la elaboración de software.

MUÑOZ MARTÍN, Manuel. Guía de implantación de un SGSI basado en la norma UNE-ISO/IEC 27001. 2015.

NOVOA, Helena Alemán; BARRERA, Claudia Rodríguez. Metodologías para el análisis de riesgos en los sgsi. *Publicaciones e Investigación*, 2015, vol. 9, p. 73-86.

NGUYEN, Minh X.; YUAN, Xiaoru; CHEN, Baoquan. Geometry completion and detail generation by texture synthesis. *The Visual Computer*, 2005, vol. 21, no 8, p. 669-678.

PIEDRAHITA VILLARRAGA, Elkin. "Análisis comparativo de un Firewall de aplicaciones web comerciales y un Open Source frente al top 10 de Owasp.". Repositorio Institucional UNAD. Universidad Nacional Abierta y a Distancia. 2016.

PINZÓN Liliana, TALERO MihdíBadí, BOHADA John. 2013. *INTRUSION TEST AND OPEN SOURCE METHODOLOGIES. Grupo de investigación MUISCA. 16 de Octubre de 2013, pág. 38.*

PINZÓN PARADA, Iraldo. *Gestión del riesgo en Seguridad Informática*. 2014. Tesis de Licenciatura. Universidad Piloto de Colombia.

REVELO, Diego Sebastian Gordón. Análisis de estrategias de gestión de seguridad informática con base en la metodología Open Source Security Testing Methodology Manual (OSSTMM) para la intranet de una Institución de Educación Superior-*Analysis of Strategies of Computer Security Management Based on the Open Source Security Testing Manual Methodology (OSSTMM) for the Intranet of a Higher Education Institution. ReCIBE, Revista electrónica de Computación, Informática, Biomédica y Electrónica*, 2018, vol. 7, no 1, p. 1-21.

RIBÓ ARNAU, Alexandre, et al. Base de datos PostGIS para la gestión de riesgos en la microrregión “Mélida Anaya Montes” (El Salvador). 2012.

ROBIDOUX, G. SQL Server INFORMATION_SCHEMA views Tutorial. [online] Mssqltips. com. 2017.

RUIZ GONZÁLEZ, Pedro Alberto, et al. *Seguridad de aplicaciones web basadas en las tecnologías Node. js y MongoDB: Estudio y caso de uso*. 2018. Tesis de Maestría.

SÁNCHEZ, Juan Ignacio; IGNOTO, María José. *La seguridad informática*. Instituto de la Pequeña y Mediana Empresa Industrial, 1991.

SALAZAR CATAÑO, Jose Alain, et al. Mitigar los riesgos de ataques a bases de datos Postgresql, de la familia de las versiones 9. x, en ambientes web.

SAIN, Gustavo. ¿Qué es la seguridad informática? *Pensamiento Penal*, 2018, vol. 5.

SECURITY ABSURDITY: *The complete, unquiestionable, and total failure of information security*. [En línea]. <https://revistas.uexternado.edu.co/index.php/derpen/article/view/9652006>. [Citado en 22 de Septiembre de 2019]

SOLARTE, Francisco Nicolás Solarte; ROSERO, Edgar Rodrigo Enríquez; DEL CARMEN BENAVIDES, Mirian. Metodología de análisis y evaluación de riesgos aplicados a la seguridad informática y de información bajo la norma ISO/IEC 27001. *Revista Tecnológica-ESPOL*, 2015, vol. 28, no 5.

SHILANE, P., MIN, P., KAZHDAN, M. Y FUNKHOUSER, T. (junio de 2004). El punto de referencia de la forma princeton. *En Proceedings Shape Modeling Applications, 2004. (págs. 167-178). IEEE.*

TARAZONA, T.; CESAR, H. Amenazas informáticas y seguridad de la información. *Derecho Penal y Criminología*, 2007, vol. 28, p. 137.

TEJADA, Ester Chicano. *Auditoría de seguridad informática. IFCT0109*. IC Editorial, 2015.

TURNER, Dean, et al. Symantec Internet security threat report: trends for July 2004-December 2004. Retrieved July, 2005, vol. 30, p. 2005

URBINA, Gabriel Baca. *Introducción a la seguridad informática*. Grupo editorial PATRIA, 2016.

VALENCIA, María Mercedes Arias. La triangulación metodológica: sus principios, alcances y limitaciones. *Investigación y educación en enfermería*, 2000, vol. 18, no 1, p. 13-26.

VÉLEZ, Laura Lotero; HEREDIA, Rafael Germán Hurtado. Vulnerabilidad de redes complejas y aplicaciones al transporte urbano: una revisión de la literatura. *Revista EIA*, 2014, vol. 11, no 21, p. 67-77.

VIEITES, Álvaro Gómez. *Enciclopedia de la seguridad informática*. Grupo Editorial RA-MA, 2011.

VOUTSSAS, M., et al. Preservación documental digital y seguridad informática. *Investigación bibliotecológica*, 2010, vol. 24, no 50, p. 127-155.

WHITEPAPER: BASES DE DATOS Y SUS VULNERABILIDADES MÁS COMUNES 2013. Texto del artículo-p.p. 2 – 5.

ZAMBRANO, Silvia M. Quiroz; VALENCIA, David G. Macías. Seguridad en informática: consideraciones. *Dominio de las Ciencias*, 2017, vol. 3, no 3, p. 676-688.

ZALLAS, Edgar Alberto Espinoza, et al. Seguridad informática una problemática de las organizaciones en el Sur de Sonora. *Revista de Investigación Académica Sin Frontera: División de Ciencias Económicas y Sociales*, 2017, no 25.

8. ANEXOS

Anexo. RAE

Fecha de Realización:	30/07/2021
Programa:	Especialización en Seguridad Informática
Línea de Investigación:	Cadena de formación en electrónica, telecomunicaciones y redes
Título:	RIESGOS Y VULNERABILIDADES EN LAS BASES DE DATOS RELACIONALES
Autor(es):	Pinilla Ramos Christian Alexander
Palabras Claves:	Base de datos, Informática, vulnerabilidad, Seguridad, TIC´S,
Descripción:	El presente documento relacionado con la detección de vulnerabilidades pretende dar a conocer una información sobre los procedimientos y herramientas utilizadas para la identificación de vulnerabilidades en las bases de datos relacionales; teniendo en cuenta, factores cualitativos y cuantitativos sobre resultados de investigaciones anteriores, relacionadas con la gestión del riesgo informático. Y asi mismo, recomendar posibles alternativas de solución a los posibles problemas de seguridad informática en bases de datos.

Fuentes bibliográficas destacadas:

AMUTIO GÓMEZ, M.; CANDAU, J. Metodología de Análisis y Gestión de Riesgos de los Sistemas de Información. 2012.

NOVOA, Helena Alemán; BARRERA, Claudia Rodríguez. Metodologías para el análisis de riesgos en los sgsi. *Publicaciones e Investigación*, 2015, vol. 9, p. 73-86.

PINZÓN PARADA, Iraldo. *Gestión del riesgo en Seguridad Informática*. 2014. Tesis de Licenciatura. Universidad Piloto de Colombia.

REVELO, Diego Sebastian Gordón. Análisis de estrategias de gestión de seguridad informática con base en la metodología Open Source Security Testing Methodology Manual (OSSTMM) para la intranet de una Institución de Educación Superior-*Analysis of Strategies of Computer Security Management Based on the Open Source Security Testing Manual Methodology (OSSTMM) for the Intranet of a Higher Education Institution. ReCIBE, Revista electrónica de Computación, Informática, Biomédica y Electrónica*, 2018, vol. 7, no 1, p. 1-21.

TARAZONA, T.; CESAR, H. Amenazas informáticas y seguridad de la información. *Derecho Penal y Criminología*, 2007, vol. 28, p. 137.

Contenido del documento:	Portada, sub portada, introducción, definición del problema, formulación del problema, justificación, objetivos, marco referencial, desarrollo de los objetivos, conclusiones, recomendaciones, bibliografías y anexos.
Marco Metodológico:	No aplica
Conceptos adquiridos:	Después del desarrollo del trabajo de grado

Conclusiones:	A partir de la revisión documental asociada a vulnerabilidades y riesgos en las bases de datos relacionales, se identificaron los principales riesgos; siendo la poca desactualización de las mismas y las credenciales por defecto, los problemas más comunes en la actualidad. Estos tipos de vulnerabilidades ofrecen a los atacantes informáticos la facilidad de ejecutar algún tipo de código no deseado o realizar un ataque inyección *Sql*, él cual es el más utilizado por los atacantes para extraer, modificar o en ocasiones secuestrar la información a través de *ransomware*. Es por esto, que las organizaciones deben ser conscientes, de la necesidad de proteger la información almacenada en las bases de datos y de esta forma, evitar la explotación de este tipo de vulnerabilidades. Una vez realizado el análisis de herramientas de detección de vulnerabilidades en bases de datos relacionales, se identificó a la herramienta *sqlmap* como la más popular para realizar test de penetración. Además de esto, esta herramienta ofrece la ventaja de ser utilizada en código abierto, automatiza el proceso de detectar y explotar las amenazas de inyección *SQL* y finalmente soporta un alto rango de motores de bases de datos en especial *Mysql*. Es por esto, que se recomienda utilizar para el escaneo en bases de datos relacionales. Posterior al análisis de las herramientas, se efectuó la verificación del funcionamiento de las mismas, en relación a la detección de vulnerabilidades en bases de datos relacionales: sqlmap, *nmap*, *nikto* y *DMitry*; estas herramientas trabajan con código abierto, pero, sin embargo, por sus características especiales de *test* de penetración la más recomendable, es la herramienta *Sqlmap*. Por otra parte, durante el proceso de funcionamiento de estas herramientas; entre las vulnerabilidades más detectadas, se encuentran los puertos de conexión abiertos (*5432- TCP)* y el *(3306 Mysql*).

	Él estado abierto de estos puertos, generan que las bases de datos, sean víctimas de diferentes ataques, entre ellos inyección *SQL* y de esta forma, afectan directa o indirectamente sobre la integridad, disponibilidad y confidencialidad de la información. Después del proceso de análisis y validación de las vulnerabilidades en las bases de datos relacionales; se realizaron acciones de prevención, como es la implementación de *software de detección,* estas herramientas permitieron la identificación y remediación oportuna de vulnerabilidades en la base de datos *Mysql*; cabe resaltar que, para lograr este tipo de acción, se debe concientizar a las organizaciones, con respecto a la necesidad de proteger el sistema, en razón a evitar una posible explotación de dichas amenazas, mediante inyección *Sql* y finalmente se concluyó que la mejor forma de contrarrestar la desactualización en las bases de datos y la utilización de credenciales por defecto, es con la contratación de personal idóneo en temas de seguridad informática (Oficiales de seguridad); los cuales mediante, un plan de trabajo podrían contrarrestar todo tipo de vulnerabilidad en el sistema.