

**UNIVERSIDADE DO ESTADO DO PARÁ – UEPA**

**CURSO DE Bacharelado em Engenharia de Software**

Eduardo Ribeiro de Oliveira

Ana Caroline Araujo

Fabiano Filho

Ruan Felipe

**Engenharia Reversa:**

**Rational XDE e Incode**

Castanhal

2021

Sumário

[**1 - Introdução** 3](#_Toc72150176)

[**2 - Ferramenta Incode** 3](#_Toc72150177)

[**3 - Ferramenta Rational XDE** 4](#_Toc72150178)

[**3.1. Projetando Bancos de Dados Utilizando o Rational XDE Developer**. 4](#_Toc72150179)

[**3.2. Desenvolver Modelo de Dados Lógicos** 5](#_Toc72150180)

[**3.3. Desenvolver Design de Banco de Dados Físico** 5](#_Toc72150181)

[**4 - Aplicação de Ferramenta** 6](#_Toc72150182)

[**5 - Conclusão** 12](#_Toc72150183)

[**Bibliografia** 12](#_Toc72150184)

# **1 - Introdução**

Engenharia reversa é o processo de descobrir os princípios tecnológicos e o funcionamento de um dispositivo, objeto ou sistema, através da análise de sua estrutura, função e operação. Objetivamente a engenharia reversa consiste em, por exemplo, desmontar uma máquina para descobrir como ela funciona. Existem algumas ferramentas utilizadas para tornar esse processo mais eficiente, entre eles o Incode**® e o Rational XDE** Developer e são essas ferramentas que iremos abordar nos tópicos seguintes.

# **2 - Ferramenta Incode**

O **Incode®** é um *software* baseado em Engenharia Reversa cujo objetivo principal é recuperar o conhecimento de sistemas legados, estruturando uma Base de Conhecimento e disponibilizando suas informações de forma ágil, precisa e eficiente sob qualquer contexto ou ponto de vista.

Fundamentalmente, o **Incode®** está implementado sobre três aspectos principais que são:

* **Montagem da base de conhecimento, através de processos estruturados e totalmente flexíveis capazes de absorver novos tipos de linguagens/tecnologias de forma bastante simples e eficiente;**
* **Linguagem Incode, que é uma linguagem embutida na ferramenta Incode® para extração de informações da base de conhecimento utilizando o conceito de templates que possibilitam a elaboração de uma enorme diversidade de tipos aplicabilidades e produtos;**
* **Processo de trabalho Incode®, que corresponde à estruturação e ao *know-how* de como fazer uso da solução Incode®, focando em processos de automação com produtividade e qualidade aplicadas sobre necessidades e projetos de caráter clássico/convencionais.**

Através da utilização do **Incode®**, e apoiando-se na Base de Conhecimento gerada, é possível obter:

* [**Documentação de Sistemas**](http://www.beltsysplus.com.br/documentacao-de-sistemas/)**:** Criar e manter a documentação física de um ou mais sistemas, contendo referência cruzada entre programas, telas, bases de dados, rotinas batch, transações online, gerenciadores de processos batch, relações entre sistemas, português estruturado de programas etc.
* [**Análise de Impacto**](http://www.beltsysplus.com.br/analise-de-impacto/)**:** Efetuar análise de impacto para mudanças estruturais em sistemas, indicando exatamente quais serão os componentes afetados direta ou indiretamente pela alteração, e, opcionalmente, fazendo estas alterações diretamente e de forma automatizada.
* **Métrica:** Gerar informações de métrica de programas e/ou sistemas, processo especialmente útil em caso de terceirização de desenvolvimento.
* **Qualidade:** Gerar informações de qualidade de sistemas, indicando pontos de possíveis falhas, oportunidades de melhoria de performance do código, indicação de código morto etc.
* **Automação e Reengenharia:** Efetuar conversão em sistemas, seja de banco de dados, linguagem ou plataforma, individualmente ou em conjunto.

# **3 - Ferramenta Rational XDE**

Rational XDE Developer - Java Platform Edition é uma ferramenta Case que auxilia no desenvolvimento de um software profissional. Combina seu design e o desenvolvimento Java em um ambiente totalmente integrado. Esse único ambiente evita a necessidade de alternar entre ferramentas externas.

O Rational XDE aprimora o modo no qual você trabalha. Por exemplo, ele permite:

* **- Mover das análises e do design todas as formas de codificação dentro do mesmo ambiente.**
* **- Especificar a sincronização manual ou automática entre o modelo e o código.**
* **Definir os gabaritos de código e de modelo para economizar tempo e ajudar a aplicar os padrões.**
* **- Criar modelos UML e formato livre e validar modelos para conformidade UML.**
* **Construir modelos dentro do mesmo projeto que possam ser rastreados a outro.**
* **- Reutilizar os padrões definidos para compartilhar códigos dentro da equipe.**

## **3.1.** [**Projetando Bancos de Dados Utilizando o Rational XDE Developer.**](https://www.cin.ufpe.br/~gta/rup-vc/core.base_rup/guidances/toolmentors/database_design_51B0B5D9.html)

A ferramenta de software Rational XDE (TM) inclui recursos que permitem que os desenvolvedores de software e os desenvolvedores de bancos de dados reproduzam o software e o banco de dados utilizando a mesma ferramenta.

Existem três maneiras de começar um Modelo de Dados no XDE:

* **Construir o Modelo de Dados diretamente no XDE utilizando as ferramentas do Modelador de Dados.**
* **Transformar classes persistentes no Modelo de Design para criar tabelas no Modelo de Dados.**
* **Aplicar engenharia reversa de um esquema de banco de dados ou script DDL existente para criar um Modelo de Dados.**

## **3.2. Desenvolver Modelo de Dados Lógicos**

Alguns projetos podem precisar criar um modelo "lógico" idealizado do design de banco de dados que captura uma visualização independente de aplicativo das principais entidades de dados lógicos e seus relacionamentos. O Modelo de Dados Lógicos pode ser criado diretamente utilizando as ferramentas do *XDE Data Modeler.*

## **3 .3. Desenvolver Design de Banco de Dados Físico**

Pode-se aumentar o nível de detalhamento do Modelo de Dados Lógico através de um Modelo de Dados Físico utilizando o *RationalXDE.*

As principais etapas para desenvolver um Modelo de Dados Físicos são:

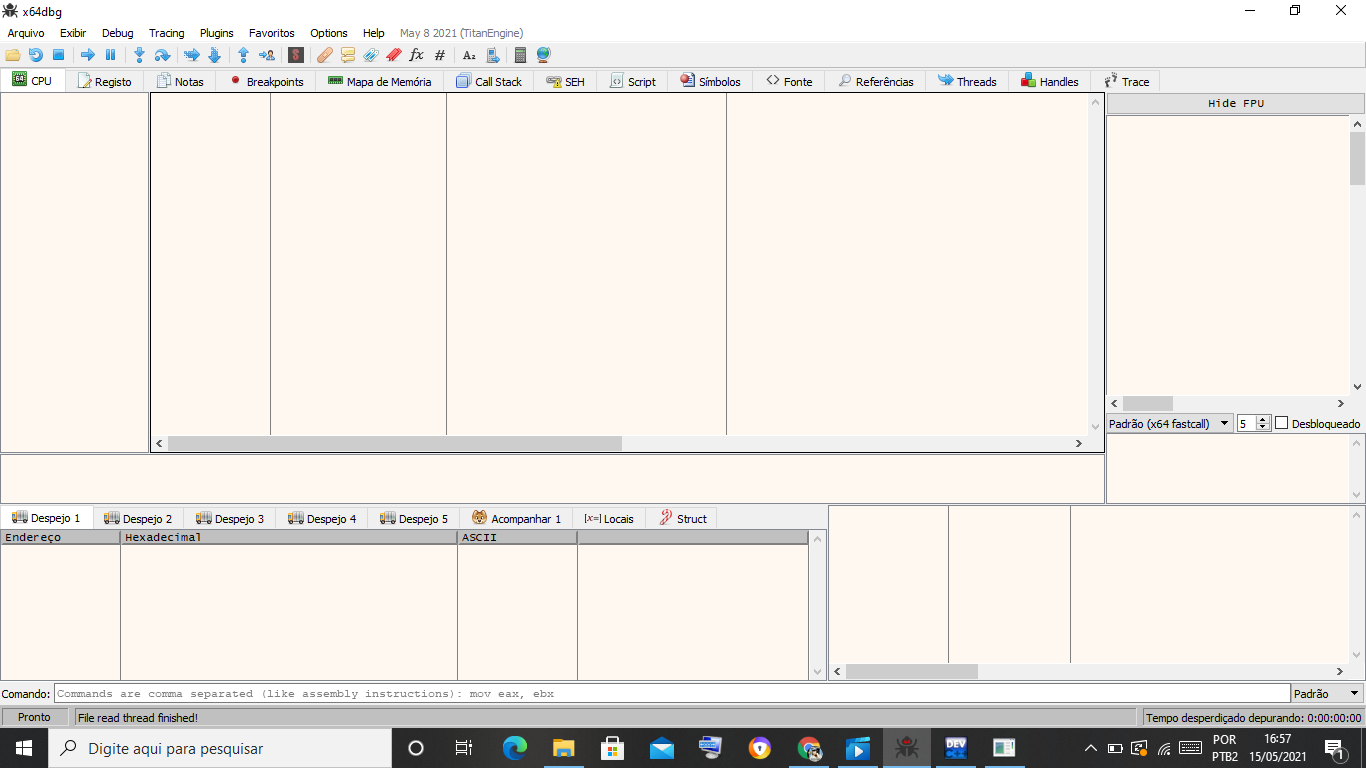
* **Definir Domínios**
* **Criar Elementos de Design de Banco de Dados Físico Inicial**
* **Definir Tabelas de Referência**
* **Criar Restrições de Chave Primária e Chave Exclusiva**
* **Definir Regras de Aplicação de Integridade Referencial e de Dados**
* **Desnormalizar o Design de Banco de Dados para aperfeiçoar o Desempenho**
* **Otimizar o Acesso a Dados**
* **Definir Características de Armazenamento**
* **Projetar Procedimentos Armazenados para Distribuir Comportamento de Classe ao Banco de Dados**
* **Interface do Programa.**

# **4 - Aplicação de Ferramenta**

**Objetivo**: Usar a ferramenta para prática o *hacking* através de engenharia reversa no programa de criptografia feito em linguagem C .

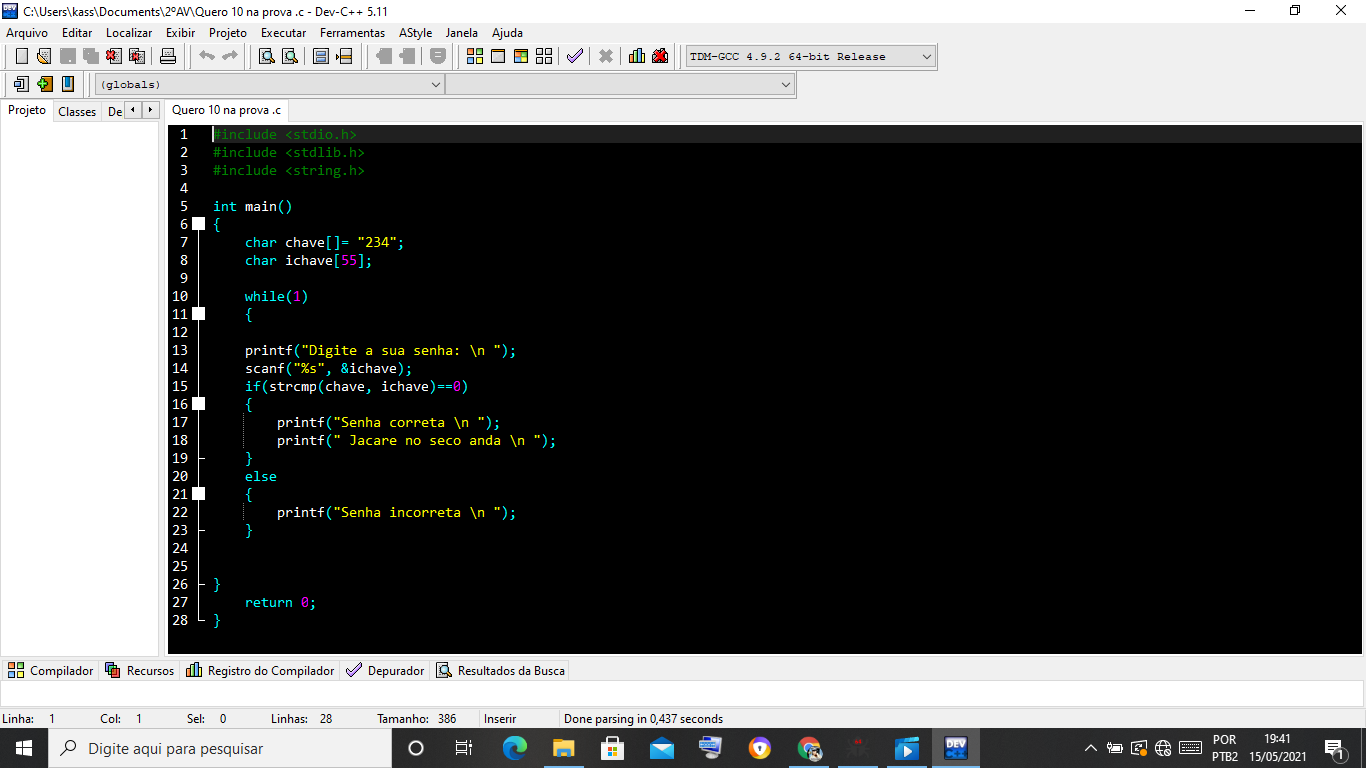
Após baixar e instalar a ferramenta x64dbg, depois de aberta irá aparecer exatamente como na imagem 1

**(imagem 1).**



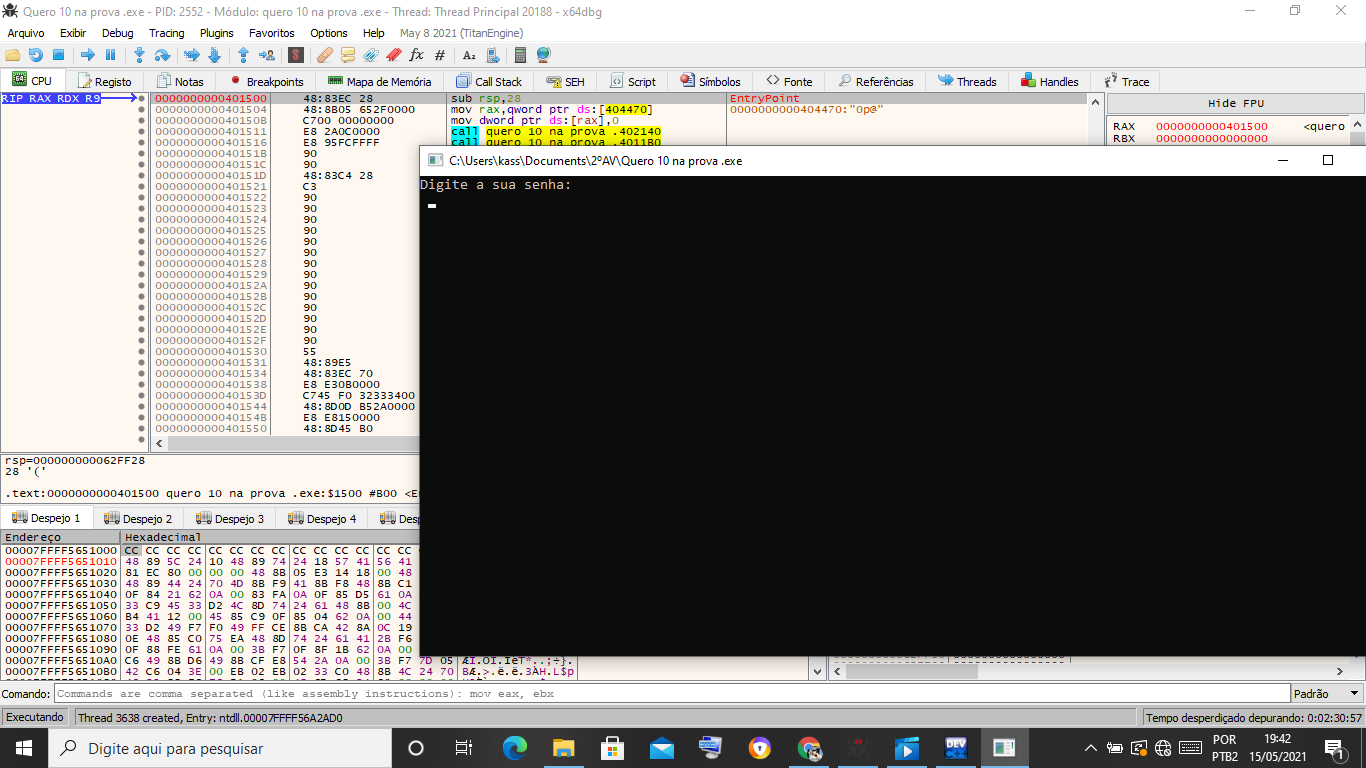
Criado um programa na linguagem C e salvo com o nome **“quero 10 na prova”**, o programa tem como objetivo criar uma criptografia para que se o usuário digitar a senha correta apareça **“senha correta”** e mais uma mensagem, caso a senha seja incorreta vai aparecer **“senha incorreta”** sem nem uma mensagem.

**( Imagem 2)**



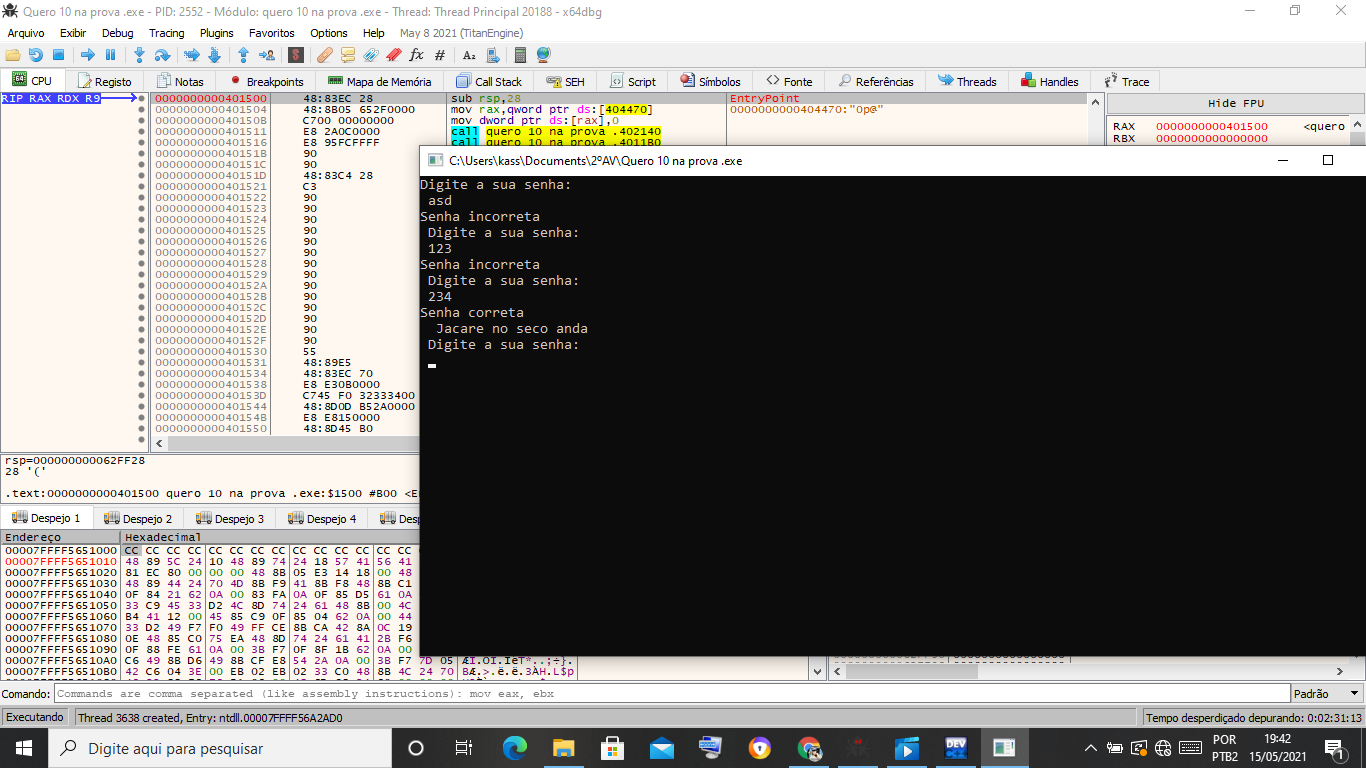
Foi aberto o código em C na ferramenta para desmontar o código.

**(imagem 3)**



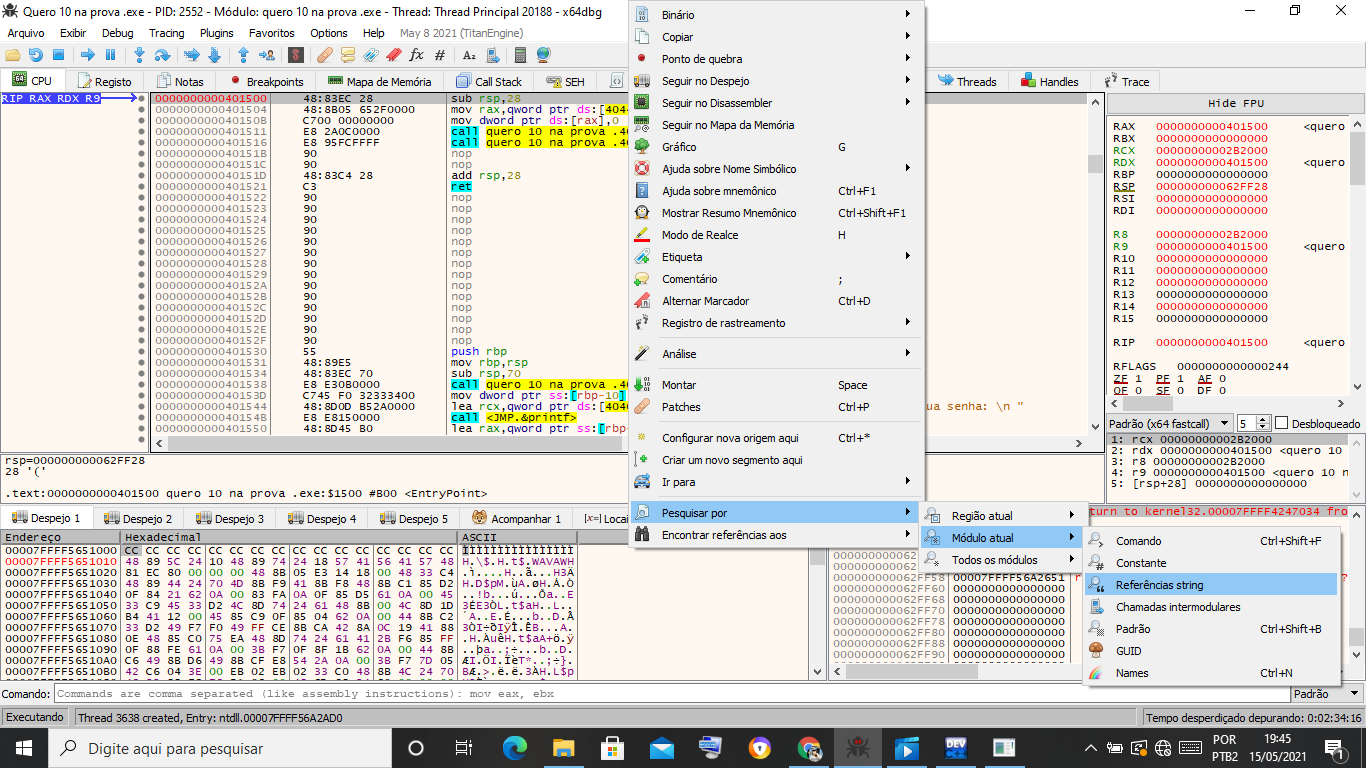
O código desmonta, porém com muitas linhas e a melhor opção era adicionar um comando para procurar strings.

**(imagem 4)**



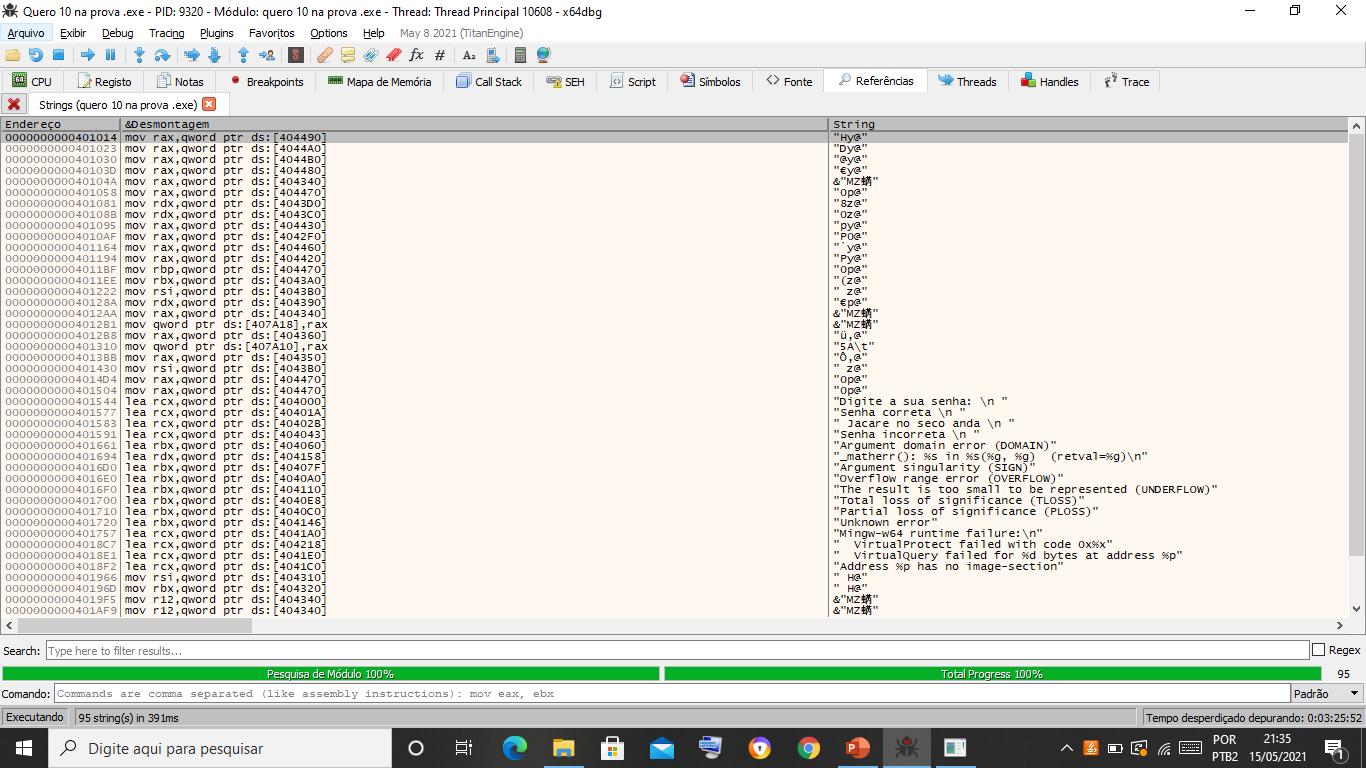
resultado da pesquisa

**(imagem 5)**



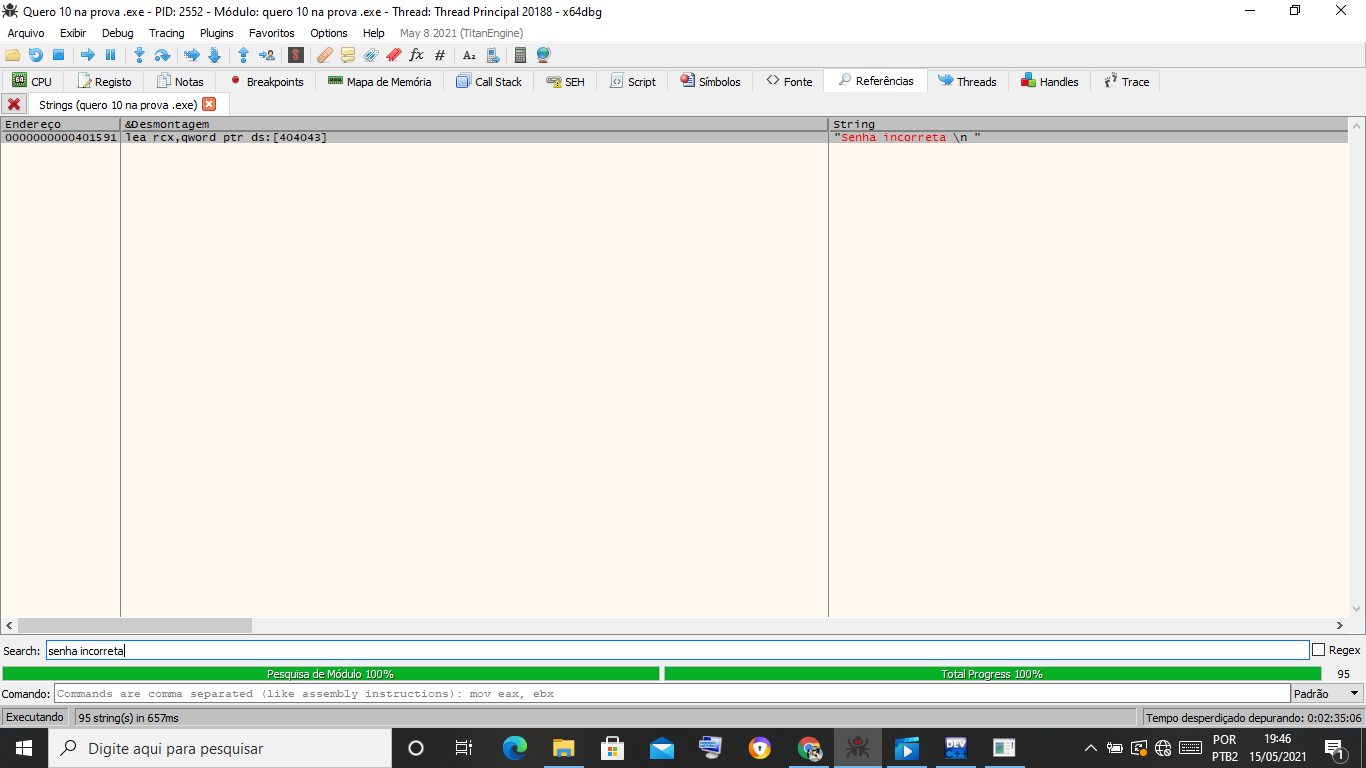
Com o filtro de “senha incorreta” e depois foi clicado no endereço dessa linha.

**(Imagem 6)**



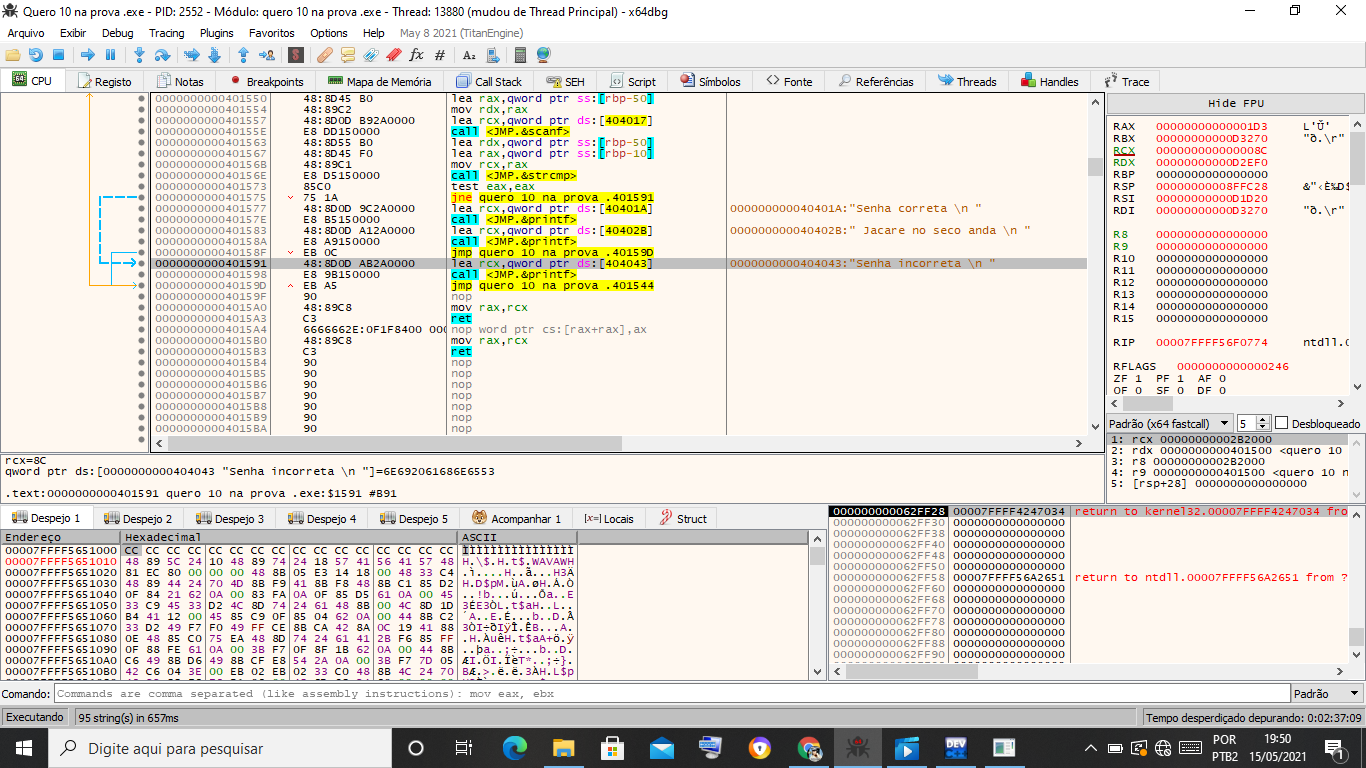
Após clicar no endereço, o usuário foi direcionado para exatamente onde estava o código.

**(Imagem 7)**



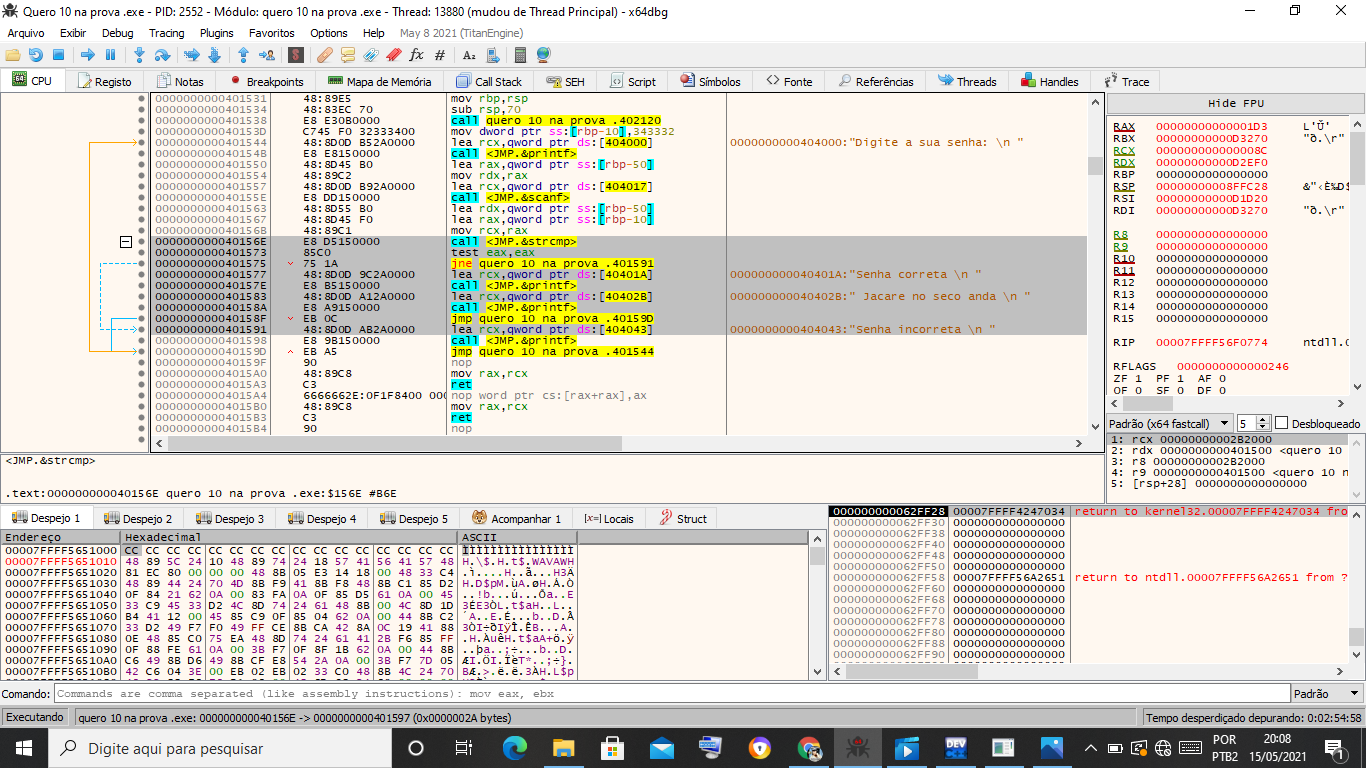
Prestem a atenção somente onde está em destaque, na primeira linha o comando “call” está chamando a função de comparação de string e logo abaixo a instrução “Jne” que verificar o resultado da operação do arquivo “quero 10 na prova”. Nesse sentido, caso seja verdadeiro vai exibir o “senha correta” e mais a mensagem, caso seja falso vai pular para o “senha incorreta” e não vai exibir nem uma mensagem.

**(imagem 8)**



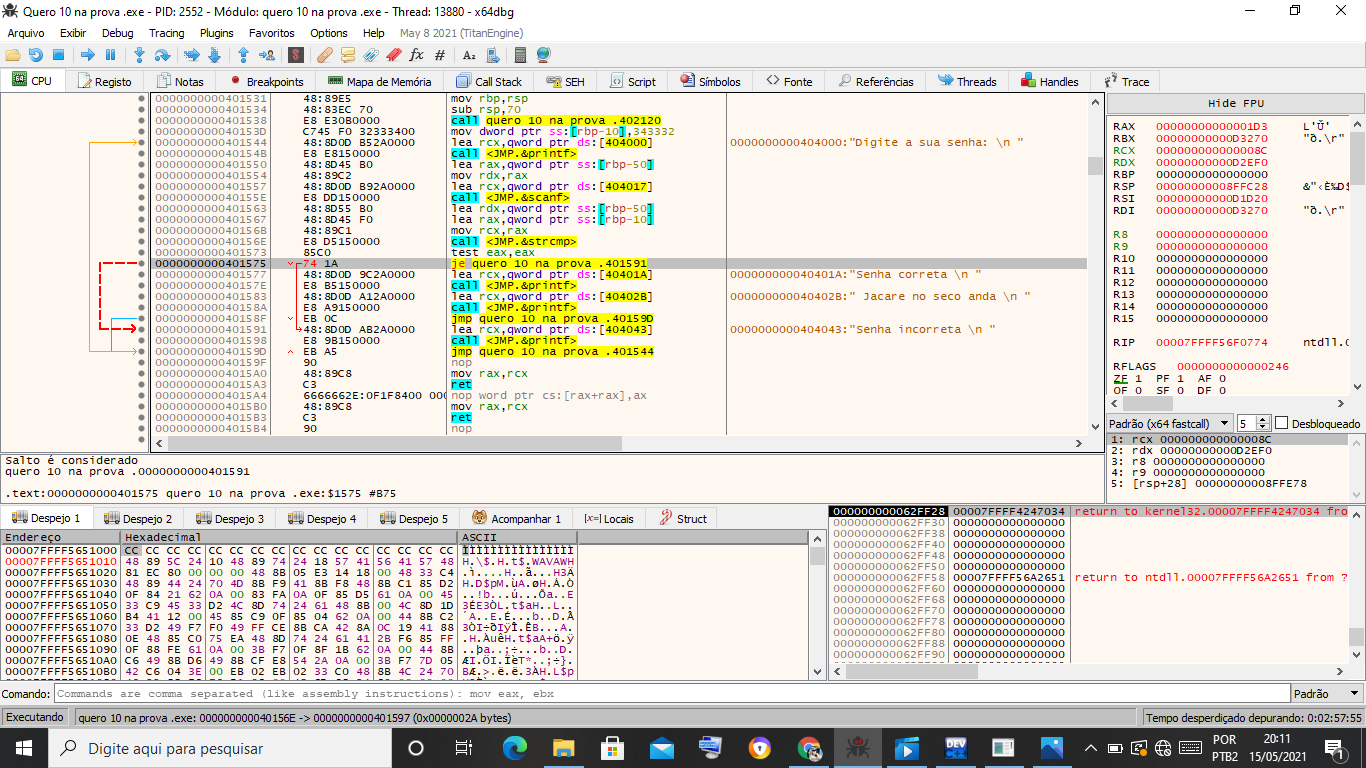
A instrução “jne” foi trocada para o oposto “je”, assim se o usuário digitar a senha correta vai aparecer “senha incorreta “ e sem a mensagem e se o usuário digitar a senha incorreta vai aparecer “senha correta “ e também a mensagem.

**(imagem 9)**



Execução.

**(Imagem 10)**



# **5 - Conclusão**

Comtemplamos o que são e como funcionam as ferramentas Incode**® e o Rational XDE** Developer, e com isso podemos avaliar o quão proveitoso pode ser o uso das mesmas em nossos projetos de engenharia reversa, tendo em vista que essas ferramentas têm a finalidade de nos ajudar nos processos de engenharia reversa podemos afirmar que sua funcionalidade nos traz benefícios e simplifica os trabalhos aos quais são destinadas.

# **Bibliografia**

<http://www.beltsysplus.com.br/engenharia-reversa-de-software/>

<https://blog.cronapp.io/engenharia-reversa/>

<https://pt.wikipedia.org/wiki/Engenharia_reversa>

<https://www.cin.ufpe.br/~gta/rup-vc/core.base_rup/guidances/toolmentors/finding_actors_and_use_cases_rup_xde_uc_A5B7A9F7.html>

<https://www.cin.ufpe.br/~gta/rup-vc/core.base_rup/tools/rup_xde_D81FFA70.html>

<https://www.ibm.com/products?lnk=STW_US_SHP_NAV_TL&lnk2=trial_MHP>

<https://www.youtube.com/watch?v=vrA6ouuuizE>