CENTRO UNIVERSITÁRIO DE ANÁPOLIS – UniEVANGÉLICA
BACHARELADO EM ENGENHARIA DE COMPUTAÇÃO

LUCAS HENRIQUE DE MOURA E SILVA

ESCOLHA DA CRIPTOGRAFIA IDEAL E ANONIMAÇÃO DE DADOS SENSÍVEIS CITADOS A LEI GERAL DE PROTEÇÃO DE DADOS

ANÁPOLIS
2020

LUCAS HENRIQUE DE MOURA E SILVA

ESCOLHA DA CRIPTOGRAFIA IDEAL E ANONIMAÇÃO DE DADOS SENSÍVEIS CITADOS A LEI GERAL DE PROTEÇÃO DE DADOS

Trabalho de Conclusão de Curso II apresentado como requisito parcial para a conclusão do curso de Bacharelado em Engenharia de Computação do Centro Universitário de Anápolis – UniEVANGÉLICA.

Orientador: Prof. Me. William Pereira dos Santos Júnior.
Co-orientadora: Prof. Ma. Natasha Sophie Pereira.

Anápolis
2020

LUCAS HENRIQUE DE MOURA E SILVA

ESCOLHA DA CRIPTOGRAFIA IDEAL E ANONIMAÇÃO DE DADOS SENSÍVEIS CITADOS A LEI GERAL DE PROTEÇÃO DE DADOS

Trabalho de Conclusão de Curso II apresentado como requisito parcial para a conclusão do curso de Bacharelado em Engenharia de Computação do Centro Universitário de Anápolis – UniEVANGÉLICA.

Aprovado(a) pela banca examinadora em [dia] de [mês] de 2020, composta por:

_____
Prof. Me. William Pereira dos Santos Júnior
Orientador

_____
Profa. Ma. Natasha Sophie Pereira
Co-orientadora

_____
Prof. [nome do professor]

_____
Prof. [nome do professor]

Anápolis
2020

RESUMO

Com a grande quantidade de dados de usuários disponíveis na internet, é de fato imprescindível o uso de metodologias de anonimação e criptografias para assegurar os dados dos usuários. Os *softwares* atualmente solicitam informações pessoais para realização de cadastros em suas plataformas com a finalidade de identificar quando executar determinadas ações em sua plataforma. Mas os dados que podem ser vinculados a um usuário para sua identificação são categorizados segundo a Lei Geral de Proteção de Dados como dados sensíveis, tendo que ser tratados para manter a anonimação de pessoas não autorizadas ao acesso destes dados. Os dados sensíveis devem ser assegurados de forma precisa, pensando nisso, neste trabalho, com base na análise da estrutura e da finalidade do *software* e no levantamento dos dados sensíveis armazenados e trafegados no mesmo, e será apresentado como podemos escolher a criptografia ideal para o *software* em estudo.

Palavras-chave: *software*, segurança da informação, lei geral de proteção de dados, LGPD, anonimação de dados, criptografia, dados, informações, dados sensíveis.

ABSTRACT

With the large amount of user data available on the Internet, it is indeed essential to use anonymization and encryption methodologies to ensure user data. The software currently requests personal information for registration in their platforms in order to identify when to perform certain actions in their platform. But the data that can be linked to a user for their identification are categorized according to the General Law of Data Protection as sensitive data, having to be treated to maintain the anonymity of unauthorized persons to access this data. The sensitive data must be assured in a precise way, thinking about it, in this work, based on the analysis of the structure and purpose of the software and the survey of the sensitive data stored and trafficked in it, and it will be presented how we can choose the ideal cryptography for the software under study.

Keywords: software, information security, general data protection law, LGPD, data anonymity, encryption, data, information, sensitive data.

LISTA DE ABREVIATURAS E SIGLAS

3DES	*Triple Data Encryption Standard*
AES	*Advanced Encryption Algorithm*
DDoS	*Distributed Denial of Service*
DER	Diagrama de Entidade Relacional
DES	*Data Encryption Standard*
DES-X	*Data Encryption Standard-X*
DESX	*Data Encryption Standard-X*
IBAN	*International Bank Account Number*
LGPD	Lei Geral de Proteção de Dados
SGBD	Sistema de Gerenciamento de Banco de Dados
SGSI	Sistema de Gestão de Segurança da Informação
SIRENE	Sistema de Registro Nacional de Emissões
SIRET	*Système d'identification du répertoire des établissements*
SQL	*Standard Query Language*
SSL	*Security Sockets Layer*
TLS	*Transport Layer Security*
WEP	*Wired Equivalent Privacy*
XOR	*Exclusive OR*

SUMÁRIO

1. INTRODUÇÃO....8
1.1. Problema....8
1.2. Objetivos....9
1.1.1 Objetivo Geral....9
1.1.2 Objetivos Específicos....9
1.3. Justificativa....9
2. FUNDAMENTAÇÃO TEÓRICA....11
2.1. Tipos de Software....11
2.1.1. Software de Sistema....12
2.1.2. Software Aplicativo....12
2.2. Informação, Dados e Dados Sensíveis....13
2.3. Banco de Dados....13
2.3.1. Sistemas de Gerenciamento de Banco de Dados....14
2.3.2. Comandos de SQL (Structured Query Languange)....14
2.4. Segurança da Informação....15
2.4.1. Modelos e Mecanismos de Segurança....17
2.5. Anonimacao de Dados....17
2.6. Criptografia....21
2.6.1. Criptografia Assimétrica....22
2.6.2. Encriptação Simétrica....22
2.6.3. Função Hash Criptográfica....26
2.6.4. Criptografias em Bancos de Dados....27
3. METODOLOGIA....28
4. DESENVOLVIMENTO E ANÁLISE DE DADOS....29
4.1. Levantamento de Dados, Informações e Dados Sensíveis....29
4.1.1. Informações e Dados Sensíveis do Software Cycleplantões....30
4.2. Escolha da Criptografia para o Software Cycleplantões....31
4.2.1. Escolhendo a criptografia para as colunas no banco de dados....32
4.3. Anonimizando os Dados Sensíveis do Software Cycleplantões....32
4.3.1. Instalando o PostgreSQL Anonymizer....32
4.3.2. Escolha das Técnicas de Anonimizar Dados....33
4.3.3. Aplicação do Método de Anonimação....35
5. CONSIDERAÇÕES FINAIS....37
REFERÊNCIAS BIBLIOGRÁFICAS....38

1. INTRODUÇÃO

1.1. Problema

Seja para diversão, trabalho, estudo ou passatempo, os *softwares* estão cada vez mais presentes em nossas vidas, segundo o E-Commerce Brasil (2020), o crescimento do mercado virtual brasileiro foi de 56,8% até o mês de agosto de 2020, a projeção de crescimento deste mercado era de 18% e saltou para 30%, com este crescimento no mercado virtual, o número de usuários da internet segundo o canaltech (2020) aumentou para 74%, e com este aumento, o número de dados pessoais de usuários trafegados e armazenados são maiores, sendo necessário manter estes dados seguros.

O relatório publicado pela Unisys (2020) revelou que o Brasil é o país que possui o maior crescimento em preocupações com relação a assuntos de segurança em tecnologia, alcançando 197 pontos em um máximo de 300, demonstrando um aumento de 7 pontos em relação ao ano de 2019.

Para proteção dos dados no cenário de transição pela rede, há o certificado SSL (*Secure Socket Layer*) que, segundo a HostGator (2019), criptografa as informações cambiadas entre as máquinas, porém ainda existe a possibilidade de interceptação. Há também outras formas de se assegurar a confidencialidade de um *software*, como no próprio *backend* (parte responsável por todo o processamento dos dados da aplicação) no qual podem ser implementados quais hosts tem permissão para acessá-lo.

Segundo a MalwareBytes (2018) em um cenário onde o hacker consiga burlar todas as camadas de segurança do servidor e possua acesso às informações lá armazenadas pode-se utilizar a criptografia a nível de banco de dados, onde os dados sensíveis armazenados dos usuários são levantados de modo a avaliar o melhor algoritmo de criptografia para a finalidade *software*, trazendo maior segurança ao *software*.

Segundo a Rocketcontent (2019) com o grande avanço da tecnologia, foram desenvolvidos *softwares* para diversos públicos, a quantidade de informações pessoais e empresariais ficaram mais expostas, e com isso é inevitável a utilização de métodos para assegurar essas informações. Segundo o portal IG (2019), *softwares* como o WhatsApp utilizam a criptografia simétrica como forma de segurança para troca de mensagens entre usuários, e seus backups são criptografados com a mesma criptografia, eles também utilizam o Spark que armazenam arquivos de logs de suas conversas em suas pastas de instalação, mas oferece a possibilidade de armazenar essas mensagens sem criptografá-las.

Segundo o blog de notícias G1 (2011), a *Playstation network* foi invadida em 2011 e com isso cerca de 2,2 milhões de usuários tiveram suas informações pessoais roubadas, inclusive dados de seus cartões de créditos. Ao se analisar o fato ocorrido na *Playstation Network* em 2011, fica clara a importância da criptografia em dados sensíveis de usuários. No ano de 2018, no Brasil foi instituída a Lei Geral de Proteção de Dados (LGPD) (Brasil, 2018), que rege sobre a necessidade de se dificultar a legibilidade de dados pessoais e empresariais nos sistemas. A partir da análise estrutural e funcional de um sistema e estudo de sua finalidade, é necessário realizar um levantamento sobre quais os dados deverão trafegar e serão armazenados em sua base, de modo que, a partir deste levantamento, é possível verificar quais destes são dados sensíveis e necessitam de uma proteção mais efetiva. Neste sentido, como escolher a criptografia ideal para um sistema com base no levantamento dos dados que ele deverá persistir?

1.2. Objetivos

1.1.1 Objetivo Geral

Analisar os dados e informações a serem armazenadas e trafegadas em um software, a fim de auxiliar na escolha da criptografia ideal e na anonimação de dados sensíveis citados pela Lei Geral de Proteção de Dados.

1.1.2 Objetivos Específicos

- Identificar o tipo de *software*.
- Identificar dados sensíveis que serão armazenados ou trafegados no sistema.
- Escolher a criptografia ideal baseando-se no tipo de software e os dados sensíveis que serão armazenados nele.
- Escolher os métodos de anonimização dos dados.

1.3. Justificativa

A segurança da informação, é o conjunto de medidas a serem aplicadas para manter a integridade, a confidencialidade e a disponibilidade de dados armazenados em *softwares*, de modo a amenizar ou extinguir qualquer dano ao sistema (STALLINGS; BROWN,2014, p.26). Ela é tão importante quanto qualquer outra medida de segurança, pois fragilidades na segurança de dados podem acarretar diversos problemas para a instituição, tais como os ocorridos na *Playstation Network*, que foi invadida por hackers que conseguiram acessar dados dos cartões de créditos dos clientes da empresa.

A criptografia de dados armazenados em um *software* tem o intuito de evitar problemas como o supra citado. Segundo Moreno et al. (2006, p. 21), criptografia é um conjunto de técnicas pensadas para proteger uma informação de modo que apenas quem a emitiu consiga decriptografá-la. Essas técnicas foram criadas durante a segunda guerra mundial, para que as transmissões de rádio de um governo não pudessem ser ouvidas ou interpretadas por seus inimigos, ou por pessoas que não deveriam recebê-las. Para que a criptografia seja efetiva, é necessário executar um algoritmo que codifique os dados para que fiquem irreconhecíveis, de mesmo modo, é necessário um algoritmo para decriptografar estes dados a partir de uma chave específica, essas técnicas de criptografia são grandes aliadas da segurança do banco de dados, podendo trabalhar em conjunto, porém com algumas restrições, visto que o mal uso da criptografia pode comprometer significativamente o desempenho do banco de dados, porém quando bem utilizada o desempenho do seu banco não é comprometido.

Segundo Valente (2019), em 2009 o índice de crescimento na segurança da informação marcava 119, já em 2019 passou a ser 175, ou seja, houve um aumento de quase 50%. Isso demonstra que, como profissionais de Tecnologia da Informação, devemos ter em mente como assegurar fielmente estes dados para que os usuários se sintam mais seguros, e um dos caminhos é o uso da criptografia nos dados. No decorrer deste trabalho, será estudado como escolher uma boa criptografia de acordo com o nicho do sistema de modo a torná-lo mais seguro.

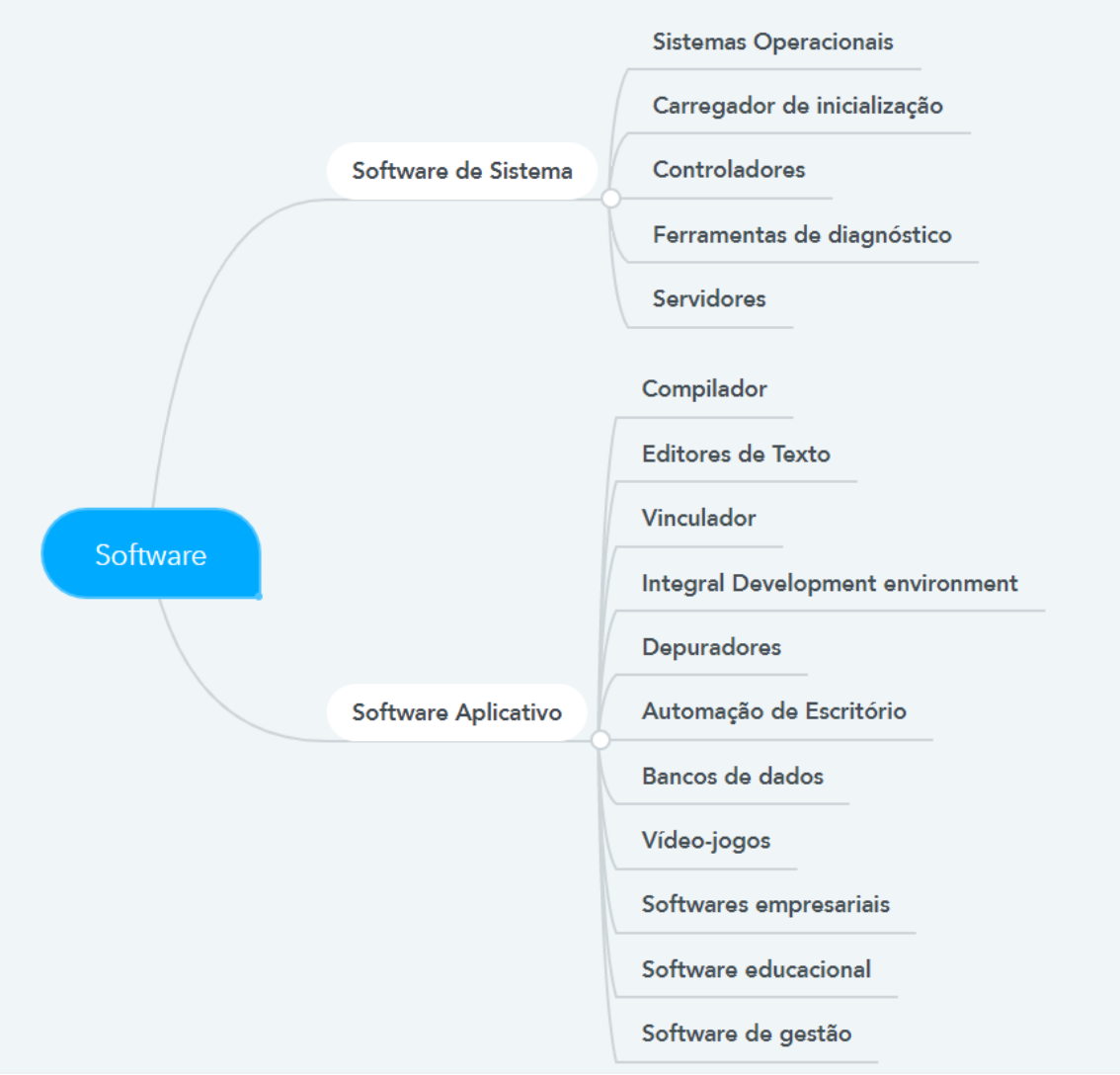
2. FUNDAMENTAÇÃO TEÓRICA

Nesta seção são apresentados conceitos importantes para este trabalho e que serão necessários para os próximos capítulos.

2.1. Tipos de Software

Segundo Stair e Reynolds (2011), um *software* consiste em programas que comandam a operação do computador ou aparato eletromecânico, sendo um termo genérico utilizado para descrever programas, aplicativos, *scripts* e macros, de modo que ao executá-lo, o mesmo passará instruções ao mecanismo informando-o como trabalhar. De acordo com os autores, os *Softwares* podem ser classificados em dois grandes grupos: i) *Software* de sistema; ii) *Software* aplicativo (Figura 1).

Figura 1 - Classificações de softwares



Fonte: Imagem de classificação de software baseada em conceitos de Stair e Reynolds.

2.1.1. Software de Sistema

Segundo Stair e Reynolds (2011, p. 170), *softwares* básicos, ou *softwares* de sistema, gerenciam o trabalho interno dos eletrônicos, como a comunicação dos periféricos[1], o processador central e as linhas de comunicação, e são eles:

- **Sistema Operacional:** Conjunto de *softwares* que tem por finalidade a gerência de recursos do sistema.
- **Carregador de inicialização:** Parte do sistema operacional que tem como função enviar as informações para a memória com a finalidade de executar programas.
- **Controlador:** *Softwar* que possibilita que o sistema operacional realize interações entre o *hardware* e a interface do usuário.
- **Ferramenta de diagnóstico:** Tem a finalidade de monitorar e controlar as funcionalidades dos recursos físicos do dispositivo.
- **Servidor:** Conjunto de ferramentas de execução, que são usadas para receber instruções de usuários para executar uma ação de acordo.

2.1.2. Software Aplicativo

Segundo Stair e Reynolds (2011, p. 173) são criados para o usuário final, com o objetivo de suprir uma necessidade específica.

- **Compilador:** Tradutor que transforma a linguagem de programação em um programa em si.
- **Editor de Texto:** Usado para criar e gerenciar arquivos digitais que são compostos por textos, cumprem a função de ler o arquivo e interpretá-lo em bytes, baseando-se no código do editor.
- **Vinculador:** Pega os objetos do processo de compilação, descartando os desnecessários e realiza um vínculo do código à biblioteca, produzindo um arquivo executável.
- **IDE (*Integral Development Environment*):** Aplicativo que fornece ao programador serviços que facilitam o desenvolvimento de um *software*.
- **Depurador:** Verifica e limpa os erros de outros programas.
- **Automação de Escritório:** Utilidades que foram projetadas para cumprir tarefas de forma otimizada, automatizando e melhorando as tarefas nessa atividade.

[1] Dispositivo auxiliar usado para enviar ou receber informações do computador.

- **Banco de dados:** Informações armazenadas de forma estruturada, para que um especialista possa acessar fragmentos a qualquer momento.
- **Videojogos:** Jogos eletrônicos onde haja interação de uma ou mais pessoas por meio de imagens de vídeo, controles físicos ou movimentos corporais.
- ***Software* empresarial:** Criado para automatizar, otimizar ou medir produtividade de uma empresa ou indústria ajudando em todas as atividades do negócio.
- ***Software* educacional:** Produtos digitais que tem como objetivo ensinar um tema específico a um usuário.
- ***Software* de gestão:** Conjunto de ferramentas para serem utilizadas em tarefas relacionadas à administração ou ao cálculo numérico.

2.2. Informação, Dados e Dados Sensíveis

Para que seja possível entender a necessidade de se criptografar alguns dados ou informações, inicialmente é preciso compreender a diferença entre dado e informação. Segundo SEMIDÃO (2014, p. 70), os dados não possuem significados relevantes e nenhuma compreensão por si só. É algo sem sentido a princípio, portanto, não tem valor algum para embasar conclusões. Como por exemplo vários arquivos contendo dados sobre diversos tipos todos embaralhados, esses dados sem serem tratados não possuem valor.

Ainda segundo SEMIDÃO (2014, p. 74) uma informação é a ordenação e organização dos dados de forma que façam sentido e transmitam um significado. Sendo assim um conjunto ou consolidação dos dados de forma que respalde o conhecimento.

De acordo com a LGPD (Brasil, 2019), dados sensíveis são identificados como: i) dados pessoais que revelem a origem étnica ou racial, opiniões políticas e convicções religiosas ou filosóficas; ii) filiação sindical; iii) dados genéticos, dados biométricos tratados para identificar um ser humano; iv) dados relacionados com a saúde; v) dados relativos à vida sexual ou orientação sexual.

2.3. Banco de Dados

Segundo Date (2004), banco de dados é um sistema de armazenamento de dados, sendo assim, seu principal objetivo global é registrar e manter a informação, podendo ser quaisquer informações que sejam consideradas significativas para a organização que é servida pelo sistema.

2.3.1. Sistemas de Gerenciamento de Banco de Dados

Segundo Date (2004) o SGBD (Sistema de Gerenciamento de Banco de Dados), é um conjunto de requisitos e funcionalidades que oferecem: Segurança, Integridade, controle de concorrência e recuperação a falhas. De acordo com a AWS Amazon (2018), os SGBD's mais utilizados são: PostgreSQL, MySQL, MariaDB, Oracle e SQLServer.

2.3.1.1. Arquiteturas de um SGBD

Segundo Takai, Italiano, e Ferreira (2005, p. 11) existem quatro tipos de arquiteturas em um SGBD, sendo elas: Plataformas centralizadas, Sistemas de Computador Pessoal, Banco de dados Cliente-Servidor e Banco de dados Distribuídos.

- Arquitetura centralizada: existe um computador com grande capacidade de processamento, sendo hospedeiro do SGBD e emuladores para os vários aplicativos que irão utilizá-lo, sua arquitetura tem como objetivo principal permitir que muitos usuários possam manipular um grande volume de dados, mas sua desvantagem é seu alto custo financeiro.
- Sistemas de computador pessoal: fazem seus processamentos sozinhos. No começo esse processamento é bastante limitado, porém, com a evolução do *hardware*, tem-se computadores pessoais com grande capacidade de processamento.
- Bancos de dados cliente-servidor: o cliente executa suas tarefas em um aplicativo, ou seja, fornece a interface do usuário. O servidor executa os comandos solicitados pelo cliente, e retorna os resultados esperados.
- Bancos de dados distribuídos: cada servidor atua como um sistema cliente-servidor, porém as consultas realizadas pelos aplicativos são feitas para qualquer servidor indiretamente, e caso a informação solicitada seja mantida em outro servidor ou servidores, o sistema é encarregado de obter todas as informações necessárias para o aplicativo.

2.3.2. Comandos de SQL (Structured Query Languange)

Segundo Nield (2019), a linguagem SQL é dividida em DDL (*Data Definition Language*) e DML (*Data Manipulation Language)* e cada uma delas possui comandos específicos para finalidades diferentes.

Ainda de acordo com o autor, os comandos DDL têm por finalidade lidar com esquemas e descrições do banco de dados e como os dados devem ser salvos na base de dados. Seus comandos são (NIELD, 2019):

- CREATE – utilizado para criar banco de dados e seus objetos como: tabelas, visualizações, índice, procedimento de armazenamento, gatilhos e funções.
- ALTER – altera a estrutura do banco de dados.
- DROP – exclui objetos do banco de dados podendo ser utilizado também para a exclusão do banco de dados.

Segundo Nield (2019), os comandos DML lidam com a manipulação dos dados, usados para armazenar, modificar, recuperar, excluir e atualizar dados no banco de dados. São eles:

- SELECT - recupera dados do banco de dados
- INSERT - insere dados em uma tabela
- UPDATE - atualiza os dados existentes em uma tabela
- DELETE - exclui todos os registros de uma tabela de banco de dados

2.4. Segurança da Informação

A segurança da informação é um conjunto de métodos de proteção aplicados em dados diversos. Segundo a ISO/IEC 27002 (2013, p. 22), a segurança da informação vem para minimizar riscos e maximizar o retorno do investimento em sua implantação. São um conjunto de práticas a serem implementadas, tais como: políticas, processos, procedimentos, estruturas organizacionais e funções de *hardware* e *software*.

Essas práticas são fundamentais para a segurança dos dados, e precisam ser monitoradas a todo momento sendo necessário um planejamento para implementá-las. Segundo a ISO/IEC 27002 (2013, p. 22) alguns termos devem ser definidos ao se tratar de segurança da informação. São eles:

- **Incidente de Segurança:** Qualquer evento que envolve a segurança da informação, como roubo de dados e ataques *Distributed Denial of Service* (DDoS[2]).
- **Ativo:** Qualquer informação que possui algum valor para a empresa.

[2] Segundo a KASPERSKY (2016), servidores web atendem um limite finito de solicitações simultâneas e o ataque DDoS aproveita de tais limites enviando múltiplas solicitações para o recurso Web com a finalidade de excedê-lo, impedindo o funcionamento do servidor. As solicitações podem ser acessos sem permissões, vazamentos de informações, vírus e códigos maliciosos, sequestro de dados, desfiguração de sites, modificação de um sistema sem consentimento prévio do proprietário, entre outros.

- **Ameaça:** Evento ou atitude indesejável que remove, desabilita ou destrói um recurso. Elas se aproveitam de falhas de segurança do estabelecimento ou organização. Tendo a possibilidade de explorar acidentalmente ou propositalmente uma vulnerabilidade específica.
- **Vulnerabilidades:** Pontos que podem ser explorados em um sistema comprometendo sua segurança. Configurações no computador ou na rede podem fazer com que essas vulnerabilidades sejam exploradas.
- **Risco:** Qualquer evento que cause impacto negativo que impossibilita a capacidade de empresas alcançarem seus objetivos de negócio. Sendo assim a probabilidade de uma fonte de ameaça, explorar uma vulnerabilidade, resultando em um impacto na organização.
- **Ataque:** São situações em que um cibercriminoso, tenta danificar, destruir ou violar uma rede ou sistema. Tem por objetivo acessar dados sigilosos de empresas ou usuários podendo utilizá-los para fins criminosos.
- **Impacto:** Deve ser analisado com base no valor que a informação tem para empresa ou usuário no caso de roubo, alteração ou exclusão destas.

Segundo STALLINGS (2015, p. 20), existem 3 fundamentos que são primordiais ao se tratar da segurança da informação, que são a confidencialidade, a integridade e a disponibilidade das informações.

Confidencialidade é a informação restrita e somente disponível para usuários autorizados, é comum que as informações sejam categorizadas com base no seu nível de criticidade, ou seja, a extensão do dano que poderia ser causado, caso fossem expostas para os não autorizados. Integridade tem por finalidade defender informações contra modificações ou destruição imprópria, garantindo o não repúdio e a autenticidade destas informações. A perda da integridade consiste na modificação ou destruição não autorizada de informações. Disponibilidade é assegurar que o acesso às informações seja confiável e realizado no tempo apropriado. Uma perda de disponibilidade se baseia na disrupção do acesso à utilização das informações ou de um sistema de informação (STALLING, 2015, p. 8-9).

Ainda segundo STALLINGS (2015, p. 20) existem definições necessárias ao se tratar da segurança da informação, tais como: i) identificação: permitir que uma entidade se identifique; ii) autenticação: verificar a veracidade da entidade, como pré-requisito ao acesso a sistemas de informação; iii) autorização: permitir que entidades realizem determinadas ações; iv) não repúdio: evitar que entidades neguem ações realizadas por ela.

2.4.1. Modelos e Mecanismos de Segurança

Existem formas de aplicar modelos e mecanismos de segurança para evitar possíveis transtornos, um exemplo é a ISO/IEC 27002 (2013, p.22), que é projetada para empresas que desejam implantar um Sistema de Gestão de Segurança da Informação (SGSI), que tem por finalidade a orientação para a implementação desses métodos sendo diversas políticas de segurança da informação, como por exemplo:

- **Controle de acesso**: Conjunto de procedimentos e medidas com o objetivo de proteger dados de usuários evitando o acesso por pessoas não autorizadas.
- **Cópias de segurança (*Backup*)**: É fundamental realizar *backups* de qualquer sistema, de um dispositivo de armazenamento a outro para que possam ser restaurados em caso de perda dos dados originais.
- **Registro de eventos**: Utilizado para a identificação de atacantes, delineando o modo de ataque e, também, para identificar as falhas que foram exploradas.
- ***Firewall***: É um dispositivo de segurança da rede que monitora todo o seu tráfego de entrada e saída, decidindo se é permitido ou não tráfegos específicos de acordo com um conjunto definido de regras de segurança.
- ***Antimalwares***: Remove vírus e códigos maliciosos de um computador.
- **Certificado digital**: Arquivo eletrônico que tem por finalidade identificar pessoas ou empresas, garantindo sua autenticidade sem a necessidade de uma representação presencial.
- **Assinatura digital**: Arquivo eletrônico que tem por finalidade autenticar documentos garantindo validade jurídica a um arquivo.

2.5. Anonimacao de Dados

A lei LGPD (2018) explica que os dados anonimizados, são aqueles que originalmente são relativos a uma pessoa, mas passaram por etapas que garantiram a desvinculação de tais dados a este usuário. Estas técnicas são utilizadas em ambientes de teste, para que o foco de manter assegurados os dados sensíveis fique apenas no ambiente de produção.

Os bancos de dados possuem funções para auxiliar na anonimização dos dados, fazendo com que o dado tratado não permita identificar o usuário. As técnicas de anonimização dos dados, de acordo com PostgreSQL Anonymizer (2018) são:

- **Adição de ruídos:** Aplicar uma variação de valor desejado como por exemplo, aplicar uma variação de +10% ou -10% em uma coluna de salário ou até mesmo a adição de um ruído de dois anos em um campo data.
- ***Randomization* (Randomização):** Existem diversas funções de randomização para gerar dados puramente aleatórios.
- ***Faking* (Falsificação):** Substituir os dados confidenciais por valores aleatórios, evitando qualquer identificação do registro de dados, mas permanecendo adequado para teste, análises e processamento de dados.
- ***Advanced Counterfeiting* (Falsificação Avançada):** Gerar dados falsos na base de dados utilizando uma biblioteca *python faker*[3], sendo necessário realizar a instalação de outros *softwares*.
- **Pseudonimização:** Semelhante à falsificação pois ela gera valores realistas, e sua principal diferença é que sempre retornarão o valor falso, ela utiliza um número de *salt* sendo opcional informá-lo. O *salt* tem por finalidade aumentar a complexidade e evitar ataques de força bruta.
- **Codificação parcial:** Substituir algumas partes de dados, por exemplo, um e-mail "felipe2131@gmail.com" poderia ser substituído por "******2131@gm***.com".
- **Generalização:** Substituir o valor original por um intervalo contendo esses valores, por exemplo, "Felipe tem 24 anos", com a generalização poderia ser dito que "Paulo tem entre 20 e 40 anos".

Segundo a documentação PostgreSQL anonymizer (2018) cada técnica possui diversos comandos com diferentes objetivos, são eles:

- Adição de ruídos:
    - *anon.noise(original_value, ratio)* onde o campo *original_value* pode ser *integer*, *bigint* ou *double precision*¸ e o campo *ratio* é o tamanho do ruído que se deseja adicionar.
    - *anon.noise(original_value, interval)* onde o campo *original_value* pode ser uma data, e o campo *interval* é informado em dia (1 *day*), meses (3 *month*) ou anos (2 *years*).
- *Randomization (*Randomização*):*
    - *anon.random_date()* retorna uma data aleatória.

[3] Disponível em: https://gitlab.com/dalibo/postgresql_faker.

- *anon.random_date_between(d1, d2)* retorna uma data entre as datas informadas.
- *anon.random_int_between(i1, i2)* retorna um número inteiro entre os valores informados.
- *anon.random_bigint_between(b1, b2)* retorna um número inteiro grande entre os valores informados.
- *anon.random_string(n)* retorna um *TEXT* contendo um número de letras desejadas, basta informá-lo no *n*.
- *anon.random_zip()* retorna um código de 5 dígitos.
- *anon.random_phone(p)* retorna um telefone com 8 dígitos com o prefixo que você deseja, basta informá-lo no *p*.
- *anon.random_in(ARRAY[1,2,3])* retorna um elemento de matriz inteira, basta informá-la dentro da *ARRAY*.
- *anon.random_in(ARRAY['a','b','c']* retorna um elemento de matriz em TEXT, basta informá-la dentro da ARRAY.

- *Faking* (Falsificação):
  - *anon.fake_first_name()* retorna um primeiro nome genérico.
  - *anon.fake_last_name()* retorna um sobrenome genérico.
  - *anon.fake_email()* retorna um e-mail genérico.
  - *anon.fake_city()* retorna uma cidade existente.
  - *anon.fake_city_in_coutry(c)* retorna uma cidade no país desejado, basta informá-lo no *c* .
  - *anon.fake_region()* retorna uma região existe.
  - *anon.fake_region_in_country(c)* retorna uma região no país desejado, basta informá-lo no *c*.
  - anon.fake_country() retorna um país.
  - *anon.fake_company()* retorna um nome genérico de empresa.
  - *anon.fake_iban()* retorna um IBAN válido.
  - *anon.fake_siret()* retorna um SIRET válido.
  - *anon.fake_siren()* retorna um SIREN válido.

Para colunas com valor de TEXT e VARCHAR, pode ser utilizado o gerador de textos *Lorem Ipsum*.

- *anon.lorem_ipsum()* retorna 5 parágrafos de texto.
- *anon.lorem_ipsum(2)* retorna 2 parágrafos de texto.
- *anon.lorem_ipsum(paragraphs := 4)* retorna 4 parágrafos de texto.
- *anon.lorem_ipsum(words := 20)* retorna 20 palavras.
- *anon.lorem_ipsum(characters := 7)* retorna 7 caracteres.
- Pseudonimização:
    - *anon.pseudo_first_name('seed','salt')* retorna um primeiro nome genérico.
    - *anon.pseudo_last_name('seed','salt')* retorna um sobrenome genérico.
    - *anon.pseudo_email('seed','salt')* retorna um endereço de e-mail válido.
    - *anon.pseudo_city('seed','salt'*) retorna uma cidade existente.
    - *anon.pseudo_region('seed','salt')* retorna uma região existente.
    - *anon.pseudo_country('seed','salt')* retorna um país.
    - *anon.pseudo_company('seed','salt')* retorna um nome genérico de empresa.
    - *anon.pseudo_iban('seed','salt')* retorna um IBAN válido.
    - *anon.pseudo_siret('seed','salt')* retorna um SIRET válido.
    - *anon.pseudo_siren('seed','salt')* retorna uma SIRENE válida.
- Codificação parcial:
    - *anon.partial('abcdehij',1,'xxxx',3)* retornará 'axxxxhij';
    - *anon.email('teste@gmail.com')* se tornará 'te**@gm***';
- Generalização:
    - *anon.generalize_int4range (value, step)* utilizados para números inteiros.
    - *anon.generalize_int8range (value, step)* utilizado para números inteiros longos.
    - *anon.generalize_numrange (value, step)* utilizado para números decimais.
    - *anon.generalize_tsrange (value, step)* utilizado para *timestamp without time zone.*
    - *anon.generalize_daterange (value, step)* utilizado para datas.

2.6. Criptografia

A criptografia é a prática de codificar informações legíveis através de algoritmos que convertem um texto original em um texto completamente ilegível, sendo possível realizar a decriptografia, processo inverso, para recuperar essas informações (SIMON, 1999).

Segundo STALLINGS (2014), na criptografia existem duas áreas principais: i) os protocolos e os algoritmos de criptografia que possuem uma ampla gama de aplicações; e ii) segurança de rede e de Internet, que se baseia de forma expressiva em técnicas de criptografia.

Segundo Sousa; Moreira e Machado (2010) diversas empresas estão migrando o armazenamento de seus dados para a nuvem, pois estão procurando maior disponibilidade e agilidade em seus negócios. Com a tendência de a quantidade dados e informações que estão sendo armazenadas em nuvem aumentar, é de grande importância o uso da criptografia para evitar que informações sejam visualizadas, acessadas ou alteradas por usuários sem permissão.

Os algoritmos e protocolos podem ainda ser divididos em quatro principais áreas: i) encriptação simétrica; ii) encriptação assimétrica; iii) algoritmos de integridade de dados; e iv) protocolos de autenticação (STALLINGS, 2014).

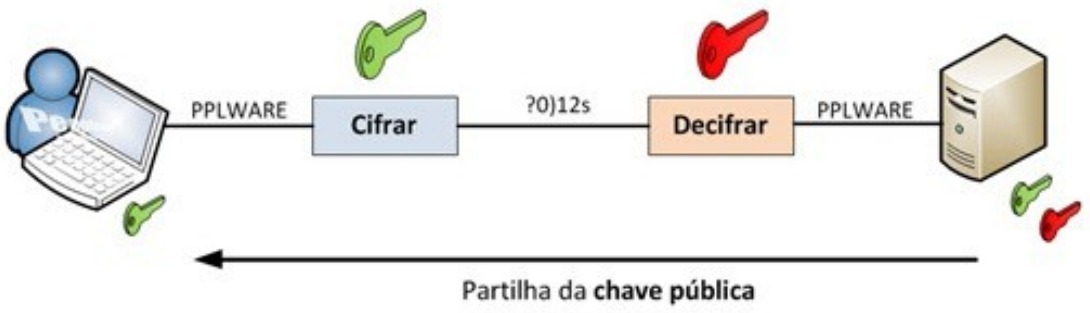
2.6.1. Criptografia Assimétrica

Segundo STALLINGS (2014) é utilizada para cifrar dados pequenos, sua principal utilização é na assinatura digital e para realizar a troca de chaves de forma segura como exemplificado na figura 2. Esse tipo de criptografia está associado a criptografia de chaves, sendo elas, chave pública e chave privada, que serão utilizadas para cifrar e decifrar, respectivamente, esses dados.

O primeiro algoritmo segundo Sousa (2013) a cifrar e decifrar utilizando o princípio de chaves públicas e privadas foi o RSA (*Rivest-Shamir-Adleman*) suas chaves são geradas por números primos, e suas principal vantagem é a facilidade de realizar a multiplicação desses números e a dificuldade de decomposição desse resultado. Segundo o autor, o Elgamal é outro algoritmo muito conhecido por sua finalidade de realizar assinaturas digitais em documentos, e sua força proveniente de calcular um logaritmo discreto.

Figura 2 - Funcionamento básico da chave assimétrica[4].

[4] Disponível em: https://pplware.sapo.pt/tutoriais/networking/criptografia-simetrica-e-assimetrica-sabe-a-diferenca/.



Fonte: PPLWare (2010)

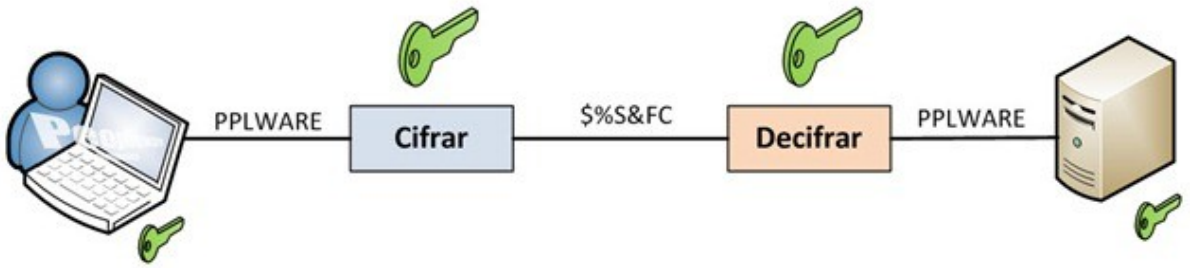
2.6.2. Encriptação Simétrica

Segundo STALLING (2014), a encriptação simétrica é utilizada para ocultar o conteúdo dos blocos ou fluxos contínuos de dados de qualquer tamanho, incluindo mensagens, arquivos, chaves de encriptação e senhas, este sistema possui cinco itens, são eles: i) texto claro, que é a mensagem ou os dados originais que serão encriptados; ii) algoritmo de encriptação que é o responsável por realizar as substituições e transformações do texto claro; iii) chave secreta, que, assim como o texto claro, também é uma entrada para o algoritmo de encriptação; iv) texto cifrado que é a mensagem criptografada, produzida pelo algoritmo de encriptação; e v) algoritmo de decriptação, que é responsável por fazer o trabalho inverso, ou seja, receber o texto cifrado e a chave secreta e por fim produzir o texto claro, como mostrado na figura 3.

Para o uso seguro da encriptação simétrica, Stallings (2014) cita que dois requisitos[5] devem ser levados em consideração. O primeiro apresenta a obrigatoriedade de se possuir um algoritmo de encriptação forte[6] para que, no mínimo, um oponente que conheça o algoritmo e tenha acesso a um ou mais textos cifrados seja incapaz de decifrá-los. O segundo rege que tanto o emissor quanto o receptor precisam obter de forma segura cópias da única chave secreta e mantê-la protegida.

Figura 3 - Funcionamento básico da criptografia simétrica[7].

[5] SOMMERVILLE (2007) cita que "Requisitos são objetivos, propriedades, restrições que o sistema deve possuir para satisfazer contratos, padrões ou especificações de acordo com o usuário."
[6] Segundo STALLING (2014) encriptação forte é o texto (dado) cifrado que é muito difícil decifrar sem possessão da ferramenta apropriada para decodificação.
[7] Disponível em: https://pplware.sapo.pt/tutoriais/networking/criptografia-simetrica-e-assimetrica-sabe-a-diferenca/.



Fonte: PPware (2010)

Estes algoritmos são divididos em: Cifra de Bloco e Cifra de Fluxo.

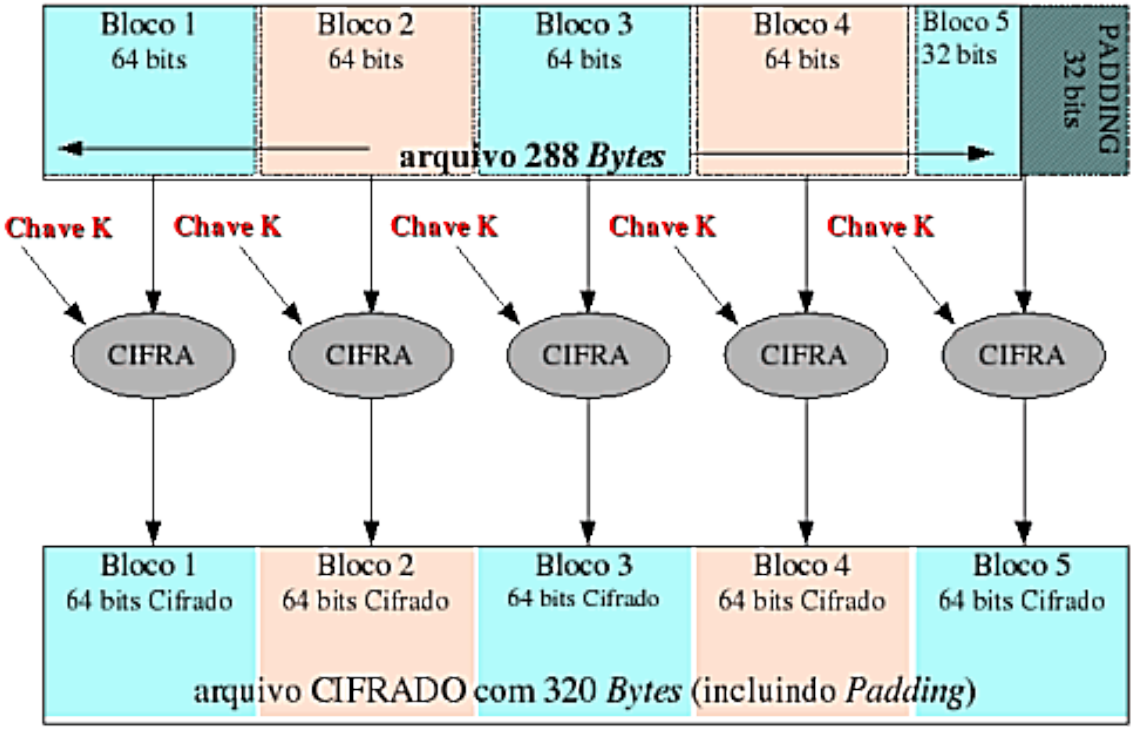
2.6.2.1. Cifra de blocos

Segundo Biryukov e Shamir (2000) este modelo tem como princípio a cifra de blocos com tamanho definido, de modo que a mensagem criptografada deve ter o mesmo tamanho da mensagem contida no texto claro. Um dos algoritmos mais conhecidos que trabalha deste modo é o DES (*Data Encryption Standard*) que foi desenvolvido pela IBM. Por ser uma das primeiras criptografias utilizadas ela é considerada uma proteção básica com cerca de 56 bits.

Existem outras variações da criptografia DES, algumas delas são: i) 3DES, que foi projetada para encriptar blocos de 64 bits por meio de três chaves independentes, também de 64 bits, tanto para encriptar quanto para decriptar, sendo que 8 destes bits são utilizados para verificação de paridade; ii) DES-X ou DESX (*Data Encryption Standard-X*), que usa uma técnica conhecida como *key whitening*, que tem por objetivo dificultar ataques de força bruta[8] (BAKHTIARI; MAAREF, 2011).

Figura 4 - Funcionamento básico da cifra de bloco[9].

[8] De acordo com a KASPERSKY (2020), “Um ataque de força bruta consiste em uma tentativa de violar uma senha ou um nome de usuário, encontrar uma página da Web oculta ou descobrir uma chave usada para criptografar uma mensagem, usando uma abordagem de tentativa e erro e esperando que, em algum momento, seja possível adivinhá-la”.
[9] Disponível em: https://www.projetoderedes.com.br/artigos/artigo_cifras_em_bloco_cifras_de_fluxo.php.



Fonte: Projeto de Redes (2020)

Bakhtiari e Maaref (2011) afirmam que ainda é possível, para o modelo de cifra de blocos, se utilizar o algoritmo AES (*Advanced Encryption Standard*), que é o contrário do DES, pois não utiliza a cifra de bloco como base. Neste modelo AES são utilizadas chaves com tamanhos que variam de acordo com a quantidade de *rounds*[10], por exemplo, para 10 *rounds* o tamanho da chave é 128 bits, para 12 *rounds* é 192 bits e para 14 *rounds* é 256 bits. O AES opera em uma matriz 4x4 de modo que todos os dados dos *rounds* são idênticos com exceção do último.

2.6.2.2. Cifra de fluxo

Algoritmos que fazem cifra de fluxo são aqueles que fazem a criptografia bit a bit ou byte a byte. Este modelo tem desempenho superior ao modelo de cifra de blocos, sendo muito utilizado em sistemas de redes, tais como *bluetooth*, o SSL[11] (*Secure Sockets Laye*r), entre outros. É possível transformar um algoritmo de cifra de bloco em um de cifra de fluxo, bastando definir, para isso, que o tamanho do bloco seja bit ou byte (BAKHTIARI; MAAREF, 2011).

[10] Algoritmos de chave simétrica baseados em blocos realizam um conjunto de passos repetidas vezes, denominados rounds.
[11] Cria um canal criptografado entre o servidor e o navegador para garantir que todos os dados durante a transmissão fiquem seguros.

Segundo BAKHTIARI & MAAREF (2011) Cifras de fluxo geram, a partir de uma chave inicial, uma sequência de bits que será usada como chave, essa sequência é conhecida como *keystream*. A encriptação ocorre pela combinação do texto plano com a *keystream* através de operações XOR[12]. Alguns dos algoritmos de cifra de fluxo mais populares são: A5/1, A5/2, E0 e RC4

- **Algoritmo A5/1:** Neste algoritmo existem 3 registradores, cada um contendo bits importantes para a criptografia, cujas posições são pré-determinadas. O tamanho dos registradores são: 19 bits para o primeiro registrador (R1); 22 bits para o Segundo (R2); e 23 bits para o terceiro registrador (R3). A chave secreta desse algoritmo possui 64 bits e um vetor de inicialização de 22 bits. O *keystream* possui 228 bits, sendo que os primeiros 114 bits são a *keystream* de *downlink* ou seja, das ligações descendentes e os outros 114 bits são *keystream* de *uplink* que são ligações ascendentes (BIRYUKOV; SHAMIR, 2000).
- **Algoritmo A5/2:** Este algoritmo é semelhante ao A5/1 e foi desenvolvido por motivos de restrições de exportação. Nele são utilizados 4 registradores, de modo que o *clocking* dos registradores R1, R2 e R3 é acionado com base na regra de maioridade do quarto registrador (R4), ou seja, cada registrador é acionado dependendo do tamanho dos bits (BIRYUKOV; SHAMIR, 2000).
- **Algoritmo E0:** É utilizado para proteger informações que utilizam tecnologias *bluetooth*. O algoritmo é baseado no uso de registradores, e a saída é gerada em bits para cada registrador. Em sua estrutura são utilizados 4 registradores com tamanhos de 25, 31, 33 e 39 bits. O sistema da geração do *keystream* é um pouco diferente das demais, a cada *clock* todos os registradores são acionados e é feito um XOR[13] (*Exclusive OR*) da saída de cada um. Além de possuir 4 registradores, o sistema também é composto por mais dois registradores de 2 bits ct e ct-1, de forma que a cada *clock* todos os registradores são acionados e o valor de ct é transferido para ct-1, em seguida o valor de ct é atualizado com os valores do ct-1 e dos registradores (BIRYUKOV; SHAMIR, 2000).
- **Algoritmo RC4:** Segundo AlFardan et. al. (2013) este algoritmo é utilizado em sistemas de redes como TLS (*Transport Layer Security*), SSL e WEP (*Wired Equivalent Privacy*). Devido a sua simplicidade e alto desempenho, suas chaves

[12] segundo BAKHTIARI & MAAREF (2011) é uma operação lógica entre dois operandos em que os dois operandos são diferentes

[13] É uma operação lógica entre dois operandos que resulta em um valor lógico verdadeiro se e somente se os dois operandos forem diferentes, ou seja, se um for verdadeiro e o outro for falso.

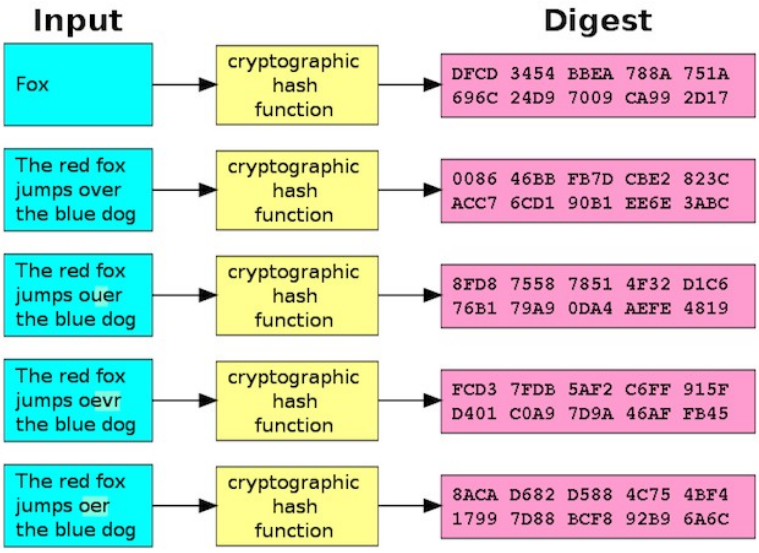
possuem tamanhos variáveis, e se baseiam em substituições aleatórias. Ela utiliza um gerador de números aleatórios e suas substituições são realizadas em cima dos números gerados.

2.6.3. Função Hash Criptográfica

Segundo Mendes e Enomoto (1999, p.3) uma função *hash* é uma função que a partir de uma entrada de tamanho variável, a converte em uma saída de tamanho fixo. A ideia base das funções *hash* é que o valor *hash* serve como uma representação compactada da *string* de entrada, e pode ser usado com a identificação única da *string*.

Ainda segundo Mendes e Enomoto (1999, p.5), as funções *hash* são classificadas de acordo com seu propósito, sendo elas:

Figura 5 - Funcionamento básico da criptografia Hash[14].



Fonte: Kaspersky (2014)

- **Códigos de Detecção de Manipulação**: Proporcionar junto a outros mecanismos, a integridade da mensagem.
- **Códigos de autenticação de mensagens**: Garantir a integridade tanto do emissor quanto do receptor a integridade dos dados.

2.6.4. Criptografias em Bancos de Dados

Segundo Gevioli e Seehagen (2009), as criptografias que que são encontradas no banco de dados são: cifras de blocos com tamanhos de bits de 56, 192, 256 e as criptografias *hash*, com tamanhos de bits de 128, 160, 224, 256, 384 e 512, e que cada banco de dados possui suas particularidades em criptografias, que podem ser iguais ou diferentes.

[14] Disponível em: https://www.kaspersky.com.br/blog/hash-o-que-sao-e-como-funcionam/2773/.

No Quadro 1 foram levantadas as criptografias utilizadas em cada banco de dados com base em suas documentações. Foram analisados os bancos de dados PostgreSQL (2020), Mysql (2020), MariaDB (2020), Oracle (2020), SQL Server (2017).

Quadro 1 – Criptografias nos bancos de dados

	BF	**MD5**	**DES**	**DESX**	**3DES168**	**SHA1**	**SHA2**	**AES128**	**AE192**	**AES 256**	**RC2**	**RC4**	**RSA**	**DSA**
PostgreSQL	X	X	X	X		X	X						X	X
MySQL		X	X			X	X	X					X	
MariaDB		X	X			X							X	
Oracle	X	X	X		X	X	X	X	X	X	X	X	X	X
SQL Server		X				X	X	X	X	X			X	

Fonte: O Autor.

3. METODOLOGIA

Na primeira etapa de categorização do *software*, foi realizada a análise sobre a finalidade do *software* CyclePlantões, onde conseguimos identificar se o mesmo seria um *software* aplicativo ou um *software* de sistema, e em seguida realiza uma análise dos requisitos do sistema CyclePlantões, visando suas funcionalidades, conseguindo assim realizar a categorização completa do *software*.

Na parte sequente a classificação do *software*, foi analisado a fundo seus requisitos onde foi realizado o levantamento de todos dados armazenados e trafegados dentro do *software* CyclePlantões, e em seguida, realizada o levantamento dos dados sensíveis segundo a Lei Geral de Proteção de Dados, com base nessa identificação de dados, foi realizado a escolha da criptografia ideal, para a finalidade do *software* e os dados que serão armazenados e trafegados dentro do sistema.

Na próxima etapa foi necessário realizar a instalação de extensão Postgres Anonymizer do banco de dados PostgreSQL para auxiliar na anonimação dos dados, em seguida foi escolhida as técnicas de anonimação baseando-se nos dados armazenados no banco de dados, visando escolher as funções e métodos de acordo com o ambiente de produção e ambiente de teste, sendo que no ambiente de produção a anonimação de dados deve elaborada com cuidado para não atrapalhar a finalidade do *software*, e no ambiente de teste, foi realizada a escolha de funções para substituição dos dados sensíveis.

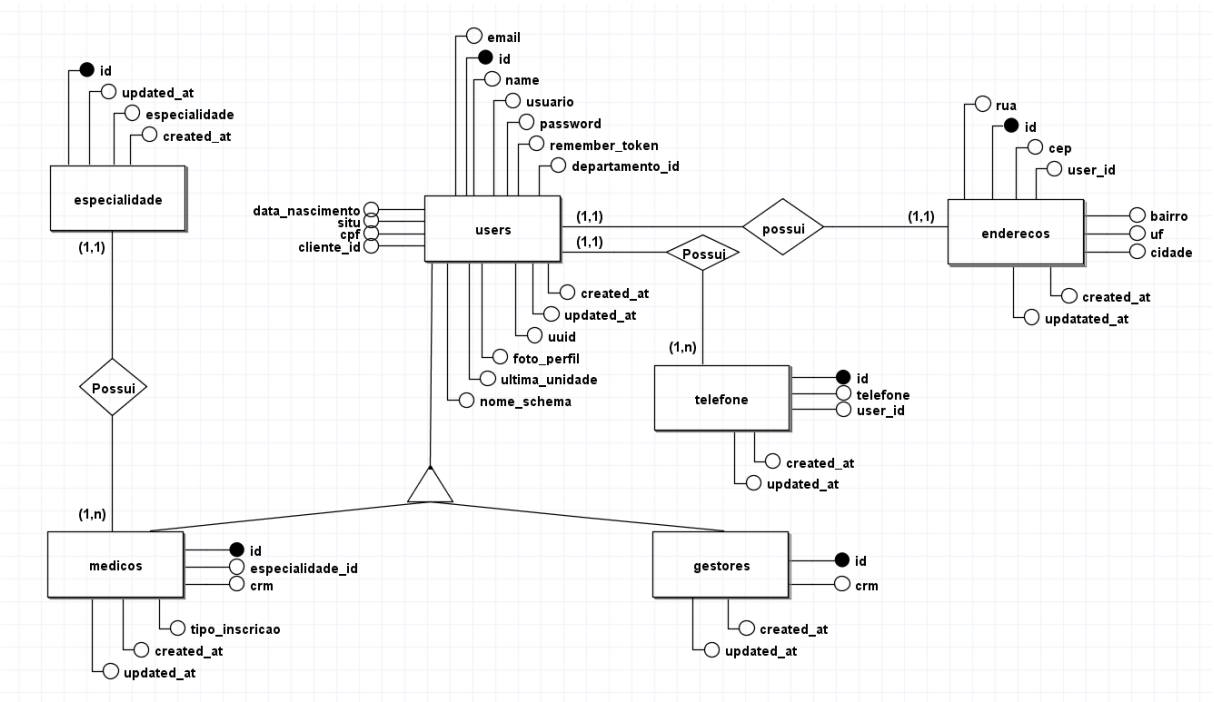
Na etapa de aplicação da metodologia de anonimação dados, foi necessário criar comandos SQL para o ambiente de teste quanto para o ambiente de produção, visando a as escolhas das funções e métodos da etapa anterior.

4. DESENVOLVIMENTO E ANÁLISE DE DADOS

O sistema analisado, CyclePlantões[15], é um gerenciador de plantões hospitalares que tem por finalidade armazenar, identificar e realizar trocas ou substituições de plantões médicos de forma ágil. Em conformidade com Stair e Reynolds (2011), ao analisar os requisitos do sistema CyclePlantões é possível distinguir o mesmo como sendo um *software* aplicativo, pois está sendo criado para um usuário final, e também como um *software* de gestão, pois ele possui um conjunto de ferramentas para serem utilizadas em tarefas relacionadas à administração e cálculo numérico.

Após analisar o Banco de Dados do sistema CyclePlantões, foi definido que as tabelas que serão utilizadas para anonimação e criptografia são: especialidade, médicos, users, gestores, telefone e enderecos. Elas são relacionadas entre si conforme apresentado na Figura 6.

Figura 6 - Modelo lógico das tabelas selecionadas no sistema CyclePlantões.



Fonte: Imagem desenvolvida para este trabalho (2020)

4.1. Levantamento de Dados, Informações e Dados Sensíveis

Para o levantamento dos dados e informações de um sistema, é necessário realizar uma análise de seus requisitos, onde poderá ser visível o sistema de forma abrangente e os dados que serão armazenados e trafegados dentro dele. Os dados sensíveis citados pela LGPD (2018), precisam ser armazenados com segurança e extremo sigilo.

[15] Disponível em: https://app.cycleplantoes.com.br/login.

4.1.1. Informações e Dados Sensíveis do Software Cycleplantões

Os dados sensíveis foram levantados após análise do DER (Diagrama de Entidade Relacional) no *software* CyclePlantões, e estão apresentados no Quadro 2.

Quadro 2 - Dados armazenados no banco de dados.

Tabela	Coluna	Tipo da coluna	Restrições
users	usuario	varchar (50)	Privado
users	senha	varchar (255)	Privado
users	name	varchar (255)	Publico
users	cpf	varchar (14)	Restrito para: o usuário e recursos humanos
users	data_nascimento	date	Restrito para: o usuário e recursos humanos
users	email	varchar (255)	Restrito para: o usuário e recursos humanos
telefones	telefone	varchar (25)	Restrito para: o usuário e recursos humanos
enderecos	cep	varchar (25)	Restrito para: o usuário e recursos humanos
enderecos	uf	char (2)	Restrito para: o usuário e recursos humanos
enderecos	cidade	varchar (50)	Restrito para: o usuário e recursos humanos
enderecos	bairro	varchar (50)	Restrito para: o usuário e recursos humanos
enderecos	rua	varchar (70)	Restrito para: o usuário e recursos humanos
especialidades	especialidade	varchar (60)	Restrito para: o usuário e recursos humanos
medicos	crm	varchar (60)	Restrito para: o usuário, recursos humanos e gestores
gestores	crm	varchar (60)	Restrito para: o usuário e recursos humanos

Fonte: Quadro desenvolvido para este trabalho (2020)

Após análises nos requisitos foram identificados que as seguintes informações serão trafegadas no sistema CyclePlantões: i) Jornada de trabalho; ii) Solicitações de trocas; iii) Solicitações de substituições.

A partir do levantamento destes dados e informações, é possível identificar quais são considerados dados sensíveis segundo a LGPD (2018), ou seja, aqueles que possibilitam a identificação do usuário. Os dados sensíveis identificados no CyclePlantões são os apresentados no Quadro 3:

Quadro 3 - Dados sensíveis do *software* CyclePlantões de acordo com a LGPD

Tabela	Coluna	Tipo da coluna	Restrições
users	usuario	varchar (50)	Privado
users	senha	varchar (255)	Privado
users	name	varchar (255)	Publico
users	cpf	varchar (14)	Restrito para: o usuário e recursos humanos
users	data_nasciment	date	Restrito para: o usuário e recursos humanos

	o		
users	email	varchar (255)	Restrito para: o usuário e recursos humanos
telefones	telefone	varchar (25)	Restrito para: o usuário e recursos humanos
enderecos	cep	varchar (25)	Restrito para: o usuário e recursos humanos
enderecos	uf	char (2)	Restrito para: o usuário e recursos humanos
enderecos	cidade	varchar (50)	Restrito para: o usuário e recursos humanos
enderecos	bairro	varchar (50)	Restrito para: o usuário e recursos humanos
enderecos	rua	varchar (70)	Restrito para: o usuário e recursos humanos
especialidades	especialidade	varchar (60)	Restrito para: o usuário e recursos humanos
medicos	crm	varchar (60)	Restrito para: o usuário, recursos humanos e gestores
gestores	crm	varchar (60)	Restrito para: o usuário e recursos humanos

Fonte: Quadro desenvolvido para este trabalho

4.2. Escolha da Criptografia para o Software Cycleplantões

Após a classificação do *software*, do levantamento de dados, da identificação dos dados sensíveis e da definição das informações trafegadas no sistema, é possível escolher a criptografia ideal para um sistema.

Na classificação do *software* é possível definir que é um *software web*, sendo assim se faz necessária a utilização do certificado SSL, que segundo a HostGator (2019) é utilizado para aumentar a segurança dos dados compartilhados na *web*, utilizando a criptografia de cifras de fluxo, ajudando a evitar a pratica do *cyber* crime, *phishing*, que utilizam o próprio site oficial da empresa, para copiar as informações trocadas durante uma transação, são copiadas todas as informações trafegadas inclusive dados de cartões de créditos e informações pessoais dos usuários.

Para a utilização do certificado SSL é necessário realizar a instalação do mesmo, e para isso é necessário acessar o painel administrativo da hospedagem onde o certificado se encontra. O sistema CyclePlantões utiliza o CPainel como painel administrativo do site.

- *1º passo:* Acesse a opção *Let's Encrypt* SSL, para que seja possível visualizar todos os domínios e subdomínios.
- *2º passo:* Selecione em qual domínio deseja instalar o certificado SSL clicando em *Issue*.
- *3º passo:* Selecione agora as opções que deseja instalar nos domínios. Podem ser instalados em seu domínio o *webmail* e o *web disk*. No sistema CyclePlantões será utilizado apenas o SSL para o domínio, pois este sistema não utiliza *softwares* de *webmail* e nem *web disk*.

4.2.1. Escolhendo a criptografia para as colunas no banco de dados.

O banco de dados utilizado pelo CyclePlantões, é o PostgreSQL, possuindo 4 tipos de criptografias para suas colunas. Para o *software* CyclePlantões, a criptografia a ser utilizada é a MD5, pois o tamanho da senha do usuário é definido por ele, e a criptografia MD5 não limita a quantidade máxima de caracteres. O sistema operacional onde será realizada a aplicação é o CentOS (linux), após abrir o terminal Linux os comandos para instalação da extensão são:

- *sudo yum install postgresql12-contrib*: Função utilizada para realizar o *download* e instalação do postgresql 12-contrib, que possui a função *pgcrypto*, utilizada para criptografar os campos.
- *create extension pgcrypto*: para utilização desta função é necessário acessar o banco de dados, ela tem por finalidade a criação da extensão *pgcrypto* para utilização de suas funções.

Segundo o PostgreSQL (2020) a finalidade da utilização da extensão *pgcrypto* para a *hash* MD5, é a dificuldade da utilização da força bruta para a identificação da senha, demorando de 1 dia até 3 anos variando de acordo com a complexidade da senha informada. Para a utilização da criptografia MD5 serão utilizadas as funções:

- *insert into [tabela] values (crypt('texto da senha',gen_salt('md5')))*: utilizada para inserir as senhas já criptografadas na base de dados.
- *SELECT [nome da coluna] = crypt('texto da senha', [nome da coluna]) FROM [nome da tabela]*: utilizado para realização da autenticação dos usuários.

4.3. Anonimizando os Dados Sensíveis do Software Cycleplantões

Para anonimação dos dados sensíveis do sistema CyclePlantões, será utilizada a extensão Postgresql anonymizer do banco de dados Postgresql, que possui técnicas que facilitam a anonimação sem a necessidade de utilização de outros *softwares*, mas segundos os desenvolvedores da extensão, é necessário utiliza-la com cuidado, pois a mesma se encontra ainda no estágio de desenvolvimento.

4.3.1. Instalando o PostgreSQL Anonymizer

Segundo PostgreSQL Anonymizer (2018), para que seja possível aplicar as técnicas de anonimação, é necessário realizar a instalação da extensão PostgresSQL Anonymizer, após abrir o terminal Linux os comandos para instalação da extensão são:

- *sudo yum install https://.../pgdg-redhat-repo-latest.noarch.rpm* este comando adiciona o repositório de *download* do PostgresSQL, e permite efetuar o *download* da extensão PostgreSQL anonymizer.
- *sudo yum install postgresql_anonymizer12* este comando efetua o *download* e a instalação da extensão do PostgreSQL anonymizer,

Após a execução dos comandos de instalação, deverá ser acessado o banco de dados para a execução dos comandos de carreamento da extensão, criação e carreamento dos dados de anonimato e inicialização da extensão.

- *ALTER DATABASE [nome da sua base de dados] SET session_preload_libraries = 'anon';* este comando realizará o carregamento da extensão PostgreSQL Anonymizer.
- *CREATE EXTENSION anon CASCADE;* este comando irá criar e carregar os dados de anonimato.
- *SELECT anon.init();* este comando inicializará a extensão na base de dados.

4.3.2. Escolha das Técnicas de Anonimizar Dados

Segundo a LGPD (2018) é de extrema importância que apenas os usuários legítimos tenham acesso as suas informações, sendo assim, serão aplicadas técnicas para anonimizar estes dados no ambiente de teste, pois é neste ambiente onde será utilizada uma cópia do banco de dados do ambiente de produção para a realização de testes. Para o banco de produção serão utilizadas criptografias para evitar acesso não autorizado de terceiros no banco de dados, e a utilização de mascaramento parcial dos dados.

Para escolher quais funções utilizar, é necessário analisar os tipos dos dados armazenados e verificar qual função se encaixa com o tipo do dado. No ambiente de teste do *software* CyclePlantões, as técnicas utilizadas para anonimar os dados são: i) *randomization*, para gerar dados aleatórios de datas, números inteiros e telefones; e ii) *faking* para gerar dados aleatórios de nomes, endereços e *e-mails* e no ambiente de produção será utilizada apenas a função de anonimação parcial, pois devido a sua finalidade é necessária a identificação dos usuários plantonistas no ambiente de produção. É possível que para algum dado seja necessário utilizar uma junção de uma ou mais funções, isso é possível utilizando o comando de concatenação ‘||’ entre as funções.

No ambiente de produção do *software* CyclePlantões, será utilizada a função para anonimar parcialmente o dado CPF dos usuários em geral, evitando a visualização do usuário recursos humanos após o cadastro dos usuários em geral.

- CPF: *anon.partial('[campo CPF]',[números de casas sem máscara no início],'xxxx', [números de casas sem máscara no fim]);* esta função será utilizada para mascarar parcialmente ou totalmente o CRM do médico.

No *Software* CyclePlantões, serão utilizados os seguintes comandos para cada tipo de dado no ambiente de teste:

- Dados de acesso:
    - Usuário: *anon.fake_first_name()* ||'.'|| *anon.fake_last_name()* pois os nomes de usuários existentes no banco possuem dois nomes separados por um '.'
- Dados pessoais:
    - Nome: *anon.fake_firt_name()* || ' ' || *anon.fake_last_name()* pois os nomes armazenados possuem nome e sobrenome.
    - CPF: *anon.random_bigint_between()* o CPF são 11 dígitos, sendo necessário utilizar a função de *bigint*.
    - Data de nascimento: *anon.random_date_between()* será utilizada a função com *between* para colocar uma data mínima e uma data máxima.
    - E-mail: *anon.fake_email()* para gerar e-mails falsos.
    - Telefones: *anon.fake_phone* utilizado para gerar telefones aleatórios.
- Dados de moradia:
    - CEP: *anon.random_int_between()* para gerar os dados de tamanho inteiro com 8 dígitos.
    - Rua: *anon.fake_last_name()* || ' ' || *anon.random_int_between()*
    - UF: *anon.lorem_ipsum(characters:=)* para gerar dois caracteres simulando o UF.
    - Cidade: *anon.fake_city()* para gerar cidades aleatórias na base de dados.

- Bairro: *anon.fake_first_name()* utilizaremos para gerar nomes de bairros, deixando assim mais próximo a realidade, pois existem diversos bairros com nomes de pessoas.

- Dados profissionais:
  - CRM: *anon.random_int_between()||'/'|| anon.lorem_ipsum(characters:=)* os dados de CRM possuem duas letras que identificam um estado do país, então utilizaremos esta função para gerar as duas letras finais.

4.3.3. Aplicação do Método de Anonimação

Após a definição das funções que serão utilizadas, é necessário desenvolver um comando SQL, para a substituição dos dados, a função SQL a ser utilizada é o *update*, que tem a finalidade de substituir os dados das colunas na base de dados, neste comando serão informadas as funções, as tabelas e as colunas que serão anonimadas.

Será desenvolvido um SQL para o ambiente de produção onde será anonimado apenas o campo CPF da tabela *users* para o usuário dos recursos humanos foi desenvolvido um comando para cada tabela no banco de dados do CyclePlantões no ambiente de produção, sendo um para a tabela *users*, um para a tabela de médicos, um para a tabela de gestores e outro para a tabela de endereços.

4.3.3.1. Ambiente de Produção:

```
SELECT anon.partial(u.cpf,2,$$********$$,2),* FROM producao.users u
        INNER JOIN producao.medicos med ON u.id = med.id
        INNER JOIN producao.enderecos ed ON ed.id = u.id
INNER JOIN producao.especialidades esp ON esp.especialidade_id =
        med.especialidade_id
INNER JOIN producao.telefones tel ON tel.id = u.id
INNER JOIN producao.departamentos dep ON u.departamento_id =
        dep.departamento_id;
```

4.3.3.2. Ambiente de Teste:

- Comando para a tabela users:

```
update teste.users
set name = anon.random_first_name()|| ' ' || anon.random_last_name(),
   email = anon.fake_email(),
   usuario = anon.random_first_name()|| '.' || anon.random_last_name(),
   data_nascimento= anon.random_date_between('1950/01/01','2000/12/12'),
```

```
    cpf = anon.random_bigint_between (11111111111,99999999999);
```

- Comando para a tabela de médicos:

```
update teste.medicos set crm = anon.random_int_between(11111,99999) ||'/'||
anon.lorem_ipsum(characters := 2);
```

- Comando para a tabela de gestores:

```
update teste.gestores set crm = anon.random_int_between(11111,99999)
||'/'|| anon.lorem_ipsum(characters := 2);
```

- Comando para a tabela de endereços:

```
update teste.enderecos
set cep = anon.random_int_between(11111111,99999999),
     rua = anon.random_last_name () || ' ' || anon.random_int_between(1,99),
     uf = anon.lorem_ipsum(characters := 2),
     cidade = anon.random_city(),
     bairro = anon.random_last_name();
```

Executando os comandos acima, os dados que podem identificar um usuário ficariam substituídos por dados aleatórios, tornando assim impossível identificá-lo. Desta forma, a implantação da nova lei de proteção de dados LGPD (2018) na base de dados seria realizada com êxito.

5. CONSIDERAÇÕES FINAIS

Este trabalho teve como objetivo auxiliar na categorização do *software* e levantamento de dados sensíveis para a utilização da criptografia correta e a utilização da anonimização de dados por meio de funções disponibilizadas pelo sistema de gerenciamento de banco de dados.

Foram realizadas análises na finalidade e nos requisitos do sistema CyclePlantões, identificando em qual quadrante de *software*s o mesmo se encaixa. Foram utilizados também os requisitos do *software* para levantamento dos dados armazenados e trafegados no sistema, e após o levantamento de todos estes dados, foram categorizados todos os dados considerados como sensíveis pela LGPD (2018).

A utilização da extensão PostgresSQL anonymizer auxiliou na anonimação dos dados no ambiente de teste e produção, pois foi desenvolvida para a anonimação de dados tornando-os indetectáveis, e com a análise e categorização dos dados sensíveis segundo a LGPD (2018) foi possível identificar quais funções serão utilizadas para cada tipo de dado.

Após a identificação das funções foi possível desenvolver comandos SQL, para a substituição e visualização dos dados de forma anonimada por meio da junção das funções da extensão PostgresSQL anonymizer e com os comandos SQL do PostgresSQL.

REFERÊNCIAS BIBLIOGRÁFICAS

AGRELA, Lucas. O escândalo de vazamento de dados do Facebook é muito pior do que parecia. Revista Exame. 06 de abril de 2018. Disponível em: <https://exame.abril.com.br/tecnologia/o-escandalo-de-vazamento-de-dados-do-facebook-e-muito-pior-do-que-parecia/>. Acesso em: 19 de março de 2020.

ALFARDAN, Nadhem J. et al. On the Security of RC4 in TLS and WPA. 2013. Disponível em: <http://www.isg.rhul.ac.uk/tls/RC4biases.pdf>. Acesso em: 8 out. 2020.

AMORIM, Diego Felipe Borges. Softwares de sistemas e aplicações livres: Benefícios e limitações no uso dessas tecnologias nos negócios. 2015. Disponível em: < https://www.researchgate.net/publication/307924382_SOFTWARES_DE_SISTEMAS_E_DE_APLICACOES_LIVRES_BENEFICIOS_E_LIMITACOES_NO_USO_DESSAS_TECNOLOGIAS_NOS_NEGOCIOS/link/5c2a3908458515a4c7039da4/download>. Acesso em: 24 out. 2020.

AWS Amazon. Mecanismos de BDs populares. 2018. Disponível em: < https://aws.amazon.com/pt/rds/?trk=ps_a131L0000083bBMQAY&trkCampaign=pac_ps_Q1_120_RDS_PDP_P_NBrand_BR&sc_channel=ps&sc_campaign=pac_q1-1-2020_paidsearch_RDS_OpenSource_BR&sc_outcome=PaaS_Digital_Marketing&sc_geo=LATAM&sc_country=BR&sc_publisher=Google&sc_category=Database&sc_detail=%2Bbancos%20%2Bde%20%2Bdados&sc_content=database_bmm&sc_matchtype=b&sc_segment=436567689819&sc_medium=PAC-PaaS-P|PS-GO|Non-Brand|Desktop|PA|Database|RDS|BR|PT|Text&s_kwcid=AL!4422!3!436567689819!b!!g!!%2Bbancos%20%2Bde%20%2Bdados&ef_id=Cj0KCQjw8rT8BRCbARIsALWiOvSUI3lHmuHB4qVd82OXCIY8DLcYtrmTEog6Snsf_x1O0lOMWiyQF4UaAu83EALw_wcB:G:s&s_kwcid=AL!4422!3!436567689819!b!!g!!%2Bbancos%20%2Bde%20%2Bdados>. Acesso em: 19 out. 2020.

BAKHTIARI, M.; MAAREF, M. A. An efficient stream cipher algorithm for data encryption. University Technology Malaysia, 2011. Citado na página 27.

BBC, Entenda o escândalo de uso político de dados que derrubou valor do Facebook e o colocou na mira de autoridades. G1, 10/03/2018. Disponível em: https://g1.globo.com/economia/tecnologia/noticia/entenda-o-escandalo-de-uso-politico-de-dados-que-derrubou-valor-do-facebook-e-o-colocou-na-mira-de-autoridades.ghtml>. Acesso em: 12 de fev. de 2020.

BIRYUKOV, A. Block ciphers and stream ciphers: The state of the art. Katholieke Universiteit Leuven. Citado na página 22.

BRASIL, LEI Nº 13.853 DE 8 DE JULHO DE 2019. Lei Geral de Proteção de Dados (LGPD). BrasíliaDF, Ago 2020. Disponivel em: <http://www.planalto.gov.br/ccivil_03/_Ato2019-2022/2019/Lei/L13853.htm#art1>

CANALTECH. Internet alcança 74% dos brasileiros e 58% utilizam a rede apenas pelo celular. São Paulo, 2020. Disponível em: < https://canaltech.com.br/internet/internet-alcanca-74-dos-brasileiros-e-58-utilizam-a-rede-apenas-pelo-celular-165851/>, Acesso em: 07 de dezembro de 2020.

CARDOZO, Eleri; MAGALHÃES, Maurício F.. Introdução aos Sistemas Operacionais. São Paulo, 2002. Disponível em: <http://www.dca.fee.unicamp.br/~eleri/ea876/02/so-apst.pdf>, Acesso em: 07 de Outubro de 2020.

CIRIACO, Douglas. O que é criptografia e por que você deveria usá-la. Disponível em: < https://canaltech.com.br/seguranca/o-que-e-criptografia-e-por-que-voce-deveria-usa-la/ >. Acesso em: 13 de fev. de 2020.

E-COMMERCE BRASIL. Faturamento do e-commerce brasileiro bate a marca de R$ 41,92 bilhões. Disponível em: < https://www.ecommercebrasil.com.br/noticias/faturamento-do-e-commerce-brasileiro-2020/#:~:text=O%20ano%20de%202020%20já,crescimento%20não%20param%20de%20subir.&text=Com%20esse%20crescimento%2C%20a%20projeção,para%2030%25%20no%20acumulado%20anual.>. Acesso em: 13 de fev. de 2020.

GONZÁLES, Daniela. Conheça os tipos de criptografia digital mais utilizados

HSC Brasil, Conheça 6 grandes empresas invadidas por hackers. HSC High Security Center, 22 de jan. de 2019. Disponível em: < https://www.hscbrasil.com.br/grandes-empresas-invadidas-por-hackers/ >. Acesso em: 13 de fev. de 2020.

KASPERSKY. O que é um ataque de força bruta?. Disponível em: < https://www.kaspersky.com.br/resource-center/definitions/brute-force-attack>. Acesso em: 08 de out. de 2020.

KASPERSKY. O que são ataques de DDoS?. 2016. Disponível em: < https://www.kaspersky.com.br/resource-center/threats/ddos-attacks>. Acesso em: 08 de out. de 2020.

LAMBERT, JORGE DE ALBUQUERQUE. CIFRADOR SIMÉTRICO DE BLOCOS: LÜDKE, Menga; ANDRÉ, Marli E. D. A. Pesquisa em Educação: Abordagens Qualitativas. Editora Pedagógica e Universitária LTDA.

MALWAREBYTES. Tudo sobre Hacking. Disponível em: <https://br.malwarebytes.com/hacker/>. Acesso em: 04 de nov. de 2020.

MariaDB. Encrypt. 2020. Disponível em: < https://mariadb.com/kb/en/encrypt/>. Acesso em 04 de nov de 2020.

MENDES, Fabiana Valença; ENOMOTO, Cristina. Funções Hash. Disponível em: < ftp://vm1-dca.fee.unicamp.br/pub/docs/marco/disciplinas/ia364_99_1/monografias/hash.pdf.gz>. Acesso em: 19 de out. de 2020.

MORAIS, Edilson. Por que sua empresa vai migrar para o armazenamento em nuvem. Conta Azul Blog. 05 de Maio de 2017. Disponível em: <https://blog.contaazul.com/lgpd-lei-geral-protecao-dados-pessoais/ >. Acesso em: 12 de fev. de 2020.

MORENO, Edward David. Criptografia em *Software* e Hardware. Novatec Editora, 2005. Disponível em:<http://www.martinsfontespaulista.com.br/anexos/produtos/capitulos/143116.pdf>. Acesso em: 16 de fev. de 2020.

MySQL. 12.14 Encryption and compression functions, 2020. Disponível em:< https://dev.mysql.com/doc/refman/8.0/en/encryption-functions.html>. Acesso em: 04 de nov. de 2020.

NIELD, Thomas. Introdução à Linguagem SQL: Abordagem prática para iniciantes. Novatec. São Paulo, 2019. Disponível em < http://www.nrsystem.com.br/SQL.pdf>. Acesso em: 11 de nov. de 2020.

Oracle. 3 Oracle Encrypto. 2020. Disponível em:< https://docs.oracle.com/cd/E23943_01/security.1111/e10037/crypto.htm#SDTRG002>. Acesso em: 04 de nov. de 2020.

PINTO, Pedro. Conheça a história da criptografia. PPLware, 13 de março de 2013. Disponível em: <https://pplware.sapo.pt/informacao/conhea-a-historia-da-criptografia/>. Acesso em: 16 de fev. de 2020.

PORTAL IG. Criptografia de ponta a ponta protege conversas no Whatsapp; mas como funciona? Disponível em: < https://tecnologia.ig.com.br/2019-06-17/criptografia-de-ponta-a-ponta-whatsapp.html >. Acesso em: 04 de nov. de 2020.

PostgreSQL. Várias estratégias de mascaramento. 2018. Disponível em: < https://postgresql-anonymizer.readthedocs.io/en/latest/masking_functions/#write-your-own-masks>. Acesso em: 19 de out. de 2020.

PostgresSQL. F.20. Pgcrypto. 2020. Disponível em: < https://www.postgresql.org/docs/8.3/pgcrypto.html>. Acesso em: 04 de nov. de 2020.

PROJETO E AVALIAÇÃO. Laboratório de Pesquisa Cibernética. Rio de Janeiro, 2004. Disponível em: < http://www.defesacibernetica.ime.eb.br/pub/repositorio/2004-Jorge_Lambert.pdf>. Acesso em: 16 de fev. de 2020.

Rocketcontent. Descubra quais os tipos de software existentes e como cada um deles funciona. 2020. Disponível em: < https://rockcontent.com/br/blog/tipos-de-software/>. Acesso em: 04 de nov. de 2020.

ROSA, Natalie. Quora é invadido e dados de 100 milhões de usuários são expostos. 04 de dezembro de 2018. Disponível em: <https://canaltech.com.br/hacker/quora-e-invadido-e-dados-de-100-milhoes-de-usuarios-sao-expostos-128293/>. Acesso em: 28 de mai. de 2020.

SEMIDÃO, Rafael Aparecido Moron. DADOS, INFORMAÇÃO E CONHECIMENTO ENQUANTO ELEMENTOS DE COMPREENSÃO DO UNIVERSO CONCEITUAL DA CIÊNCIA DA INFORMAÇÃO: CONTRIBUIÇÕES TEÓRICAS. São Paulo: Marília, 2014. Disponível em: <https://www.marilia.unesp.br/Home/Pos-Graduacao/CienciadaInformacao/Dissertacoes/semidao_ram_me_mar.pdf>, Acesso em: 07 de outubro de 2020.

SOMMERVILLE, Ian. Engenharia de *Software*. 9. ed. São Paulo: Pearson, 2011. Disponível em: <http://www.facom.ufu.br/~william/Disciplinas%202018-2/BSI-GSI030-EngenhariaSoftware/Livro/engenhariaSoftwareSommerville.pdf>. Acesso em: 12 out. 2020.

SOUSA, Antonio Nilson Laurindo. Criptografia de Chave Pública, Criptografia RSA. Disponível em: < https://repositorio.unesp.br/bitstream/handle/11449/94349/sousa_anl_me_rcla.pdf;jsessionid=AA6671992159ABCBDCE754D09E5ED624?sequence=1>, Acesso em: 04 de nov de 2020.

SOUSA, Bruno Jefferson; JÚNIO, José Jorge Lima Dias; FORMIGA, Andrei de Araújo. Introdução Programação. São Paulo: Editora UFPB, 2014. Disponível em: <https://cotemar.com.br/wp-content/uploads/2019/10/introducao_a_programacao_compressed.pdf>, Acesso em: 07 de Outubro de 2020.

SOUSA, Flávio R. C.; MOREIRA, Leonardo.; MACHADO, Jarvam C.. Computação em Nuvem: Conceitos, Tecnologias, Aplicações e Desafios. Ceará, 2010. Disponível em: <https://www.researchgate.net/profile/Javam_Machado/publication/237644729_Computacao_em_Nuvem_Conceitos_Tecnologias_Aplicacoes_e_Desafios/links/56044f4308aea25fce3121f3.pdf>, Acesso em: 08 de Outubro de 2020.

SQL Server. Criptografia do SQL Server. 2017. Disponível em: < https://docs.microsoft.com/pt-br/sql/relational-databases/security/encryption/sql-server-encryption?view=sql-server-ver15>. Acesso em: 04 de nov. de 2020.

STAIR, Ralph M.; REYNOLDS, George W. Princípios de Sistemas de Informação ed 11. Disponível em: <https://docplayer.com.br/71446211-Principios-de-sistemas-de-informacao.html>, Acesso em: 10 de Outubro de 2020.

STALLINGS, William; BROWN, Lawrie. Segurança de Computadores 2.ed. Elsevier Editora Ltda, Rio de Janeiro, 2014, Acesso em: 19 de abril de 2020.

TAKAI, Osvaldo Kotaro; ITALIANO, Isabel Cristina; FERREIRA, João Eduardo. Introdução a banco de dados 2005. Disponível em: < https://www.ime.usp.br/~jef/apostila.pdf>, Acesso em: 04 de nov de 2020.

Tecnologia da informação - Técnicas de segurança - Código de prática para gestão da segurança da informação (ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS, 2005, p. 10).

TOBACE, Ewerthon. Hacker acessou dados de 77 milhões de usuários de PlayStation, diz Sony. BBC News Brasil. 27 de abril de 2011. Disponível em: < https://www.bbc.com/portuguese/noticias/2011/04/110427_sony_hacker_bg>. Acesso em: 16 de fev. de 2020.

UNISYS, Brasil é o país com maior crescimento em preocupações com segurança no mundo, mostra estudos da UNISYS. Disponível em: < https://www.unisys.com.br/offerings/security-solutions/news%20release/br-brasil-e-o-pais-com-maior-crescimento-em-preocupacoes-com-segurança>. Acesso em 20 de agosto de 2020.

VALENTE, Jonas, Preocupação com segurança de dados é a maior em 10 anos. Agência Brasil. 19 de junho de 2019. Disponível em: < https://agenciabrasil.ebc.com.br/geral/noticia/2019-06/preocupacao-com-seguranca-de-dados-e-maior-em-10-anos>. Acesso em 28 de maio de 2020.

Yin, Robert K. Estudo de Caso Planejamento e Métodos. Bookman, 5ª Edição, 27 de outubro de 2014.

Comprovante de Assinatura Eletrônica

contraktor

Datas e horários baseados no fuso horário (GMT -3:00) em Brasília, Brasil
Sincronizado com o NTP.br e Observatório Nacional (ON)
Certificado de assinatura gerado em 11/12/2020 às 23:11:38 (GMT -3:00)

TCC II - Lucas Henrique de Moura e Silva_2020.12.11.docx

ID única do documento: #4d987f70-0dbd-42ef-a485-8e53e4e824e1

Hash do documento original (SHA256): 9d78867a9f8fd48dfbbec02f9e327b321c0d69c4ee128741529609a01a7d5a85

Este Log é exclusivo ao documento número #4d987f70-0dbd-42ef-a485-8e53e4e824e1 e deve ser considerado parte do mesmo, com os efeitos prescritos nos Termos de Uso.

Assinaturas (2)

Lucas Henrique de Moura e Silva (Participante)
Assinou em 11/12/2020 às 23:13:47 (GMT -3:00)

William Pereira dos Santos Júnior (Participante)
Assinou em 11/12/2020 às 23:13:03 (GMT -3:00)

Histórico completo

Data e hora	Evento
11/12/2020 às 23:13:47 (GMT -3:00)	Lucas Henrique de Moura e Silva (Autenticação: e-mail lucashmsilva1@gmail.com; IP: 189.123.60.110) assinou. Autenticidade deste documento poderá ser verificada em [[LINK]]. Assinatura com validade jurídica conforme MP 2.200-2/01, Art. 10o, §2.
11/12/2020 às 23:11:38 (GMT -3:00)	Lucas Henrique de Moura e Silva solicitou as assinaturas.
11/12/2020 às 23:13:03 (GMT -3:00)	William Pereira dos Santos Júnior (Autenticação: e-mail williampsjuniortn@hotmail.com; IP: 143.0.255.112) assinou. Autenticidade deste documento poderá ser verificada em [[LINK]]. Assinatura com validade jurídica conforme MP 2.200-2/01, Art. 10o, §2.