Ministério da Educação
Universidade Tecnológica Federal do Paraná
Câmpus Toledo
Engenharia de Computação

Relatório Técnico da prática de Disassembly – Organização e Arquitetura de Computadores

Acadêmicos: **RA:**
Gabriel Duarte Santos (2382954)
Gustavo d'Avila (2370506)

Data: 25 de Junho de 2024

Sumário

1. Explicação do algoritmo

O executável solicita ao usuário a entrada de uma senha e verifica se a senha fornecida está correta. Dependendo da senha inserida, o programa retorna uma mensagem indicando se a senha está correta ou incorreta.

2. Processo para encontrar a senha

Para encontrar a senha, foi utilizado o x64 Debugger para realizar a engenharia reversa do executável. Durante a análise, encontramos um valor em string equivalente a "**wlrulyhvw**". Após vários patches e testes, percebemos que a verificação da senha utilizava uma lógica de deslocamento de caracteres, onde cada caractere da string encontrada era três posições anteriores no alfabeto em relação ao caractere da senha correta (por exemplo, 'w' corresponde a 't' porque 't' é três posições antes de 'w' no alfabeto).

Utilizando essa lógica de deslocamento, descobrimos que a senha correta é "**tiorivest**".

3. Solução equivalente em C

Abaixo está a implementação equivalente do executável em C.

```c
#include <stdio.h>
#define SIZE 8

// Confere se a senha está correta
int checkPW(char* input, char* pw){
    int j;
```

```
    for(j = 0; j < SIZE-1; j++){
        // Conferir se o input é equivalente ao char - 3 da senha
        if((pw[j] - 3) != input[j]){
            return 0;
        }
    }
    return 1;
}

int main(){
    // Armazena a "senha" como wlrulyhvw
    char senha[SIZE+1] = "wlrulyhvw";
    char tentativa[SIZE] = {};

    printf("Organizacao e Arquitetura de Computadores\n");
    printf("Trabalho Pratico parte 2 - Disassembly\n");
    printf("Analizar e Escrever o algoritmo/fluxograma correspondente ao algoritmo
central; Descobrir a senha.\n");
    printf("Formato do Executavel: PE (Windows)\n");
    printf("Informe a Senha:\n");
    fgets(tentativa, 8, stdin);

    // Checa se o palpite está correto
    int temp = checkPW(tentativa, senha);

    // Verificação final da senha
    if(temp){
        printf("CERTA RESPOSTA!!!!! PARABENS!\n");
    } else {
        printf("Resposta Errada! Tente novamente...\n");
    }

    //scanf("%d"); // Facilitar a visualização no arquivo .exe
    return 0;
}
```

A implementação em C facilita a compreensão do algoritmo original. A transcrição do código assembly para C permitiu explorar a lógica de operações básicas, oferecendo uma visão clara de como um programa em assembly pode ser traduzido para uma linguagem de alto nível como C.

4. Conclusão

Este projeto nos proporcionou uma valiosa experiência prática e teórica. Através do disassembly, adquirimos conhecimentos que são diretamente aplicáveis na proteção e segurança do software. Entender como nossos programas podem ser analisados e potencialmente comprometidos nos permite desenvolver soluções mais robustas e seguras, garantindo a integridade e a confiança nos sistemas que criamos.

5. Referências

x64dbg - GitHub Repository: Este repositório no GitHub contém o projeto x64dbg, um depurador open source para plataformas Windows. Ele é amplamente utilizado para realizar disassembly e análise de executáveis, oferecendo uma interface amigável e

poderosas funcionalidades de depuração.

Introduction to Reverse Engineering with x64dbg - YouTube Video: Este vídeo oferece uma introdução detalhada à engenharia reversa usando o x64dbg. Ele abrange conceitos fundamentais e demonstra o uso da ferramenta para analisar e modificar executáveis.

Reverse Engineering with IDA Pro - YouTube Video: Este vídeo apresenta um tutorial sobre engenharia reversa utilizando IDA Pro. Através de exemplos práticos, ele ilustra como transformar o código desmontado em uma solução equivalente em C, permitindo uma compreensão mais profunda do funcionamento do executável.

Software Engineering | Reverse Engineering - GeeksforGeeks: Este artigo explica os conceitos fundamentais de engenharia reversa em software. Ele discute as motivações para realizar engenharia reversa, as técnicas utilizadas e os benefícios que podem ser obtidos através dessa prática, proporcionando uma visão abrangente e educacional sobre o assunto.