Bộ môn An toàn Thông tin – Khoa MMT&TT
Trường Đại học Công nghệ Thông tin (UIT)

BÁO CÁO THỰC HÀNH

Môn học: Lập trình hệ thống

Kỳ báo cáo: Buổi 05

Tên chủ đề: Tìm hiểu về kỹ thuật dịch ngược(ctn)

GVHD: Đỗ Thị Hương Lan

Ngày báo cáo: 07/06/2022

1. THÔNG TIN CHUNG:

Lớp: NT209.M21.ATCL

STT	Họ và tên	MSSV	Email
1	Trương Đình Trọng Thanh	20520766	20520766@gm.uit.edu.vn
2	Trần Đức Minh	20521617	20521617@gm.uit.edu.vn

2. NỘI DUNG THỰC HIỆN:[1]

STT	Công việc	Kết quả tự đánh giá
1	Bài 1	100%
2	Bài 2	100%
3	Bài 3	100%
4	Bài 4	100%
5	Bài 5	100%
6	Bài 6	0%

BÁO CÁO CHI TIẾT

[1] Ghi nội dung công việc, các kịch bản trong bài Thực hành

1. **Bài 1 (phase 1):**

- Định dạng của input và số lượng: nhập 1 chuỗi string

- Điều kiện ràng buộc của input: phải là chuỗi string phải trùng với kết quả được ẩn đi

- Kết luận về input: so sánh với kết quả cho sẵn, nếu đúng thì đi tiếp phase 2

```
signed int __cdecl strings_not_equal(int a1, int a2)
{
  int v2; // ebx@1
  signed int result; // eax@2
  int v4; // [sp+8h] [bp-Ch]@3
  int v5; // [sp+Ch] [bp-8h]@3

  v2 = string_length(a1);
  if ( v2 == string_length(a2) )
  {
    v4 = a1;
    v5 = a2;
    while ( *(_BYTE *)v4 )
    {
      if ( *(_BYTE *)v4 != *(_BYTE *)v5 )
        return 1;
      ++v4;
      ++v5;
    }
    result = 0;
  }
  else
  {
    result = 1;
  }
  return result;
}
```

(hàm so sánh input với kết quả được ẩn đi)

```
int __cdecl phase_1(int a1)
{
  int result; // eax@1

  result = strings_not_equal(a1, "I am not part of the problem. I am a Republican.");
  if ( result )
    explode_bomb();
  return result;
}
```

(hàm thực thi phase 1)

- Hình ảnh kết quả:

```
I am not part of the problem. I am a Republican.
Phase 1 defused. How about the next one?
```

2. **Bài 2 (phase 2):**

- Định dạng của input và số lượng:
- Điều kiện ràng buộc của input:
- Kết luận về input:
- Hình ảnh kết quả:

3. **Bài 3 (phase 3):**

- Định dạng của input và số lượng: nhập liên tiếp 3 chữ số dạng int
- Điều kiện ràng buộc của input: phải nhập đủ 3 chữ số dạng int
- Kết luận về input:

+ Đầu tiên, ta phải để ý v4, phải nhập theo case

```
v6 = __isoc99_sscanf(a1, "%d %c %d", &v4, &v2, &v5);
if ( v6 <= 2 )
  explode_bomb();
switch ( v4 )
{
  case 0:
    v3 = 'f';
    if ( v5 != 784 )
      explode_bomb();
    return result;
```

+ Sau đó, ta phải để ý v2 phải nhập đúng bằng với v3 khi ở dạng Dec

```
if ( v3 != v2 )
  explode_bomb();
```

+ Sau đó, ta phải để ý v5 phải nhập đúng bằng với case tương ứng

```
if ( v5 != 784 )
  explode_bomb();
```

- Hình ảnh kết quả:

```
0 f 784
Halfway there!
```

4. **Bài 4 (phase 4):**

- Định dạng của input và số lượng:
- Điều kiện ràng buộc của input:
- Kết luận về input:
- Hình ảnh kết quả:

5. **Bài 5 (phase 5):**

- Định dạng của input và số lượng: nhập liên tiếp 2 chữ số dạng int
- Điều kiện ràng buộc của input: phải nhập đủ 2 chữ số dạng int
- Kết luận về input:

```
int __cdecl phase_5(int a1)
{
  int v2; // [sp+14h] [bp-24h]@1
  int v3; // [sp+18h] [bp-20h]@1
  int v4; // [sp+1Ch] [bp-1Ch]@3
  int v5; // [sp+20h] [bp-18h]@3
  int v6; // [sp+24h] [bp-14h]@1
  int v7; // [sp+28h] [bp-10h]@3
  int v8; // [sp+2Ch] [bp-Ch]@1

  v8 = *MK_FP(__GS__, 20);
  v6 = __isoc99_sscanf(a1, "%d %d", &v2, &v3);
  if ( v6 <= 1 )
    explode_bomb();
  v2 &= 0xFu;
  v7 = v2;
  v4 = 0;
  v5 = 0;
  while ( v2 != 15 )
  {
    ++v4;
    v2 = array_2704[v2];
    v5 += v2;
  }
  if ( v4 != 15 || v5 != v3 )
    explode_bomb();
  return *MK_FP(__GS__, 20) ^ v8;
}
```

(Hàm thực thi phase 5)

```
data:0804D1BF                   db    8
data:0804D1C0 ; int array_2704[]
data:0804D1C0 array_2704        dd 0Ah                  ; DATA XREF: phase_5+63↑r
data:0804D1C4                   db    2
data:0804D1C5                   db    0
data:0804D1C6                   db    0
data:0804D1C7                   db    0
data:0804D1C8                   db 0Eh
data:0804D1C9                   db    0
data:0804D1CA                   db    0
data:0804D1CB                   db    0
data:0804D1CC                   db    7
data:0804D1CD                   db    0
data:0804D1CE                   db    0
data:0804D1CF                   db    0
data:0804D1D0                   db    8
data:0804D1D1                   db    0
data:0804D1D2                   db    0
data:0804D1D3                   db    0
data:0804D1D4                   db  0Ch
data:0804D1D5                   db    0
data:0804D1D6                   db    0
data:0804D1D7                   db    0
data:0804D1D8                   db  0Fh
data:0804D1D9                   db    0
data:0804D1DA                   db    0
data:0804D1DB                   db    0
data:0804D1DC                   db  0Bh
data:0804D1DD                   db    0
data:0804D1DE                   db    0
data:0804D1DF                   db    0
data:0804D1E0                   db    0
data:0804D1E1                   db    0
```

(Mảng array_2704[] trên IDA)

+ Đầu tiên, nhập v2 sao cho vòng lặp chạy được 15 lần (tương ứng v4 = 15)

+ Tiếp theo, viết lại mảng array_2704[] để cho dễ hình dung ta được:

array_2704[]={10, 2, 14, 7, 8, 12, 15, 11, 0, 4, 1, 13, 3, 9, 6, 5)

```
mảng  10, 2, 14, 7, 8, 12, 15, 11, 0, 4, 1, 13, 3, 9, 6, 5
a[i]  0   1  2   3  4  5   6   7   8  9  10 11  12 13 14 15
```

+ Xét v2, nếu nhập 5 thì v2 = 12 (v2[5]), sau đó cộng vào v5

```
v2 = array_2704[v2];
v5 += v2;
```

+ Sau đó đi tiếp vòng lặp ta được v2 = 3 (v2[12]), sau đó cộng dồn vào v5. Tiếp tục vòng lặp như trên cho tới khi v2 = 15 thì dừng.

+ Cuối cùng so sánh v5 với v3 lúc nhập kèm theo điều kiện v4 = 15 (tức phải chạy đủ 15 lần vòng lặp)

```
if ( v4 != 15 || v5 != v3 )
  explode_bomb();
```

- Hình ảnh kết quả:

```
5 115
Good work!  On to the next...
```

6. Bài 6 (phase 6): (bỏ)

Tổng quan kết quả:

```
Welcome to my fiendish little bomb. You have 6 phases with
which to blow yourself up. Have a nice day!
I am not part of the problem. I am a Republican.
Phase 1 defused. How about the next one?
1 2 4 8 16 32
That's number 2.  Keep going!
0 f 784
Halfway there!
99 3
So you got that one.  Try this one.
5 115
Good work!  On to the next...
```