ISSN: 2675-1828

ARTIGO

Ferramentas de segurança para banco de dados: focando em SQL Injection

Database security tools: focusing on SQL Injection

Guilherme Manfrim Basilio[1]

Wdson de Oliveira[2]

Resumo: *Com a evolução e aumento do armazenamento digital dos dados e informações das empresas e organizações, também aumentou o surgimento de novos tipos de ataques voltados ao banco de dados, tentando explorar as vulnerabilidades presentes, causando enormes prejuízos aos proprietários da base, e expondo a informações de inúmeros indivíduos. Este documento tem como finalidade identificar e apresentar ferramentas, como credenciais e níveis diferentes de acesso, criptografia de dados, parametrização de consultas, utilização de consultas armazenadas, backups, armazenamento seguro, dentre outras, que visam garantir a segurança de banco de dados, principalmente à utilização de SQL Injection. Sendo assim é possível identificar como objetivo a explicação do papel da segurança de bancos de dados apresentando alguns dos tipos mais comuns de ataques, e explicar os métodos e ferramentas de proteção da Injeção SQL, apresentando algumas das principais e mais utilizadas no mercado para segurança de bancos de dados.*

Palavras-Chave: *banco de dados; ciberataque; ferramentas de segurança; injeção SQL; vulnerabilidade.*

Abstract. *Together to the evolution and increase of data and information digital storage on companies and organizations, the emergence of new attacks aimed at the database types also increased, trying to exploit it vulnerabilities, causing enormous damage to the database owners, and exposing numerous individual information's. This document aims to identify and present tools, such as credentials and different levels of access, data encryption, query parameterization, use of stored queries, backups, secure storage, among others, which aim to ensure database security, especially to the use of SQL Injection. Therefore, it is possible to identify as the objective the explanation of the role of database security, presenting some of the most common types of attacks, and explaining the methods and tools for SQL Injections, and presenting some of the main and most used in the market for database security.*

Keywords: *cyber-attack; database; security tools; SQL injection; vulnerability.*

[1] Faculdade de Tecnologia de Araraquara. e-mail: guilherme.basilio@fatec.sp.gov.br

[2] Faculdade de Tecnologia de Araraquara. e-mail: wdson.oliveira01@fatec.sp.gov.br

1. Introdução

Os dados e informações de uma empresa são considerados alguns dos seus bens mais valiosos, e com o aumento da tecnologia e da facilidade de aquisição e comunicação dos dados, surgiu a necessidade de um armazenamento amplo e seguro. Acompanhado dessa necessidade crescente surgiram os Sistemas Gerenciadores de Bancos de Dados (SGBD) cuja função é manipular e gerenciar os bancos de dados. Dessa forma, os dados das empresas começaram a ser armazenados digitalmente, suprindo essas necessidades, porém, também trazendo alguns riscos como por exemplo SQL Injection, privilégios excessivos ou esquecidos, malwares, exposição de armazenamento de mídia, exploração de vulnerabilidades e configurações fracas de banco de dados, entre outros.

Um grande problema que foi "criado" com a utilização e exposição, mesmo que restrita, dos dados das instituições foram as Injeções SQL, podemos defini-la como:

> "A injeção de SQL é uma técnica para hackers executarem consultas SQL maliciosas no servidor de banco de dados. Ele pode ser executado em um aplicativo baseado na web para acessar os bancos de dados que contêm informações confidenciais. De acordo com a National Security Agency (NSA), injeção de SQL é a forma mais comumente usada por hackers."(YUNUS et al, 2018, p. 215, tradução nossa[3])

No presente documento são apresentados alguns tipos de ataques e vulnerabilidades mais comumente exploradas nos bancos de dados e as características gerais de algumas ferramentas, que devem ser estudas e caso necessário implementadas, para proteção geral para bancos de dados, com enfoque na prevenção e tratativa à Injeção SQL.

2. Bancos de Dados

Segundo a revista The Economist (2017, tradução nossa[4]) "O recurso mais valioso não é mais o petróleo, mas sim dados". Dados são bens de valores imensuráveis às empresas atualmente, pois podem armazenar informações de produtos, serviços, colaborados, clientes, dados financeiros, fiscais e muitos outros de diversos tipos e finalidades, que caso perdidos ou acessados indevidamente podem representar prejuízos e quebras legais e de contratos que podem até mesmo levar a empresa à falência.

Atualmente os bancos de dados podem ser relacionais e não relacionais. O banco de dados relacional armazena dados em tabelas, o que se tornou uma limitação em quantidades massivas de dados, causando o surgimento dos bancos não relacionais. De acordo Győrödi, Győrödi e Sotoc (2015, p. 1) podemos dividir o banco não relacional em quatro diferentes tipos: bancos de dados de valor-chave, de documentos, armazenamentos de famílias de colunas, e bancos de dados gráficos.

[3]"SQL injection is a techniquefor hackers to execute malicious SQL queries on the database server. It can be executed over a web-based application to access over the databases that contain sensitive information. According to National Security Agency (NSA), SQL injection is the most typically ways used by hackers."

[4] "The world's most valuable resource is no longer oil, but data"

3. Ciberataques

Em um meio geral, cyber ataques são tentativas realizadas por cybers criminosos a fim de garantir acesso a ele ou indisponibilidade do detentor à um objeto virtual. John Chambers, ex-CEO da Cisco, uma vez disse: "Existem dois tipos de empresas: as que foram hackeadas e as que ainda não sabem que foram hackeadas".

Segundo Caporale *et al.* (2019, tradução nossa[5]):

> "O ataque cibernético é um ataque lançado de um ou mais computadores contra outros computadores ou redes (para desativá-los ou obter acesso aos dados e gerenciá-los); compromete a segurança da informação ao afetar sua confidencialidade, integridade e disponibilidade."

Segundo o site Cisco, os meios de cyber ataques mais comuns são Malwares, Phishing, Man-in-the-Middle (MitM), Denial-of-Service (DoS), SQL Injection, Zero-Day Exploit e Tunelamento DNS.

4. SQL Injection

Segundo Yunus *et al.* (2018, p. 215, tradução nossa[6]) "A injeção de SQL é uma técnica para hackers execute consultas SQL maliciosas no servidor de banco de dados".

Existem vários tipos de SQL Injection. De acordo Yiğit e Arnavutoğlu (2017, p. 352) podemos classificá-los como:

- **Tautologias:** Como parte desta técnica, o atacante modifica as instruções SQL, alterando a cláusula WHERE usando termos tautológicos para obter resultados.
- **Consultas Logicamente Incorretas:** O atacante injeta uma instrução SQL ilegal ou incompleta resultando que a mensagem de erro vaza o esquema do banco.
- **Consultas PiggyBacked:** O invasor representa uma ameaça para a integridade dos dados, acrescentando uma consulta maliciosa.
- **Consultas de União:** Este tipo de ataque está na categoria de manipulação de SQL, uma vez que está fazendo operações em *"union select"* injetando consultas maliciosas adicionais a consulta segura original.
- **Procedimento Armazenado:** Esta abordagem está na categoria chamada de injeção de função, é uma técnica capaz de inserir diferentes chamadas de banco de dados, como até mesmo chamadas de sistema operacional.
- **Ataques Baseados em Inferência:** Os ataques baseados em inferência podem ser divididos em 2 categorias, injeção cega e injeção de temporização.
  - **a) Injeção Cega:** Os desenvolvedores ocultam detalhes de mensagens de erro que ajudam usuários mal-intencionados a fazer ataques ao banco de dados. Nesta situação, o atacante ao invés de acessar mensagem de erro genérica,

[5] "Cyber-attack is an attack launched from one or more computers against other computers or networks (either to disable them or to gain access to data and manage them); it compromises information security by affecting its confidentiality, integrity, and availability."

[6] "SQL injection is a technique for hackers to execute malicious SQL queries on the database server"

acessa a mensagem provida ao desenvolvedor. Com essa abordagem, o invasor ainda pode roubar dados perguntando por meio de verdadeiro ou falso.

b) **Injeção de Temporização:** O atacante reúne informações com base em atrasos de tempo de resposta do banco de dados. Este ataque é semelhante à técnica de injeção cega e o invasor pode medir o tempo que a página leva para carregar para determinar se a instrução injetada é verdadeira.

- **Codificações Alternativas:** Neste tipo de ataque, os invasores mudam a consulta usando codificação, como hexadecimal, ASCII e Unicode, dessa forma, eles podem escapar do filtro do desenvolvedor que verifica as consultas de entrada em busca de "caracteres inválidos" especiais conhecidos.

Utilizando a SQL Injection, o atacante pode obter qualquer tipo de dado armazenado no banco e, dependendo da versão do baco de dados, é possível executar comandos para acesso a permissões de administrador e até mesmo acesso as permissões de “root” ao servidor onde o banco está hospedado.

5. Ferramentas Para Segurança do Banco de Dados

5.1. Credenciais e Níveis Diferentes de Acesso

Níveis de controles de acessos, são medidas gerenciais que já estão disponíveis nas SGBD e podem ser aplicadas para que credenciais diferentes possuam permissões, de acesso, alteração, criação e demais operações, diferentes entre si. Isso possibilita a individualidade de cada credencial, ou a liberação e bloqueio por grupos, e se faz necessário para que os usuários tenham acesso somente ao necessário. O método mais comum para controle de acesso é a permissão ou negação de privilégios, e podem ser classificados em dois métodos:

- **Nível de conta**: O DBA especifica os privilégios individuais de cada conta existente no banco de dados, independente das suas relações no mesmo.
- **Nível de relação**: O DBA pode controlar o privilégio para acessar cada relação ou visão individual no banco de dados. Tais privilégios especificam para cada usuário as relações individuais sobre as quais cada tipo de comando pode ser aplicado. (DE SOUSA, DEVMEDIA, 2015)

5.2. Criptografia de Dados

Segundo o site Kaspersky (tradução nossa[7]) “A criptografia é um método de proteção de informações e comunicações por meio do uso de códigos, de forma que somente aqueles a quem as informações se destinam possam lê-las e processá-las.”.

Para que seja possível a aplicação da criptografia em um banco de dados, é necessário definir algumas regras para esta, sendo elas:

[7] “Cryptography is a method of protecting information and communications through the use of codes, so that only those for whom the information is intended can read and process it.”

- **Chave primária:** representa a origem da criptografia utilizada, é única para cada banco de dados e é utilizada na proteção da regra (certificado) a ser implementado.
- **Certificado:** é o segundo objeto a ser criado para implementação da criptografia. É responsável por proteger as chaves secundárias.
- **Chaves Secundárias:** É utilizada para criptografar e descriptogafar os dados. Podendo ser simétricas ou assimétricas.

Alguns dos métodos mais comuns de criptografia são: DES, TRIPLE_DES, TRIPLE_DES_3KEY, RC2, RC4, RC4_128, DESX, AES_128, AES_192 e AES_256, BLOWFISH, TDE, Camellia, RSA, Twofish, SAFER e IDEA.

5.3. Store Procedure

A Store Procedure, ou procedimento armazenado, segundo o escritor Delio, D. do site Elipse (2019), é "um conjunto de instruções desenvolvidas em linguagem T-SQL (Transact-Sql) que, quando armazenadas ou salvas, ficam dentro do servidor de forma pré-compilada.". O SGBD armazena em cache os planos de execução criados, e quando estes são chamados, são reutilizados, evitando que seja necessárias a criação, a análise e a execução de novos procedimentos, economizando tempo e recursos de máquina.

A Store Procedure suporta uma grande gama de comandos DML (Data Manipulation Language) e DDL (Data Definition Language) permitindo assim que seja utilizada em inúmeros casos e funções.

5.4. Backup

O Backup do banco de dados nada mais é do que uma cópia realizada em um momento, e que foi salva exteriormente ao banco.

De acordo com o site da IBM (2018):

> "Para manter a consistência de dados durante a recuperação de dados, é recomendável executar o backup pelo menos uma vez por semana. Ao planejar backups regulares, deve-se considerar a dinâmica de seu ambiente, por exemplo, mudanças na infraestrutura e instalações de software. É uma boa prática fazer backup do banco de dados antes de fazer o upgrade do servidor ou introduzir mudanças significativas no middleware ou no sistema operacional."

5.5. Ocultação de Mensagens de Erros

Muitas vezes em sites, aplicativos, sistemas operacionais, e quaisquer tipos de softwares as mensagens de erros não são tratadas de maneira adequada. Para o SQL Injection, isso é uma enorme vulnerabilidade, pois, invasores que consigam entender a lógica, trecho de código, variáveis, dados ou qualquer informação pertinente na mensagem de erro, podem se aproveitar para buscar aprender mais sobre os dados e funções armazenadas. Dessa forma, faz-se com que estas mensagens sejam o mais genéricas possível, limitando-se ao compreendimento de um usuário comum.

5.6. Atualizações de Sistemas

Atualizações de Sistemas são importantes para todo o contexto de segurança da informação, pois estas buscam corrigir problemas e melhorias em geral. No contexto de SQL Injection as atualizações se tornam importante pois tendem a bloquear falhas e vulnerabilidades detectadas em versões anteriores.

Apesar de importantes, as atualizações podem apresentar alguns riscos, como indisponibilidade temporária, lentidão durante o processo e novas falhas que não foram detectadas durante a fase de testes, dessa forma, torna-se importante também definir horários e procedimentos para que a atualização seja feita da forma mais imperceptível e suave possível para o sistema e os usuários, e é uma excelente prática realizar atualizações somente após a liberação de versões estáveis.

5.7. Validações de Entradas de Formulários

Conforme explicado em tópicos anteriores, o SQL Injection é realizado através de formulários de texto, com a inserção de um trecho de código malicioso, utilizando caracteres especiais para o banco de dados, que são entendíveis como instruções e não como valores. Alguns desses caracteres são “;”, “`”, “--”, “/*” por exemplo. Dessa forma, é uma boa prática, sempre que possível inabilitar a utilização desses caracteres em formulários digitáveis. Outra limitação aplicável a campos digitáveis que auxilia na proteção contra injeções de SQL é a limitação de tamanho de campos sempre que possível.

5.8. Banco de Dados de Log

Um log de transações de um banco de dados é uma maneira que o banco de dados tem de garantir que uma transação que já foi armazenada, através de um *commit*, seja salva mesmo caso haja algum problema de acesso, como a queda de energia, ou qualquer outro problema que faça a conexão com banco de dados ser perdida repentinamente. Além disso, o log, pode auxiliar na restauração de um banco de dados a um estado anterior e estável em caso de algum problema crítico.

Os bancos de dados de log também apresentam alguns empecilhos, sendo o maior deles a utilização de recursos.

5.9. Auditoria de Bando de Dados

A auditoria de banco de dados é uma análise realizada por uma empresa terceira especializada e certificada, a partir de um conjunto de normas e políticas da empresa, da legislação vigente ou de frameworks existentes, com o objetivo de entender as atividades realizadas no banco auditado, e verificar se estas estão de acordo.

A auditoria em bancos de dados pode ser feita de duas maneiras:

a)Metodologia tradicional: Nesse caso a auditoria obtém um entendimento do banco de dados, e das informações e políticas dele e da empresa, referente a todo o tipo

de informação armazenada, e com o auxílio de uma lista de checagem é feita uma varredura por vulnerabilidades e falhas.

b)Exclusão de Riscos: Esta técnica busca realizar o controle da vulnerabilidade e analisar e impor a melhor medida alcançada para cada objetivo, não se limitando apenas a uma técnica, e pode utilizar de técnicas preventivas ou corretivas.

5.10. Monitoramento e Testes de Vulnerabilidade

No banco de dados, pode-se monitorar diversos itens como credenciais utilizadas, operações realizadas, dados inseridos, alterados ou removidos, recursos disponíveis e consumidos, falhas de segurança, tentativas de invasão etc., ou seja, qualquer operação realizada na base, supostamente lícita ou ilícita, pode ser monitorada.

5.10.1. Monitoramento de Bancos de Dados e Ambiente

Segundo o site DEMETIUN2 (s.d.), algumas das melhores ferramentas para monitoramento de bancos de dados são: SolarWinds Database Performance Analyzer para SQL Server, Monitor de rede Paessler PRTG, Monitoramento de servidor Site24x7, SolarWinds AppOptics APM, Atera, Controle de banco de dados dbWatch, Idera SQL Diagnostic Manager, Ferramentas elétricas SQL, Sentinela Um, Monitor SQL de porta vermelha, Auditoria Lepide, Monitor de integridade SQL grátis do ManageEngine, e Monitoramento do Spiceworks

5.10.2. Ferramentas de Vulnerabilidade Voltadas para SQL Injection

Tratando de testes de vulnerabilidades voltados para o SQL Injection, existem diversas ferramentas disponíveis, dentre elas, algumas das mais utilizadas segundo Donda (2019) são:

- **The Mhole:** Ferramenta automatizada de SQL Inection presente no Kali Linux.
- **Blind-Sql-Bitshifting:** Injeção de SQL cega via *bitshifting*.
- **SQLMap:** Ferramenta de injeção SQL e aquisição de banco de dados.
- **BBQSQL:** Ferramenta de exploração de injeção de SQL cega.
- **NoSQLMap:** Mapeador de banco de dados NoSQL automatizado
- **Safe3 SQL Injector:** Ferramenta voltada para SQL Injection com recursos WEB
- **DSSS:** Scanner SQLi
- **jSQL Injection:** Ferramenta Java para injeção automática de banco de dados SQL
- **Blisqy:** Explorar injeção de SQL cego baseada em tempo em cabeçalhos HTTP
- **Whitewidow:** Verificador de vulnerabilidades SQL

6. Discussões

Após os levantamentos de pesquisas apontadas nas sessões anteriores, é observado uma lista de padrões de ciberataques mais comuns à bancos de dados, que utilizam de vários tipos de vulnerabilidades, operacional, falhas de código, ausência de proteção,

falhas humanas, ausência de conformidade, dentre outras. Esta lista de ataques, pode ser colocada da seguinte forma:

- **Phishing:** São comunicações fraudulentas disfarçadas para que pareçam verídicas. O objetivo é obter informações confidenciais ou pessoais ou instalação de um malware ou software duvidoso. De acordo com a empresa Verizon, em seu relatório Verizon Data Breach Investigations Report 2021 (DBIR), em 2021 os casos de *phishing* representam 36% do total de ataques.
- **Engenharia social:** De acordo com o site Kaspersky

  > "Engenharia social é uma técnica empregada por criminosos virtuais para induzir usuários desavisados a enviar dados confidenciais, infectar seus computadores com malware ou abrir links para sites infectados."

  Devido a interligações possíveis entre a engenharia social e os demais tipos de ataques, ela também é uma das mais exploradas vulnerabilidades, apresentando cerca de 35% dos tipos de ataques em 2021, segundo o relatório da Verizon.
- **Privilégios excessivos:** Como privilégios excessivos podemos definir quando usuários, integradores ou aplicativos recebem usuários do banco de dados com privilégios adicionais as suas funções.
- **Exposição de Mídia:** O mal armazenamento de mídias de backup, ou controle de acesso à mídia física do servidor, ou ausência de proteção para bancos alocados em nuvem podem ser caracterizados como a exposição de mídia.
- **Ausência de Atualizações:** Assim como estes falhas e vulnerabilidades são descobertas de forma recorrente, o banco de dados e demais softwares devem também ser atualizados a fim de corrigi-las.
- **Malware:** Malware é o termo usado para descrever software maliciosos. Os malwares são divididos em Vírus, *Trojan*, *Spyware*, *Worms*, *Ransoware*, *Adware* e *Botnets*.
- **SQL Injection:** A injeção de SQL é uma técnica para que sejam injetadas consultas ou comandos maliciosos no banco de dados disfarçados como uma variável, adquirida através de campos editáveis de texto. Segundo o site Appknox (2020, tradução nossa[8]) "Entre 2017 e 2019, cerca de dois terços (65.1% para ser mais preciso) de todos os ataques em softwares de aplicação continham apenas a injeção de SQL".

É expressiva a taxa de participação da injeção SQL nos tipos de ataques mais comuns, conforme levantamento realizado pelo site Appknox (2020), devido a sua facilidade de execução, de forma que até mesmo criminosos com níveis mais baixos de conhecimento conseguem executar e ausência de proteção em sistemas aplicações.

Visando a proteção, ou ao menos a tentativa de mitigação, destes ataques e em principal o SQL Injection, foi discorrido, de forma teórica, sobre algumas ferramentas que podem ser aplicadas nos bancos de dados e ambientes. Ao se basear nas ferramentas apresentadas o administrador de banco de dados pode acrescentar muita segurança para sua base, porém é necessário ter em mente que nunca estará cem por cento seguro, e deve-

[8] "Between 2017 and 2019, around two-thirds (65.1% to be precise) of all the attacks on software applications were SQL Injection attacks only."

se manter atento e receoso principalmente a programas, *links*, pessoas e valores de entradas suspeitos.

7. Considerações finais

A utilização de Sistemas Gerenciadores de Bancos de Dados desde o seu surgimento cresce de maneira exponencial, proporcionando agilidade, organização e economia de recursos para o armazenamento de informações. Por outro lado, culminou-se no surgimento de novas vulnerabilidades e falhas. Sendo um dos principais e mais comuns ataques a essas vulnerabilidades o SQL Injection, demonstrando que a utilização destas ferramentas de armazenamento digital deve ser acompanhada de ferramentas e medidas de segurança.

Utilizando-se de pesquisas por empresas especializadas e estudos realizados por terceiros e de desenvolvimento próprio, obteve-se como resultados os principais tipos de ataques e vulnerabilidades que podem ser enfrentados por profissionais e organizações que utilizam bancos de dados e, a partir dessas informações, foi possibilitado criar e expor uma lista de ferramentas e métodos consolidados que apresentam como resultado uma maior segurança e integridade destes sistemas.

Entretendo, é indispensável que o administrador do banco de dados tenha em mente que a total segurança do sistema é um *status* impossível de ser alcançado, e que sempre estará vulnerável de alguma forma. Também, fica claro que as ferramentas e processos a serem implementados devem ser estudados de maneira minuciosa, pois muitas delas apresentam efeitos colaterais, como uso excessivo de recursos, possível lentidão, altos custos, dentre outros, sendo assim, só é possível apresentar indicações, e não definir ferramentas obrigatórias, ou imprescindível como um todo.

Referências Bibliográficas

APPKNOX. **Biggest Threat to Application Security: SQL Injection Attacks**. Disponível em: https://www.appknox.com/blog/sql-injection-attacks. Acesso em: 17/05/2022.

CAPORALE, Guglielmo Maria *et al*. **Non-linearities: cyber attacks and cryptocurrencies.** Finance Research Letters, v.32, n. 101297, p. 1-7, 2020. Disponível em: https://www.sciencedirect.com/science/article/pii/S1544612319309377. Acesso em: 20/02/2022. DOI: 10.1016/j.frl.2019.09.012

CISCO. **What Are the Most Common Cyber Attacks?**. Disponível em: https://www.cisco.com/c/en/us/products/security/common-cyberattacks.html. Acesso em: 22/02/2022.

DANIEL DONDA. **10 melhores ferramentas para SQL Injection**. https://danieldonda.com/10-melhores-ferramentas-para-sql-injection/. Acesso em: 25/05/2022

DEMENTIUM2. **13 Melhores Ferramentas de Monitoramento de Banco de Dados**. Disponível em: https://dementium2.com/admin-da-rede/13-melhores-ferramentas-de-monitoramento-de-banco/. Acesso em: 8/04/2022.

ELIPSE KNOWLEDGEBASE ENGLISH. **Linguagem SQL: Capítulo 9 – Stored Procedures.** Disponível em: https://kb.elipse.com.br/linguagem-sql-capitulo-9-stored-procedures/. Acesso em: 14/03/2022.

GYORÖDI, Cornelia; GYORÖDI, Robert; SOTOC, Roxana. **A Comparative Study of Relational and Non-Relational Database Models in a Web- Based Application.** International Journal of Advanced Computer Science and Applications, v. 6, n. 11, p. 78-83, 2015. Disponível em: https://thesai.org/Publications/ViewPaper?Volume=6&Issue=11&Code=IJACSA&SerialNo=11. Acesso em: 23/02/2022. DOI: 10.14569/IJACSA.2015.061111

IBM. **Fazendo Backup e Restaurando o Banco de Dados**. Disponível em: https://www.ibm.com/docs/pt-br/license-metric-tool?topic=database-backing-up-restoring. Acesso em: 15/03/2022.

IME USP. **Fundamentos de Armazenamento e Manipulação de Dados**. Disponível em: https://www.ime.usp.br/~andrers/aulas/bd2005-1/aula3. Acesso em: 23 fev. 2022.

KASPERSKY. **Cryptography Definition**. Disponível em: https://www.kaspersky.com/resource-center/definitions/what-is-cryptography. Acesso em: 16/03/2022.

KASPERSKY. **Engenharia social - Definição**. Disponível em: https://www.kaspersky.com.br/resource-center/definitions/what-is-social-engineering. Acesso em: 16/05/2022.

THE ECONOMIST. **The world's most valuable resource is no longer oil, but data**, 2017. Disponível em: https://www.economist.com/leaders/2017/05/06/the-worlds-most-valuable-resource-is-no-longer-oil-but-data. Acesso em: 23/02/2022.

VERIZON. 2022 **Data Breach Investigations Report**. Disponível em: https://www.verizon.com/business/resources/reports/dbir/. Acesso em: 16/05/2022.

YIGIT, Gülsüm, ARNAVUTOGLU, Merve. **SQL Injection Attacks Detection & Prevention Techniques**. International Journal of Computer Theory and Engineering, v. 9, n. 5, p. 351-356, 10/2017. Disponível em: http://www.ijcte.org/vol9/1165-T051.pdf. Acesso em: 24/02/2022.

YUNUS, Mohd Amin et al. **Review of SQL Injection: Problems and Prevention.** INTERNATIONAL JOURNAL ON INFORMATICS VISUALIZATION, v. 2, n. 3-2, p. 215-219, 30 jul. 2018. Disponível em: https://www.mendeley.com/reference-manager/reader/cba4227c-6a80-3d3c-94eb-cf043c805bda/7bee90a1-e835-8209-63b5-ee1a48de17e6/. Acessado em: 20/02/2022.