UMA PROPOSTA DE SOLUÇÃO PARA A PROTEÇÃO DAS INFORMAÇÕES DE CONEXÃO AO BANCO DE DADOS

A PROPOSED SOLUTION FOR PROTECTING DATABASE CONNECTION INFORMATION

Gustavo Silva Souza, Fatec Santana de Parnaíba,
gustavosouza.pro417@gmail.com

Lucas Mateus Braga de Souza, Fatec Santana de Parnaíba,
lucasbrouza@gmail.com

Irapuan Glória Júnior, Fatec Santana de Parnaíba,
ijunior@ndsgn.com.br

Resumo

Os bancos de dados são primordiais em qualquer empresa para armazenar as informações geradas. Dessa forma, é indispensável que tenham um ambiente seguro. Devido a isso, as organizações impõem políticas de segurança da informação por zelo aos seus dados, além de outras medidas de segurança, como as voltadas à cadeia de caracteres de conexão, a qual é um dos pontos mais sensíveis de um sistema de informação e que exige cuidados especiais. Este artigo, de natureza qualitativa, utilizou a metodologia *Design Science Research* a fim de propor uma solução para a proteção da string de conexão por meio de técnicas de criptografia. O resultado foi uma sugestão computacional que protege a cadeia de caracteres de conexão por meio de encriptação e que, além disso, mantém a *string* de conexão em sigilo. A contribuição para a teoria é apresentar formas de utilização de criptografia e arquitetura de sistemas no campo da segurança da informação. A contribuição para a prática é apresentar uma sugestão computacional que possa auxiliar na mitigação de ataques à cadeia de caracteres de conexão.

Palavras-chave: *string* de conexão, banco de dados, criptografia, cibersegurança, segurança da informação.

Abstract

The database is primordially in any company to save all generated data. So, it is important to have a secure ambient. Because of that, organizations establish security policies to protect this data, including others security protocols, such as those focused on database connection string, which is one of the most sensitive points in information system and require special cautions. This article, which is qualitative in nature, has been used the concepts of the Design Science Research methodologies to propose a solution to protecting the database connection string through cryptography techniques. The result was a computational suggestion that protects the database connection string using cryptography and keeps the database connection string secret. The contribution to theory is to present ways of using cryptography and systems architecture in information security segment. The contribution to practice is to present a computational suggestion that can help mitigate attacks on the database connection string.

Keywords*: connection string, database, cryptography, cibersecurity, information security.*

1. Introdução

Os bancos de dados são essenciais às organizações, já que por causa deles, elas obtêm informações importantes que são decisivas no processo de tomadas de decisão em seus negócios (Sharda, 2019).

Assim, devido a tal valor, ele se torna um alvo atrativo para ataques cibernéticos, seja por interesses financeiros, informacionais ou apenas pela vontade de prejudicar a organização (Fraga, 2019). Por isso, as empresas impõem políticas de segurança da informação com o intuito de protegerem seus dados (Hintzbergen, 2018), além de tomarem precauções em relação a outros fatores importantes, como a cadeia de caracteres de conexão ao banco de dados.

A cadeia de caracteres de conexão ao banco de dados é responsável por mediar o acesso ao banco de dados que contém dados sigilosos, como usuário, senha, nome do banco de dados, endereço do servidor, e podem ser alvos de ataques cibernéticos (Elmasri, 2019). Desta forma, proteções adicionais são necessárias para proteger essas informações (First, 2023).

Diante desse contexto, este artigo buscou uma forma de proteção a mais no momento de conexão para sistemas informáticos que utilizam bancos de dados, a qual atua especificamente na proteção da cadeia de caracteres de conexão ao banco de dados. Portanto, os objetivos são: (1) identificar o ambiente do sistema da Empresa-X que requisitou a implementação de segurança; e (2) propor uma solução computacional para a proteção da cadeia de caracteres de conexão ao banco de dados.

2. Referencial Teórico

2.1. Ambiente de Sistemas

Um ambiente de sistema é a plataforma na qual uma aplicação opera, como os ambientes *web*, *desktop* e *mobile* (Pressman, 2021). O ambiente *web* é uma estrutura arquitetônica que possibilita o acesso a documentos disponibilizados na *internet*, consiste em um ambiente de sistema que pode ser acessado a partir de qualquer dispositivo inteligente que possua algum navegador (Tanenbaum, 2021). Os ambientes *desktop* e *mobile* compartilham com processamento local, mas suporta diferentes linguagens e seus programas devem ser instalados (Pressman, 2021).

Independentemente do ambiente o uso de sistemas gerenciamento de Banco de Dados está presente como uma forma de armazenamento temporário, como no ambiente *mobile,* ou permanente como nos casos de sistemas desenvolvidos em *desktops* e *web* (Sommerville, 2018).

2.2. Sistema Gerenciador de Banco de Dados

O banco de dados é um conjunto de dados relacionados, que pode ter qualquer tamanho e complexidade e que costuma ser administrado por um sistema de gerenciamento de banco de dados (SGBD), como o MySQL, Oracle e o MS SQL Server (Elmasri, 2019).

O SGBD é um conjunto de programas que permitem a criação e o mantimento de um banco de dados por um longo período de tempo, que atua na manutenção da estrutura de banco de dados, na segurança dele e nas funcionalidades de tratamentos de dados, como as operações de inserção/insert, leitura/read, atualização/update e exclusão/delete de dados, que formam o acrônimo CRUD.

Existem muitos modelos de SGBD, que contemplam projetos e necessidades diferentes, como os (1) SGBDs relacionais, que representam dados por meio de tabelas; (2) os SGBDs não relacionais, que trabalham com grafos e (3) os SGBDs orientados a objetos, que operam com bancos de dados a partir de objetos.

Devido ao SGBD, também é possível acessar o banco de dados, já que ele fornece o uso da cadeia de caracteres de conexão/*string* de conexão, que é uma ferramenta mediadora entre uma aplicação e o banco de dados, que solicita informações de autenticação, endereço e medidas de segurança para o estabelecimento de uma conexão entre ambos que, após definida, dá acesso total a determinado banco de dados, desde leituras de dados até a exclusão completa dele.

2.3. Criptografia

A criptografia é a ciência que estuda técnicas de comunicação sigilosa, que são úteis para a proteção de informações confidenciais em ambiente computacional, como as encriptações simétricas e assimétricas e o *hash* (Tanenbaum, 2021).

Existem diversos tipos de criptografia (Tabela 1), em relação à encriptação simétrica, há a cifragem de informações a partir de uma chave única, essa que também pode ser usada para decriptar as informações, como ocorre nas técnicas de encriptação designadas *Advanced*

Encryption Standard (AES) e *Data Encryption Standard* (DES) (Elmasri, 2019).

No caso da encriptação assimétrica, as informações são cifradas por meio de duas chaves, uma pública e outra privada: quando a cifra é feita com a chave pública, a decifra é operada pela chave privada; quando a cifra é feita com a chave privada, a decifra é operada pela chave pública. Alguns exemplos de tal algoritmo são o *Rivest-Shamir-Adleman* (RSA) e o *Elliptic Curve Cryptography* (ECC) (Elmasri, 2019; Tanenbaum, 2021)).

Em alguns casos, não é conveniente usar uma técnica de criptografia que permita decriptação e, para atender essa demanda, é utilizada a criptografia de *hash*, que gera um código irreversível de tamanho fixo, como é feito pelas técnicas de *hash Secure Hash Algorithm* (SHA) e *Message-Digest Algorithm 5* (MD5) (Tanenbaum, 2021).

Tabela 1 – Nomes e Siglas dos Termos de Criptografia

#	Termo	Autor
01	*Advanced Encryption Standard* (AES)	Elmasri (2019)
02	*Data Encryption Standard* (DES)	Elmasri (2019)
03	*Rivest-Shamir-Adleman* (RSA)	Elmasri (2019)
04	*Elliptic Curve Cryptography* (ECC)	Elmasri (2019)
05	*Secure Hash Algorithm* (SHA)	Tanenbaum (2021)
06	*Message-Digest Algorithm 5* (MD5)	Tanenbaum (2021)

2.4. Segurança da Informação

A segurança da informação é a proteção da informação em relação a uma ampla gama de ameaças, isso por meio da confidencialidade, integridade e disponibilidade, que formam o acrônimo CID (Hintzbergen, 2018).

A confidencialidade é relativa à privacidade da informação, na qual a informação deve ser divulgada e exposta apenas para pessoas, processos ou entidades que tenham autorização para acessá-la (Freund, 2019).

Em relação à integridade, há um vínculo quanto à autenticidade da informação, em que a informação deve se manter íntegra e fidedigna, sem sofrer alterações não autorizadas pelo seu proprietário (Freund, 2019).

O último item, a disponibilidade, refere-se à obrigação da informação sempre estar disponível para quem tem direito de acesso a ela, e isso implica no dever do sistema de ter capacidade o suficiente para suportar o uso simultâneo de toda a equipe, como também de ter manutenções imediatas caso ocorram falhas que violem a disponibilidade dele (Hintzbergen, 2018).

Além das diretrizes estabelecidas pelo CID, no Brasil, também há a Lei Geral de Proteção de Dados (LGPD), que foi fundada em 2018 com base no Regulamento de Proteção de Dados (GDPR), a qual está vigente desde 2020 (Rapôso, 2020) e que dispõe do tratamento de dados pessoais, tanto em ambiente físico quanto digital, seja pessoa física ou jurídica, com a finalidade de prover privacidade e liberdade aos proprietários dos dados (Brasil, 2018).

De acordo com a LGPD, (1) titular é a pessoa natural a qual o dado pertence, (2) dado pessoal é aquele que identifica um titular, (3) dado sensível é aquele que compromete a privacidade do titular caso exposto, (4) controlador é quem manipula dados, (5) operador é toda pessoa física ou jurídica que participa de um tratamento de dados, (6) encarregado é aquele que faz ponte de comunicação entre titular, controlador e autoridade e (7) tratamento de dados pessoais é qualquer operação que englobe dados pessoais, seja coleta, compartilhamento, exclusão ou outros (Brasil, 2018).

A lei regulamenta o uso de dados do titular por terceiros, que só podem efetuar tratamento de dados de titulares caso haja autorização e enquanto o tratamento de dados for necessário, pois caso o prazo de utilização finalize, ocorra violação de dados e/ou o titular solicite pelo fim do tratamento, ele deve ser concluído e, em caso de algum transtorno, a Autoridade Nacional de Proteção de Dados (ANPD) se encarrega de sentenciar punições.

3. Metodologia

A pesquisa possui natureza qualitativa, utilizou a metodologia *Design Science Research*, que tem o objetivo de desenvolver soluções práticas, baseadas em tecnologia, para problemas (Nunamaker Jr et al., 1990). A coleta de dados foi realizada por meio de documentos digitais e informações provenientes da Empresa-X, como levantamento de dados e documentos

de arquitetura de sistemas (Tabela 1).

Tabela 1 - Aspectos Metodológicos

Itens	Conteúdo	Autores
Natureza	Qualitativa	Theóphilo e Martins (2016)
Metodologia	*Design Science Research*	Nunamaker Jr et al. (1990)
Coleta de dados	Documentos digitais e informações provenientes da Empresa-X	

Fonte: Os Autores

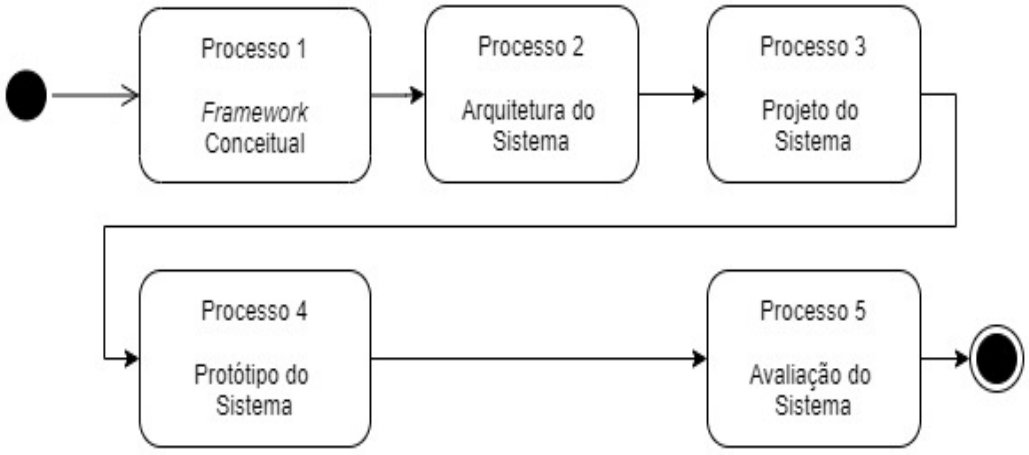
3.1. Procedimentos Metodológicos

A pesquisa possui cinco processos (Figura 1):

- *Processo 1*: *Framework Conceitual*. Levantamento dos requisitos funcionais;
- *Processo 2*: *Arquitetura do Sistema.* Apresentação das camadas do sistema;
- *Processo 3*: *Projeto do Sistema.* Funcionamento do sistema;
- *Processo 4*: *Protótipo do Sistema.* Apresentação da tela do sistema;
- *Processo 5*: *Avaliação do Sistema.* Demonstração de requisitos e processos do sistema.

O diagrama de atividades abaixo, que serve para demonstrar o comportamento de um sistema (Sommerville, 2018), ilustra os passos da metodologia do projeto.

Figura 1 - Sequência dos Procedimentos Metodológicos Baseados em Nunamaker Jr et al. (1990)



Fonte: Os Autores

3.2. Objeto de Estudo

O objeto de estudo do projeto é a Empresa-X, que é uma empresa do setor de desenvolvimento de sistemas, localizada em São Paulo, que adquiriu reconhecimento nos seus mais de 20 anos de mercado e que presta serviços a várias empresas de grande porte de diversos nichos diferentes.

A Empresa-X detêm muitos sistemas, os quais serve a seus clientes. Além do mais, todos esses sistemas estão situados em ambiente *web*, na linguagem de programação C#, isso em conjunto com o *framework* ASP.NET. Ainda, tais sistemas utilizam bancos de dados, e a Empresa-X necessita de uma rotina que dê proteção extra no momento de conexão ao SGBD para esses *softwares*.

4. Resultados e Discussões

4.1. *Framework* Conceitual

A Empresa-X necessita de uma solução computacional que fortifique a proteção de suas cadeias de caracteres de conexão. Para isso, o sistema aqui proposto, dividido em duas aplicações, comtempla alguns requisitos que visam alcançar esse objetivo.

Os requisitos são todas as funcionalidades, serviços e restrições de um sistema, que podem ser classificados de diversas formas e, dentre essas classificações, estão os requisitos funcionais, que compõem as funções do sistema e tudo que ele deve ser capaz de realizar (Sommerville, 2018).

A primeira aplicação do sistema pode atuar de forma off-line, fora do servidor, e é responsável por tratar a *string* de conexão e depois gerar um arquivo a partir das tratativas realizadas nela, que será enviado para o servidor.

No requisito SSRF01 o desenvolvedor insere a *string* de conexão em uma caixa de texto. Em seguida, no SSRF02, após um botão ser pressionado, é feita uma verificação para avaliar se a *string* de conexão digitada está estabelecendo conexão com o SGBD. Por fim, no SSRF03, depois que há interação com outro botão, a cadeia de caracteres de conexão é encriptada e armazenada em um arquivo.

Tabela 2 – Sistema Satélite - Requisitos Funcionais do Sistema

Nº	Rótulo	Descrição
SSRF01	Coleta	A *string* de conexão é coletada
SSRF02	Teste de conexão	Ocorre a tentativa de efetuar a conexão com o SGBD
SSRF03	Encriptação e arquivo	A *string* de conexão é encriptada e o arquivo é gerado

Fonte: Os Autores

Quanto à segunda aplicação do sistema, ela exerce seu papel em ambiente servidor e tem o encargo de acessar e manipular o arquivo criado na aplicação *off-line*, que será utilizado para estabelecer a conexão com o banco de dados por meio de uma classe.

No contexto do SERF01, como apresentado na Tabela 3, a classe recebe o arquivo gerado pela primeira aplicação e faz uma leitura dele. Posteriormente, por meio da análise do arquivo, a cadeia de caracteres de conexão, ainda encriptada, é coletada (SERF02) e descriptografada (SERF03) e realizada a conexão (SERF04) com o SGBD.

Tabela 3 – Sistema ERP - Requisitos Funcionais do Sistema

Nº	Rótulo	Descrição
SERF01	Lê arquivo	Lê o arquivo que contém a *string* de conexão encriptada
SERF02	Coleta	Coleta a *string* de conexão do arquivo
SERF03	Decriptação	Decripta a *string* de conexão
SERF04	Conexão	Estabelece a conexão com o SGBD

Fonte: Os Autores

4.2. Arquitetura do Sistema

O sistema é dividido em duas aplicações de ambiente *web*, compostas pela arquitetura *n-tier*, que é uma arquitetura de *software* que organiza a estrutura de uma aplicação em diferentes camadas (Microsoft, 2023).

A primeira aplicação é constituída de três camadas, que são as camadas de apresentação, negócios e dados. Por outro lado, a segunda aplicação possui apenas as camadas de negócios e dados.

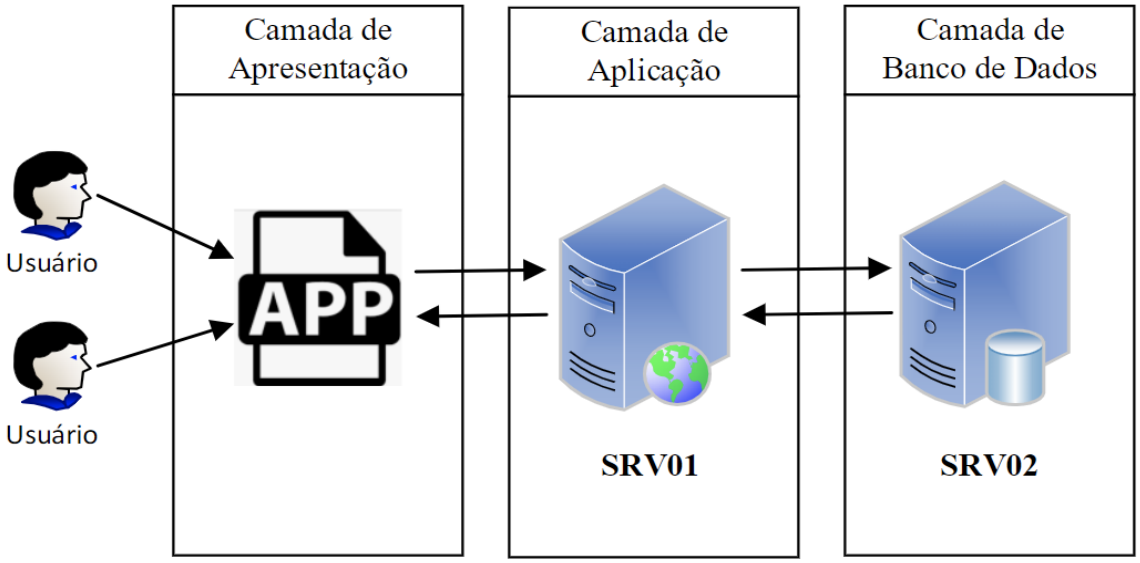
Na primeira aplicação, a camada de apresentação, que é a parte responsável pelo visual do sistema (Elmasri, 2021), possui um campo de texto e dois botões, que servem para, respectivamente, receber a *string* de conexão em texto puro, gerar um arquivo com a *string* de

conexão encriptada e testar a conexão com o SGBD a partir da decriptação dessa cadeia de caracteres de conexão.

Em relação à camada de negócios das duas aplicações, que constitui as regras de negócio do sistema (Pressman, 2021), na primeira aplicação, ela é responsável por encriptar a cadeia de caracteres de conexão, por testar a *string* de conexão computada e por gerar o arquivo que armazena a encriptação da cadeia de caracteres de conexão. Em contrapartida, o escopo das regras de negócio da classe abrangem a leitura do arquivo gerado na primeira aplicação, a coleta da *string* de conexão, a decriptação dela e o estabelecimento da conexão com o SGBD.

Referente à camada de dados das duas partes do sistema, que é a responsável pelo banco de dados e pelo SGBD (Pressman, 2021), seu funcionamento está associado, em ambos os *softwares*, quanto à necessidade de estabelecer uma conexão com o SGBD.

Figura 2 – Arquitetura dos Sistemas Baseado em Microsoft (2022)



4.3. Projeto do Sistema

Na aplicação *off-line*, a cadeia de caracteres de conexão é recebida em texto, similar à Figura 2, e o resultado final deverá ser a *string* criptografada, como na Figura 3.

Figura 2- Exemplo de Cadeia de Caracteres de Conexão

```
Server=DESKTOP-D96M1Q1\\SQLEXPRESS; DataBase=nomeDoBancoDeDados;User
Id=nomeDeUsuario; Password=12345678;
```

Fonte: Os Autores

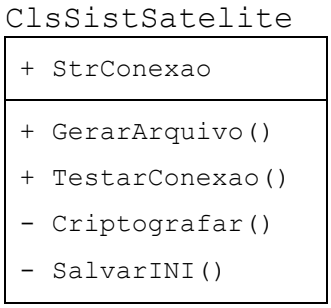
Figura 1 - Exemplo de *String* Criptografada

```
4mDC5x0rMpf1nee4#NzuqBUIzt9/2HzcfaXGnaDUgBdx8dyc0VduH7cnCjcYAXesAAQlTQnrj4Z
87dxJN/PCX2OLFFOJgxyr4aUsjKoXZ+kbe/ZF5RavgB8MkB3tUDgqBMnRNOrHBjjfkfI/Z9B+Z1
IPgy4On3ZnpKYRGYJAwK7uhxvQIJsNQ5sVRBx0=
```

Fonte: Os Autores

O diagrama de classes (Figura 4) apresenta a classe do sistema satélite (ClsSistSatelite) em que funcionará de forma paralela ao ERP. Contém um atributo que representa a *string* de conexão a ser criptografada (StrConexao) e os métodos GerarArquivo() e TestarConexao().O método de geração do arquivo irá executar os métodos Criptografar() e SalvarINI().

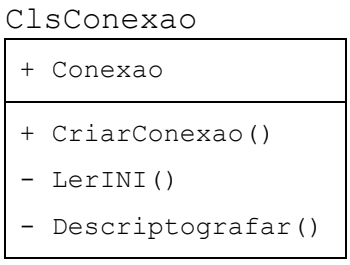
Figura 4 - Diagrama de Classe da Primeira Aplicação



Fonte: Os Autores

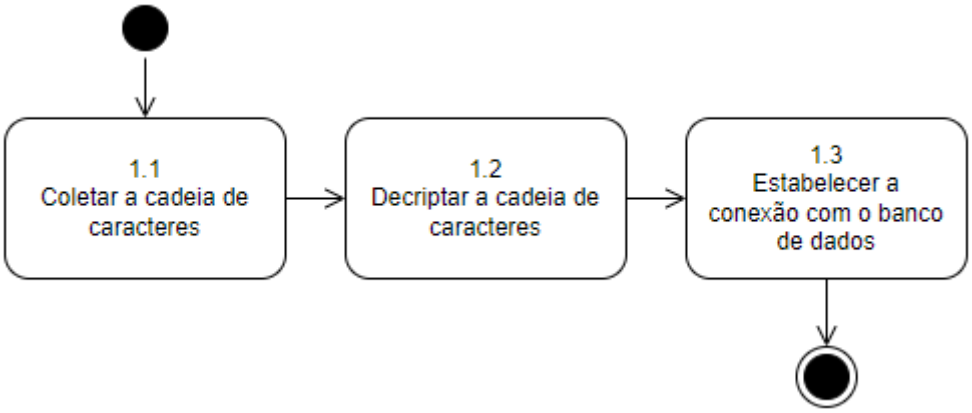
O sistema ERP irá executar uma outra classe, a ClsConexao (Figura 5 e Figura 6), que será consumida por meio do método CriarConexao() que acionará o LerINI() buscando o arquivo com a *String* criptografada, em seguida a execução do método Descriptografar(). A classe irá realizar a conexão com o SGBD e retornará por meio do método Conexao.

Figura 5 - Diagrama de Classe de Conexão



Fonte: Os Autores

Figura 6 – Diagrama de Atividade da ClsConexao

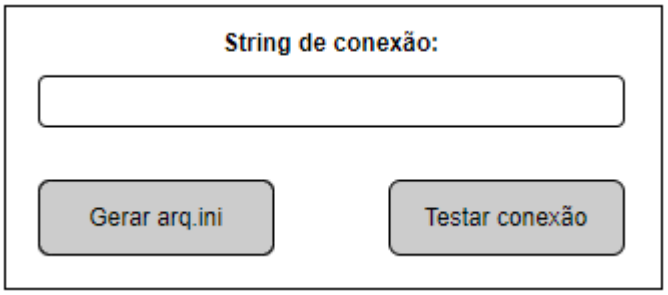


Fonte: Os Autores

4.4. Protótipo do Sistema

A Figura 7 representa o formulário da camada de apresentação da primeira aplicação, que é constituído de uma caixa de texto e dois botões: a caixa de texto é responsável por receber a *string* de conexão; o botão "*Gerar arq.ini*" é incumbido de criar um arquivo que armazena a encriptação da cadeia de caracteres de conexão, computada no *input* de texto; o botão "*Testar conexão*" é encarregado de verificar se a *string* de conexão encriptada no arquivo gerado está ou não estabelecendo conexão com o SGBD.

Figura 7 - Exemplo de Interface do Arquivo Satélite



Fonte: Os Autores

4.5. Avaliação do Sistema

A validação de que os processos foram contemplados diante das necessidades da Empresa-X foram apresentadas ao gerente de TI que realizou a aprovação das classes e estrutura sistêmica para ser incorpora nos projetos da empresa.

4.6. Discussões

A proposta exposta não direciona para um sistema de criptografia específica, deixando a cargo do Gerente de TI ou o Gerente de Segurança da Informação a tarefa de escolha. Dependendo da escolha feita, poderá ser necessário colocar um campo auxiliar para receber a semente da criptografia.

5. Considerações Finais

A cadeia de caracteres de conexão ao banco de dados deve ser devidamente protegida, já que contém informações confidenciais que estabelecem conexão com a camada de dados de um sistema para evitar ataques e, consequentemente, a obtenção de acesso ao banco de dados.

O resultado obtido é uma proposta de segurança da *string* de conexão por meio da criptografia, apresentada em formato de diagrama de atividades e diagrama de classes.

A contribuição teórica está na pesquisa de qual tipo de criptografia será mais efetiva nesse tipo de estrutura. A contribuição prática é que Gerentes de TI e de Segurança da Informação poderão utilizar a proposta em seus sistemas. Em futuros trabalhos estão o direcionamento de um tipo de criptografia e da incorporação de do tipo de SGBD que será conectado.

Referências

BRASIL. Lei Geral de Proteção de Dados (LGPD). Governo Federal, 2018.

ELMASRI, R.; NAVATHE S. Sistemas de banco de dados (7ª edição). Pearson, 2019.

FIRST. Common Vulnerability Scoring System (versão 4.0). 2023.

FRAGA, B. Técnicas de Invasão: Aprenda as técnicas usadas por hackers em invasões reais. Editora Labrador, 2019.

FREUND, G.; SEMBAY, M.; MACEDO, D. Proveniência de Dados e Segurança da Informação: relações interdisciplinares no domínio da Ciência da Informação. Revista Ibero-Americana de Ciência da Informação, v. 12, n. 3, 2019.

HINTZBERGEN, J., HINTZBERGEN, K., SMULDERS, A., & BAARS, H. Fundamentos de Segurança da Informação Com base na ISO 27001 e na ISO 27002 (3ª edição). Brasport Livros e Multimídia Ltda, 2018.

MARTINS, G.; THEÓPHILO, C. Metodologia da Investigação Científica para Ciências Sociais Aplicadas (3ª edição). Atlas, 2016.

MICROSOFT. Estilo de arquitetura de N camadas. https://learn.microsoft.com/pt-br/azure/architecture/guide/architecture-styles/n-tier. 2023.

NUNAMAKER JR, J.; CHEN, M.; PURDIN, T. System Development in Information Systems Research. Journal of Management Information Systems, v. 7, 1990 - 1991.

PRESSMAN, R. Engenharia de Software (9ª edição). Bookman, 2021.

SHARDA, R.; DELEN, D.; TURBAN, E. Business Intelligence e Análise de Dados para Gestão do Negócio (4ª edição). Bookman, 2019.

SOMMERVILE, I.; Engenharia de Software (10ª edição). São Paulo: Pearson, 2018.

TANENBAUM, A. Redes de Computadores (6ª edição). São Paulo: Bookman, 2021.