FACULDADE DE TECNOLOGIA SENAI MATO GROSSO – FATEC SENAI

CURSO SUPERIOR DE DEFESA CIBERNÉTICA

WENDREW NICK COSTA TATEHIRA

ANÁLISE DE EXECUTÁVEIS PARA INICIANTES: UMA INTRODUÇÃO À ENGENHARIA REVERSA

CUIABÁ

2024

FATECSENAI Faculdade de Tecnologia SENAI Mato Grosso	TRABALHO DE CONCLUSÃO DE CURSO FATEC SENAI MT	Data: 12/07/2024

WENDREW NICK COSTA TATEHIRA

ANÁLISE DE EXECUTÁVEIS PARA INICIANTES: UMA INTRODUÇÃO À ENGENHARIA REVERSA

Monografia apresentada ao Curso Superior de Defesa Cibernética na Faculdade de Tecnologia SENAI Mato Grosso, como requisito parcial para obtenção do título de Tecnólogo em Defesa Cibernética.

Orientador: Prof. Esp. Alan Papafanurakis Heleno

CUIABÁ

2024

WENDREW NICK COSTA TATEHIRA

ANÁLISE DE EXECUTÁVEIS PARA INICIANTES: UMA INTRODUÇÃO À ENGENHARIA REVERSA

Trabalho final apresentado à Faculdade de Tecnologia SENAI Mato Grosso, como parte das exigências para a obtenção do título de Tecnólogo em Defesa Cibernética.

Cuiabá, 12 de julho de 2024.

BANCA EXAMINADORA

_____

Prof. Esp. Alan Papafanurakis Heleno
Orientador

_____

Prof. Msc. Fabiano Pontes Pereira da Silva
Afiliações

_____

Prof. Willdson Gonçalves Almeida
Afiliações (deixar aberto para preenchimento)

DEDICATÓRIA

Ao meu orientador Prof. Esp. Alan Papafanurakis Heleno, pela paciência na orientação e incentivo que tornaram possível a conclusão deste trabalho de conclusão de curso.

AGRADECIMENTOS

A Deus, que me proporcionou força e sabedoria durante a jornada. Aos profissionais e iniciantes na área de segurança da informação, que são fundamentais para a evolução de um mundo digital mais seguro. À minha namorada, Júlia Rezende, pelo seu apoio incondicionale por sempre acreditar em mim, mesmo nos momentos mais difíceis.

RESUMO

Este trabalho tem como objetivo principal fornecer uma introdução acessível e prática à engenharia reversa de executáveis para iniciantes. A crescente dependência tecnológica e a evolução constante das ameaças cibernéticas tornam essencial a compreensão do funcionamento do *malware* para resolução eficaz dessas ameaças.
No decorrer deste estudo, será apresentado os fundamentos teóricos da engenharia reversa e do *malware*, além de discutir as principais ferramentas para análise de programas. Ferramentas como *IDA PRO*, *Ghidra*, OllyDbg e *Cheat Engine* são detalhadas, informando a sua aplicação prática para análise de executáveis.
Este estudo destaca a importância de se ter uma base sólida em Segurança da Informação, incluindo engenharia reversa, mostrando que a capacidade de bons profissionais é crucial para a proteção dos dados digitais.
Uma abordagem didática visa tornar o conhecimento mais compreensível e acessível, desta forma os iniciantes enfrentarão os desafios com confiança e competência.

Palavras-chave: engenharia reversa, *Malware*, segurança da informação, Iniciantes, Análise de malware, ferramentas de segurança da informação**.**

LISTA DE ILUSTRAÇÕES

Sumário

1. INTRODUÇÃO

A engenharia reversa (ER) é o processo de desmontar ou analisar um produto para entender seu funcionamento interno. Com isso, pode-se envolver a desmontagem física de um objeto ou análise de seu *software* e a investigação de seus componentes e processos. O principal objetivo é descobrir como algo foi feito para poder replicar, melhorar ou consertar. É muito utilizada em diversas indústrias e profissionais de segurança da informação.

Com o passar dos anos, a segurança cibernética vem se tornando uma preocupação constante, tanto para instituições quanto para indivíduos, exigindo maior atenção especialmente à evolução dos *softwares* maliciosos, conhecidos como *malwares*. Entender o seu funcionamento é fundamental para mitigar ameaças e obter respostas rápidas a incidentes, proteger dados sensíveis e para garantir a segurança digital.

O uso de ferramentas é crucial nesse processo de engenharia reversa de executáveis, permitindo que os profissionais de segurança da informação analisem, compreendam o funcionamento e o comportamento de um *software* malicioso, desenvolvendo estratégias eficazes para a neutralização dessas ameaças.

A reengenharia envolve análise de modificações de sistemas para melhorar desempenho ou adaptá-los a novos requisitos. Esse é um processo vital para a manutenção e atualizações de sistemas antigos e para garantir que eles continuem a atender às necessidades das suas organizações.

Além disso, a engenharia reversa segue sendo uma ferramenta poderosa no desenvolvimento de novos produtos. Empresas utilizam essa técnica para analisar produtos concorrentes, entender suas funcionalidades e inovações e assim, desenvolver melhorias ou novos produtos que atendam de forma mais eficaz às demandas do mercado.

Outro campo relevante onde a engenharia reversa se destaca é na Perícia Forense Digital. A Pericial Forense tornou-se determinante na investigação de incidentes cibernéticos, permitindo a coleta e análise de evidências digitais que podem ser usadas para identificar e processar criminosos. Essa prática envolve a recuperação de dados, análise de atividades suspeitas e a construção de um perfil de acordo com o

os dados coletados do ataque, provendo uma compreensão detalhada dos eventos que ocorreram e ajudando a prevenir futuros incidentes cibernéticos.

Este trabalho, explora como a engenharia reversa pode ser ensinada a iniciantes, destacando suas aplicações na análise de executáveis e suas contribuições para a segurança cibernética e o desenvolvimento de produtos.

Ao proporcionar um entendimento aprofundado e prático, será esperado prover ensino a novos profissionais para enfrentar desafios da era digital com confiança e competência.

1.1. Exemplo real da engenharia reversa fora da computação. IndústriaAutomotiva: a análise do Toyota *Prius*

Assim como na computação, a engenharia reversa é amplamente utilizada em outras indústrias, como na automotiva para entender e melhorar suas tecnologias. Um exemplo real é a análise do Toyota *Prius*, o primeiro carro híbrido produzido em massa. Empresas como Ford e General Motors demonstraram o *Prius* para estudar seu sistema híbrido e seus componentes, para desenvolver suas próprias tecnologias híbridas (Vlasic, 2010). Esse processo foi de grande avanço e inovação e a eficiência energética na indústria automotiva.

1.2 Exemplo real de engenharia reversa na Computação: a análise de *Stuxnet*

Assim como em outras indústrias, na computação, a engenharia reversa é fundamental para análise de *malwares* e a compreensão de ameaças cibernéticas. Um exemplo real é a análise do *worm Stuxnet*, é um *malware* sofisticado, descoberto em 2010, tendo como principal algo sistemas de controle industrial. Especialistas da área de segurança cibernética realizaram engenharia reversa no *Stuxnet* para entender seu funcionamento, identificar e mitigar seus efeitos. Essa análise revelou a complexidade do *malware* e destacou a necessidade de elevar as estratégias de defesa cibernética (Langner, 2011)

1.3 Executáveis e *Malware*: Definições Fundamentais e Impacto na Segurança Cibernética

Um executável é um arquivo de computador que contém instruções para o sistema operacional, ele faz um programa específico para realizar execuções. Segundo *Russinovich* (2012), os executáveis podem ser programas legítimos, como navegadores web e ferramentas de modificações de texto. Em resumo, são ferramentasessenciais para os computadores.

Por outro lado, os *malwares*, referem-se a programas maliciosos projetados para danificar sistemas, roubar informações ou ganhar acesso não autorizado. Alguns exemplos podem incluir vírus, *worms* e *ransomwares*, esses representam ameaças significativas à cibersegurança.

1.4 A importância da Segurança da Informação e o desenvolvimento das ameaças:

A segurança da informação é essencial para proteger dados e sistemas contra ataques cibernéticos e acesso não autorizado. As ameaças se tornam cada vez mais sofisticadas à medida que a tecnologia avança. Por exemplo, as soluções da Kaspersky impediram o lançamento de um malware bancário em computadores de 106.863 usuários no primeiro trimestre de 2023. Essa informação mostra que as ameaças estão se tornando cada vez mais complexas e que é fundamental ter soluções de segurança eficientes. Gráficos como os fornecidos pela Kaspersky mostram a necessidade de manter as estratégias de segurança atualizadas para enfrentar novas ameaças.

Figura 1 – Evolution malware 2023 statistic

2. PROBLEMATIZAÇÃO

Como os iniciantes podem enfrentar a complexidade do estudo da engenharia reversa para neutralizar ameaças cibernéticas que estão em constante evolução?

3. JUSTIFICATIVA

A Segurança da Informação vem se tornando cada vez mais indispensável, sendo essencial nos dias de hoje. Ao passar do tempo, a tecnologia continua em constante evolução, a engenharia reversa está para ajudar o indivíduo a entender o funcionamento, comportamento e assim, poder neutralizar essas ameaças, porém, é algo complexo e entendido como inacessível para iniciantes.

A engenharia reversa como qualquer outra linha de aprendizado, demanda aquisição de uma base sólida do assunto, caso contrário, só resultará em frustrações, podendo levar a danos estruturais, perda de dados sensíveis e financeiros em uma organização (Stalling & Brown, 2018).

A formação de novos profissionais de segurança da informação com as competências necessárias para realizar análise de executáveis de forma eficaz não se trata apenas de habilidade técnica, mas é extremamente necessária para proteção da sociedade digital.

Este estudo vem com objetivo de abrir as portas para esse conhecimento, tornando-o mais compreensível e acessível para iniciantes, auxiliando uma geração de futuros profissionais.

O médio e longo prazo, essa formação contribuirá para a sociedade, onde os indivíduos e organizações saberão operar com confiança e segurança.

4. OBJETIVOS

4.1. Objetivo Geral:

Prover ensino para iniciantes em cibersegurança para a realização de técnicas de engenharia reversa, promovendo uma base sólida de compreensão, envolvendo habilidades práticas para proteger sistemas e dados das organizações.

4.2 Objetivo Específico:

Desenvolvimento da Ferramenta Instrucional: Desenvolver uma ferramenta que facilite o aprendizado de engenharia reversa por meio de workshops, estudos de casos e exercícios práticos. O Cheat Engine serviu como um exemplo útil de análise de software e modificação de jogos, permitindo que os usuários interajam diretamente com uma ferramenta de engenharia reversa real.

Aplicação Prática: Os iniciantes podem aplicar técnicas de engenharia reversa em um ambiente controlado por meio de exercícios e simulações usando modelos reais e ferramentas como IDA Pro e Ghidra. O Cheat Engine foi projetado para fornecer aos usuários uma experiência interativa e desafiadora, permitindo que eles alterem o conteúdo de um jogo.

Avaliação e Feedback: Testar a eficácia da ferramenta instrucional com usuários iniciais e ajustar conforme necessário para aumentar a compreensão e a prática dos conceitos de engenharia reversa.

4.3 O resultado desejado:

O produto final será uma ferramenta de instrução que pode incluir um curso online, material de apoio e simulações práticas. A aplicação de conceitos de engenharia reversa em ambientes controlados deve capacitar os iniciantes em cibersegurança a identificar e analisar ameaças de segurança.

5. REFERENCIAL TEÓRICO

5.1. Segurança da Informação

A Segurança da Informação vem se tornando cada vez mais importante no Brasil, especialmente com o aumento drástico das ameaças cibernéticas e a crescente dependência tecnológica. Segundo Ribeiro (2016), a segurança da informação é primordial para proteger os dados contra acessos não autorizados, uso indevido, divulgação ou interrupção.

No Brasil, a segurança da informação se baseia em três principais pilares:

confidencialidade, integridade e disponibilidade. Esses pilares, nos asseguram de que as informações sensíveis não sejam acessadas por pessoas indesejadas, que os dados permanecem completos e que a informação sempre que necessária, esteja acessível.

5.2 Malware

A evolução dos *malwares* no contexto brasileiro representa uma ameaça significativa para a segurança digital. *Malware* são programas maliciosos desenvolvidos com intuito de causar danos, explorar ou obter dados/acesso não autorizado a sistemas e computadores, segundo Furtado et Al. (2014), no brasil, *malwares* como vírus, trojans, *spyware* e *ransomwares* são comuns, cada um possui suas características específicas.

5.3 Tipos de malware

5.3.1 ***Vírus:*** Programas que quando em contato no computador, se anexa a *softwares* ou arquivos. Quando executado, se espalham (Furtado et al. 2014);

5.3.2 ***Worms:*** *Malwares* que se replicam automaticamente se espalhando por redes. Não depende da interação do usuário (Rossi, 2018);

5.3.2 ***Trojans:*** Muito similar a programas legítimos, porém a execução traz ações maliciosas ao ser instalado (Oliveira & Souza, 2017);

5.3.3 ***Ransomware:*** *Malware* que realiza criptografia dos dados da vítima e exige resgate para descriptografia (Santos, 2019);

5.3.4 ***Spyware:*** *Software* que coletam informações sobre o usuário sem o devido conhecimento (Pereira, 2015);

5.4 Engenharia Reversa

No Brasil, a engenharia reversa tem se desenvolvido como uma prática importante para a análise e mitigação das ameaças cibernéticas. Segundo Souza e Silva (2016), a engenharia reversa é a prática para desmontar um *software* com intuito de entender seu funcionamento interno, especialmente a análise de *malwares*. Essa prática, está alinhada com técnicas como de assembladores, depuradores e analisadores de rede.

5.5 Desafios e Abordagens no Ensinamentos para Iniciantes

Promover conhecimentos a iniciantes na engenharia reversa apresenta grandes desafios devido à alta complexidade técnica e a quantidade de riscos. Segundo Martins (2016), é importante desenvolver uma base sólida de conhecimentos de segurança da informação e *malware* antes de aprender técnicas avançadas de engenharia reversa.

6. FERRAMENTAS E TÉCNICAS

A engenharia reversa de *malware* é um processo essencial para entender como programas maliciosos funcionam, permitindo a criação de estratégias para neutralizar suas ameaças. Ela envolve duas principais práticas: Análise Estática e Análise Dinâmica. A análise estática, examina o código do *malware* sem a necessidade de executá-lo, utilizando ferramentas como dessasembladores e depuradores. Por outro lado, a análise dinâmica, executa o *malware* em um ambiente controlado, para observar seu comportamento real.

Para realizar a engenharia reversa em executáveis, é necessário o uso de diversas ferramentas especializadas, exemplo:

6.1 Desassembladores: Ferramentas como *IDA Pro*, traduzem o código binário para um formato legível Souza & Silva (2016);

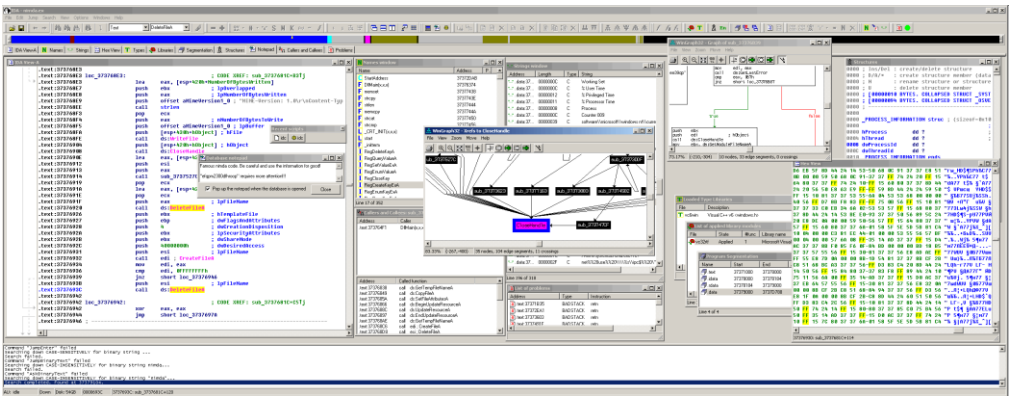

Figura 2 – Interface do IDA Pro. Fonte: E-SPIN Corp (2017)

6.2 Depuradores: Softwares como *OllyDbg* que permite a depuração passo a passo de um programa para observar cada movimento em tempo real (Freitas, 2017);

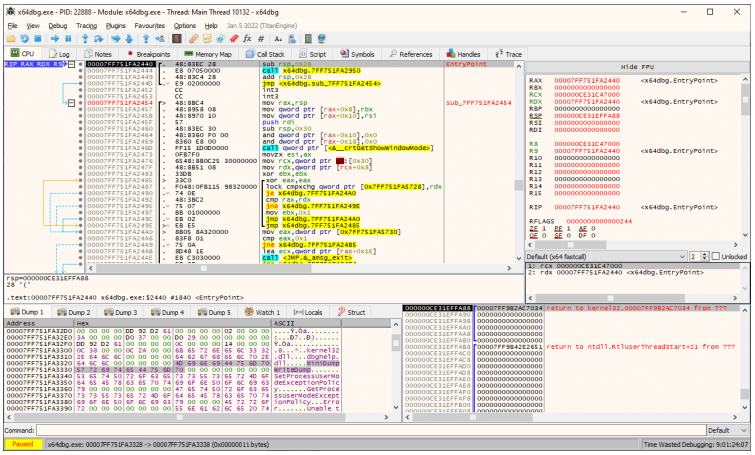

Figura 3 – Interface do x64dbg. Fonte: x64dbg (2023)

6.3 Análise Rede: Ferramentas como *Wireshark*, capturam e analisam tráfego de rede gerado pelo *malware* (Carvalho, 2018);

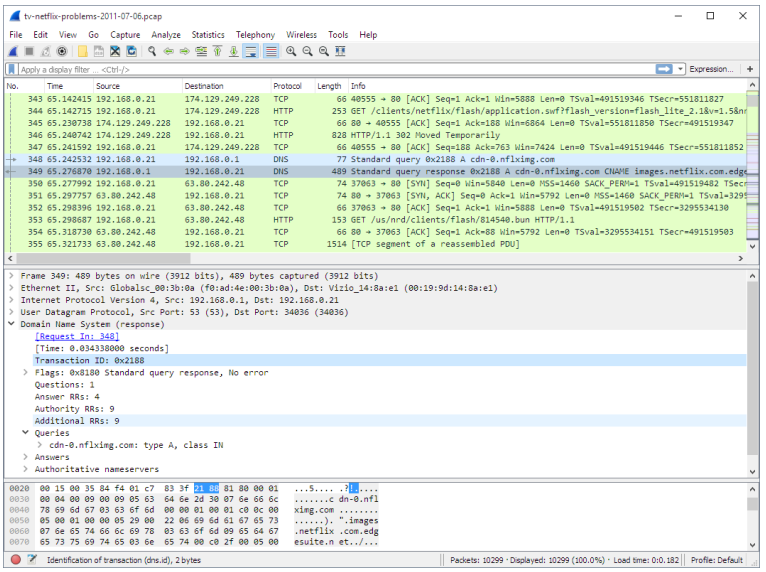

Figura 4 – Interface do Wireshark. Fonte: Wireshark (2023)

6.4 *Sandboxes*: Ambientes isolados onde o *malware* pode ser executado com total segurança para observar seus efeitos, não há riscos no sistema real (Nunes, 2019);

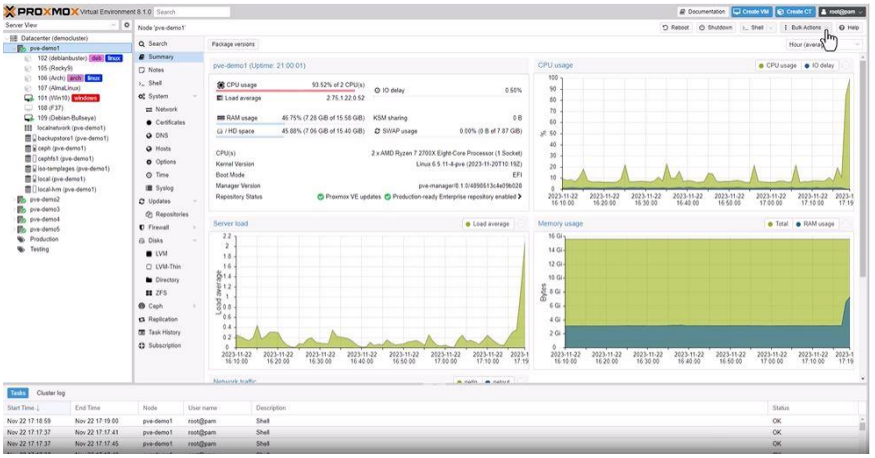

Figura 5 – Interface do Proxmox. Fonte: Proxmox (2023)

Essas ferramentas são fundamentais para desvendar o funcionamento de um programa malicioso e desenvolver técnicas para neutralizá-los.

Para solidificar a base do conhecimento teórico, é fundamental incluir atividades práticas para os iniciantes aplicar o conhecimento adquirido:

- ***Workshop* de Análise de *Malware*:** Sessões práticas onde os alunos podem usar as ferramentas como *Cheat Engine, IDA Pro Ghida* e *OllyDbg* para realizar suas análises de *malware* em *sandboxes*. Estes *workshops* incluem a análise de exemplos reais, permitindo que os iniciantes observem e compreendam o comportamento dos *malwares*.
- **Laboratório de engenharia reversa:** Aulas práticas onde os alunos debatem códigos maliciosos usando *sandboxes* para garantir ambientes seguros. Estes laboratórios envolvem análise de binários e engenharia reversa, com exercícios práticos e análise de código-fonte de *malwares*.
- **Projetos Simulados:** Projetos simulados onde os alunos trabalham em casos reais ou simulados para aplicar suas técnicas de engenharia reversa para identificar e neutralizar *malwares*. Esses projetos são supervisionados por especialistas, proporcionando uma experiência e aprendizado único.
- **Sessões de respostas a incidentes:** *Workshops* onde os alunos aprendem a responder a incidentes de segurança cibernética, simulando ataques e defesa. Estas sessões ajudam a desenvolver habilidades práticas e estrategistas de solução de programas em cenários de ataque e defesa.

Desafios e abordagens no Ensino para iniciantes

Ensinar engenharia reversa de *malware* para iniciantes apresenta desafios devido à complexidade técnica. Abordagens eficazes incluem, aprendizagem baseada em Projetos, permitindo que os alunos trabalhem em projetos reais ou simulados para solidificar os conceitos teóricos na prática (Silva, 2017), Monitoria e Suporte**,** para fornecer acesso a especialistas na área que possam guiar os iniciantes em suas análises (Santos, 2018) e oficinas práticas**,** entregando sessões *hands-on* onde os alunos possam usar ferramentas de engenharia reversa e analisar *malwares* em ambientes controlados.

Explorando engenharia reversa em Jogos Open Source com o Cheat Engine: Um Estudo sobre Manipulação de Variáveis de Jogos

A engenharia reversa é como o desmontar de um relógio para entender seu funcionamento por dentro. É essencial para criar tecnologias e proteger contra ameaças digitais. Contudo, o uso dessas técnicas de forma ética é crucial, especialmente quando são aplicadas em softwares sem autorização.

Este trabalho consiste em um ambiente controlado e educativo, onde serão exploradas técnicas de engenharia reversa para fins de aprendizado.

Uma ferramenta muito utilizada nesse processo é o *Cheat Engine*. Originalmente criado para modificar executáveis em tempo real, o *Cheat Engine* vai além, sendo excelente porta de entrada para os iniciantes em engenharia reversa. É uma ótima maneira de se iniciar antes de avançar para ferramentas mais complexas, como *IDA PRO* e *OllyDbg*, permitindo aos iniciantes entenderem conceitos fundamentais de uma maneira prática e acessível.

Esta introdução destaca a importância de começar com *Cheat Engine* antes de aprofundar em técnicas avançadas, enfatizando como essa ferramenta pode ser útil para os iniciantes em segurança da informação e engenharia reversa. Usando o *Cheat Engine*, esse trabalho visa construir uma base sólida antes de enfrentar desafios mais complexos com ética.

Quando usamos o *Cheat Engine* para buscas e alterar valores, como o caso de munições em jogos, estamos mexendo diretamente na memória do computador, os principais pontos são informações descrevendo como as munições são

guardadas na memória do RAM, em locais específicos que é acessado pelo jogo para saber quantas munições o jogador tem. O *Cheat Engine* localiza esses valores na memória e possibilita alterá-los. Por exemplo, o aumento de 10 para 1000 o número de munições. Isso acontece, pois, o *Cheat Engine* modifica diretamente os dados na memória dos jogos, fazendo com que seja reconhecido o novo valor. Esse processo altera temporariamente o valor na memória enquanto o jogo está aberto. Assim, o jogador poderá ver imediatamente os efeitos, como o aumento de munições disponíveis. Essas mudanças são validadas enquanto o *Cheat Engine* está aberto, após chegar o jogo, as alterações são perdidas

De forma resumida, o *Cheat Engine* aproveita a capacidade de acessar e modificar a memória de um determinado jogo em tempo real.

Já o *AssaultCube* é um jogo de tiro em primeira pessoa de código aberto, desenvolvido pela comunidade de desenvolvedores. Sua natureza *Open Source* permite que os desenvolvedores independentes contribuam para sua expansão e melhorias.

Neste trabalho, os testes práticos de engenharia reversa serão realizados utilizando o *AssaultCube* como plataforma de estudos, explorando como ferramentas como *Cheat Engine* que será aplicado para entender e modificar aspectos do jogo.

Ao abrir o jogo e iniciar uma partida personalizada, é possível notar no centro inferior à capacidade de munição.

Figura 6 – Interface do Jogo AssalutClubr - capacidade de munição

Primeiro passo é abrir o *Cheat Engine* e localizar o jogo aberto:

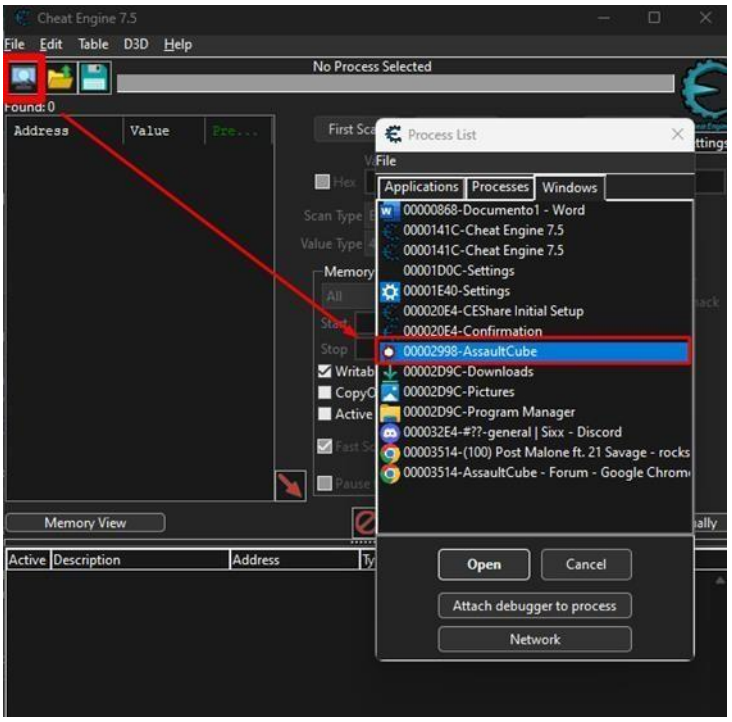

Figura 7 – Interface software cheat engine selecionando o alvo

Após selecionado, o jogo será monitorado pelo *Cheat Engine*. Em seguida, procedemos com a alteração dos valores buscados.

No caso das munições, são representadas por números inteiros de 2 *bytes*, devemos filtrar selecionando o tipo de valor como 2 *bytes.* e inserir o valor exato correspondente à capacidade atual de munição, a atual do jogo equivale a 58 no carregador. Em seguida, selecionamos para prosseguir.

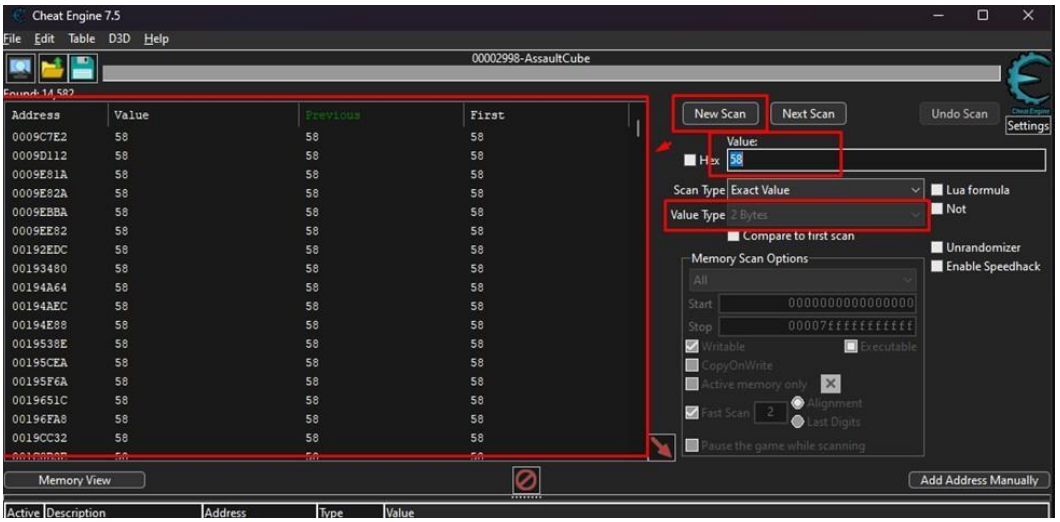

Figura 8 – Interface da interface do cheat engine realizando filtro

Após a realização da primeira busca, notamos que há uma quantidade significativa de valores com o mesmo valor encontrado no jogo, isso acontece, pois, o *Cheat Engine* realizou uma varredura completa no jogo e encontrou esses valores em aberto no jogo na memoria RAM do computador. O valor das munições está nesse meio, será realizada nova busca para filtragem mais correta e precisa. Essa busca consiste em dar um disparo com a arma para diminuir a munição e recarregar, assim, o total de 58 munições cairá para 57 e assim, sucessivamente.

Esse processo será realizado quantas vezes forem necessárias, a intenção é diminuir a quantidade o máximo possível para realizar a alteração no endereço de memoria correto.

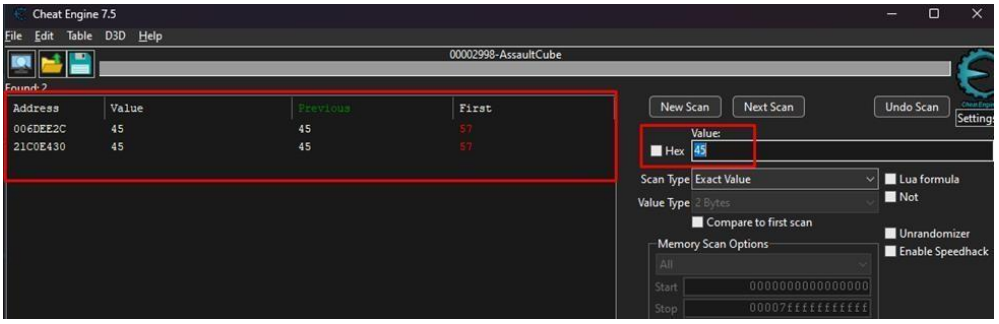

Figura 9 – Interface do cheat engine alterando o endereço de memória

Após uma sequência de filtragem, foram localizados os dois campos relacionados à capacidade de munição.

O próximo passo é selecioná-los e modificar os valores. Esse processo é o mais importante, pois na coluna esquerda podemos notar o endereço de memória que a capacidade de munição está armazenada.

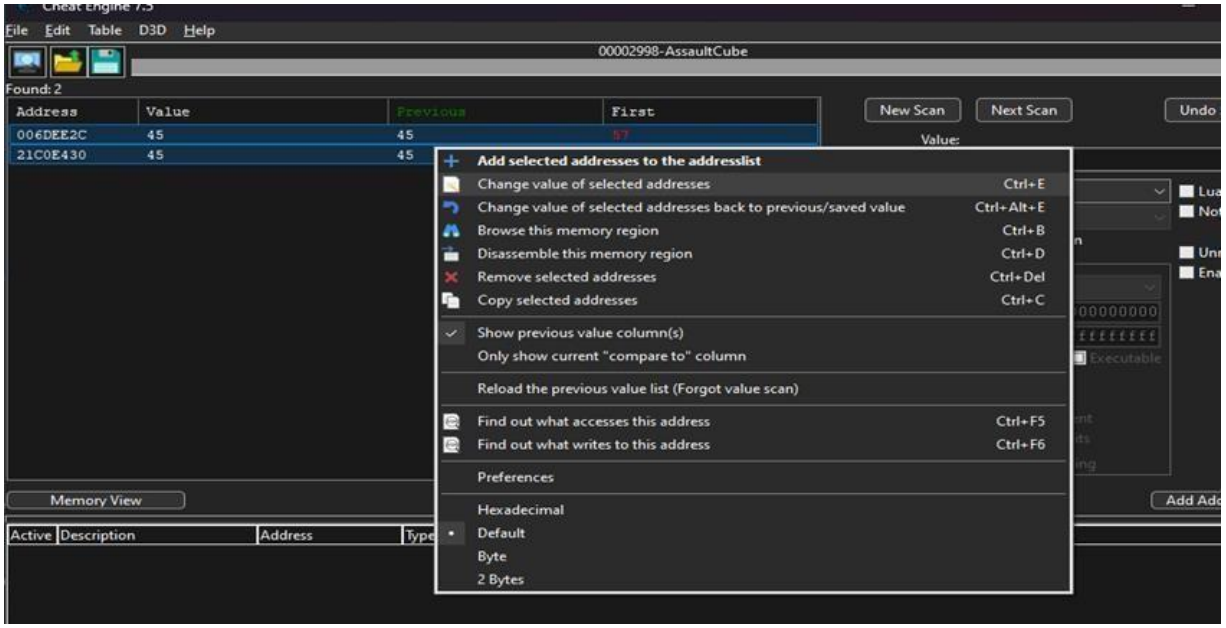

Figura 10 – Interface do cheat engine selecionando o endereço de memória para alteração

Para alterar os valores em campos de dados que são representados por números de bytes (exemplo: 1 byte, 2 bytes etc.), é fundamental entender os limites dessas representações, pois pode ocorrer um problema chamado overflow. Isso acontece devido a inserir um valor maior que o byte suporta. Por exemplo, um campo de 2 bytes pode armazenara valores entre 0 e 65525. Isso acontece, pois, cada byte pode conter 8 bits, portanto, 2 bytes podem representar 2^16. Tentar inserir um valor além dessa faixa ocorre problemas de overflow, devido à capacidade limitada de armazenamento desses campos específicos.

Figura 11 – Interface do jogo AssaultClube pós engenharia reversa

Notamos que os valores de capacidade máxima de munição foram alterados.

Ao explorar a engenharia reversa no jogo *AssaultCube*, podemos observar como essa ferramenta permite acessar e modificar diretamente a memória do jogo. Essas alterações, exemplifica como podemos alterar dinamicamente elementos do jogo em tempo real. Esse estudo não só ilustra a funcionalidade do *Cheat Engine*, mas também aponta a importância de aplicar essas técnicas de maneira ética e educativa. Através dessas abordagens, os iniciantes podem construir uma base sólida em engenharia reversa antes de explorar técnicas mais avançadas.

7. Conclusão

A engenharia reversa de *malware* é uma habilidade essencial na luta contra ameaças cibernéticas. Esse trabalho visa fornecer uma base sólida para iniciantes. Combinando teorias e práticas para desenvolver competências críticas. Ao prover ensinamento aos novos profissionais, estamos construindo um futuro mais seguro e inovador, onde a tecnologia prospera sem comprometer a segurança.

Referências

Carvalho, R. (2018). Análise de Redes: Fundamentos e Práticas. Editora Ciência Moderna.

Freitas, A. (2017). Depuração de Código: Técnicas e Ferramentas. Editora Campus.

Furtado, E., Oliveira, D., & Costa, L. (2014). Segurança da Informação e Malwares. Editora Érica.

Martins, C. (2016). formação em Segurança da Informação. Editora Atlas.

Nunes, F. (2019). Análise de Malware em Ambientes Controlados. Editora Ciência Moderna.

Oliveira, J., & Souza, M. (2017). Trojan Horses e suas Técnicas. Editora LTC.

Oliveira, P. (2019). Oficinas Práticas de engenharia reversa. Editora Saraiva.

Pereira, M. (2015). Spyware e Segurança da Informação. Editora FGV.

Ribeiro, A. (2016). Fundamentos de Segurança da Informação. Editora LTC.

Rossi, L. (2018). Worms: Comportamento e Mitigação. Editora Campus.

Santos, R. (2018). Mentoria em Segurança da Informação. Editora Atlas.

Santos, V. (2019). Ransomware: Prevenção e Mitigação. Editora Ciência Moderna.

Silva, D., & Lima, T. (2018). Ética na Análise de Malware. Editora Atlas.

Silva, L. (2017). Aprendizagem Baseada em Projetos na Segurança da Informação. Editora Saraiva.

Souza, A., & Silva, R. (2016). engenharia reversa de Software. Editora LTC.

Vlasic, B. (2010). Once Proudly Primitive, G.M. Sheds Its Swagger. The New York Time.

Langner, R. (2011). Stuxnet: Dissecting a Cyberwarfare Weapon. IEEE Security & Privacy, 9(3), 49-51.

Russinovich, M. E. (2012). Windows Internals: Parte 1. Microsoft Press.

Skoudis, E., & Zeltser, L. (2014). Malware: Fighting Malicious Code. Prentice Hall.

CHEAT ENGINE. Cheat Engine. Disponível em: https://cheatengine.org. Acesso em: 11 jul. 2024.

ASSAULTCUBE. AssaultCube - The Free FPS. Disponível em: https://assault.cubers.net. Acesso em: 11 jul. 2024.

Kaspersky. (2023). "Relatório de Ameaças Cibernéticas: Primeiro Trimestre de 2023."

E-SPIN Corp. (2017). IDA Pro Interface. Disponível em: https://www.e-spincorp.com/wp-content/uploads/2017/11/ida-idalarge-bigpicture-1024x398.gif

x64dbg.(2023).*x64dbg Interface*. Disponível em: https://x64dbg.com/img/slide2.png

Proxmox. (2023). *Proxmox VE Dashboard*. Disponível em: https://proxmox.com/images/proxmox/proxmox-ve/screenshots/proxmox-ve-dashboard.png

Wireshark. (2023). *Wireshark Main Interface*. Disponível em: https://www.wireshark.org/docs/wsug_html_chunked/images/ws-main.png