# BÁO CÁO THỰC HÀNH LẬP TRÌNH HỆ THỐNG LAB3

# Nhóm 12:

22520825 Nguyễn Đức Luân

22520661 Vũ Ngọc Quốc Khánh

22521110 Đoàn Hoàng Phúc

Ta xem flowchart của chương trình bằng công cụ IDA pro 32 bit:



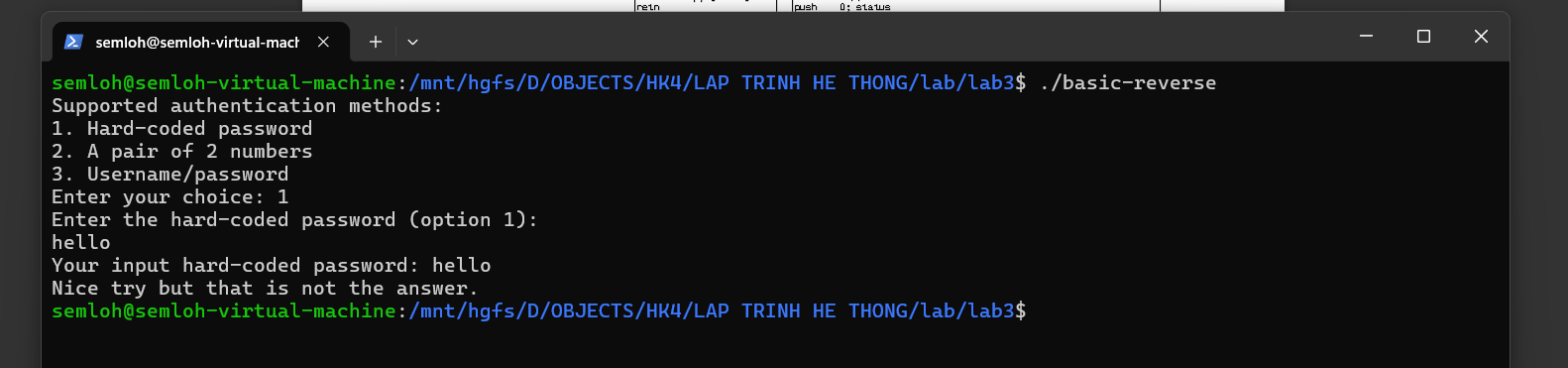
Chương trình sử dụng 3 hàm chính (3 options) tương ứng với 3 challenges cần solve:

• hardCode()

• otherhardCode()

• userpass()

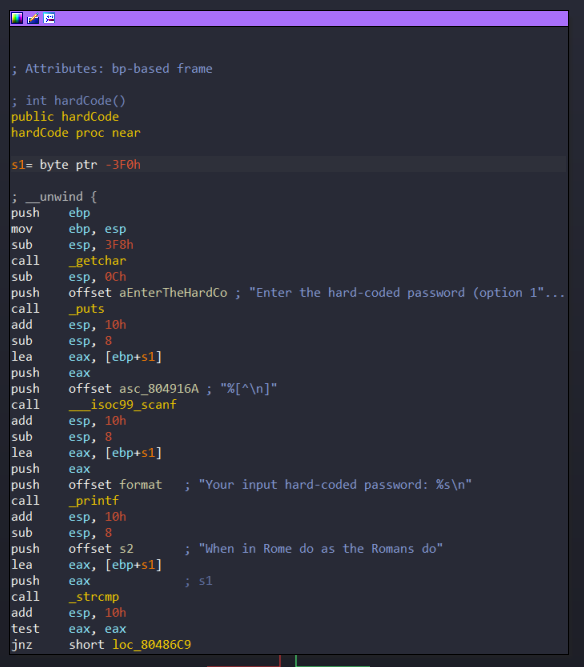
Ví dụ



A close-up of a sign

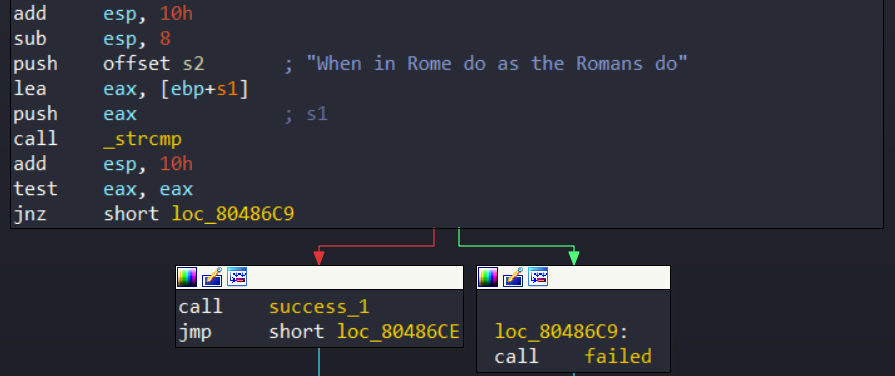
Description automatically generated

Xem xét hàm hardCode dưới mã assembly:



Hàm thực hiện tạo một biến s1 và được cấp phát. Ban đầu chương trình gọi hàm getchar() có lẽ để lấy ký tự xuống dòng “\n” trước đó. Sau đó dùng hàm thư viện C và chuẩn bị một số tham số cho hàm \_puts và in dòng chữ “Enter the hard-coded password (option 1):” được lưu trữ trong biến aEnterTheHardCo

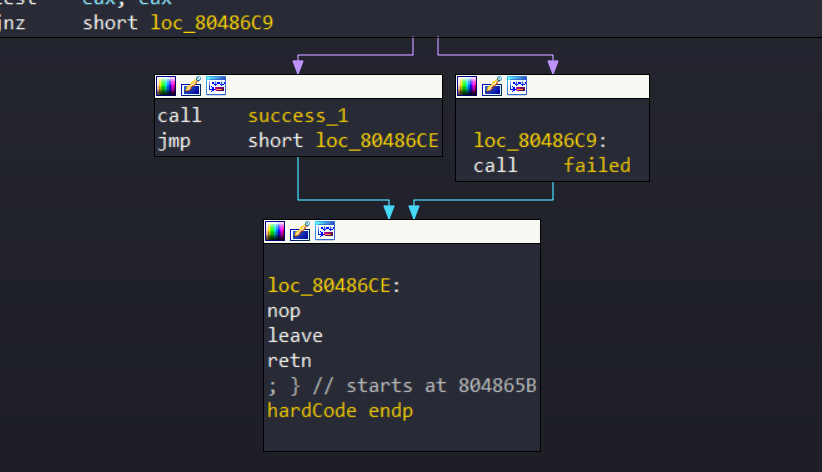
Tiếp tục, dùng \_\_\_isoc99\_scanf để cho người dùng nhập input. Dùng hàm \_printf để in ra thông báo “Your input hard-coded password: %s” được lưu trong biến format và thay thế chuỗi “%s” với password chúng ta nhập tại s1. Cuối cùng đến đoạn chương trình chính:



Để ý kỹ 2 dòng push các tham số vào s1 (password ta nhập), s2 (chuỗi cần so sánh). Chúng ta có thể thấy chuỗi được lưu tại biến s2

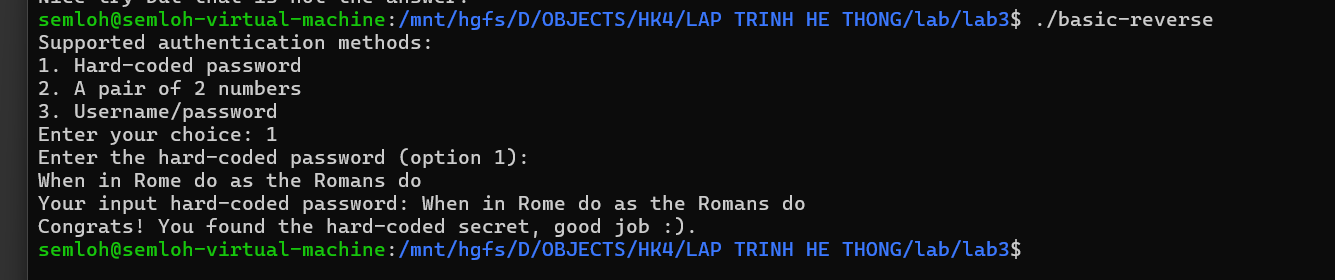
When in Rome do as the Romans do

Sau đó kết quả trả về ở eax, nếu eax = 0 thì 2 chuỗi bằng nhau thì thực hiện tiếp đến hàm success\_1 và in ra thông điệp báo thành công, còn nếu không bằng thì nhảy đến short loc\_80486C9 và thông báo không phê duyệt



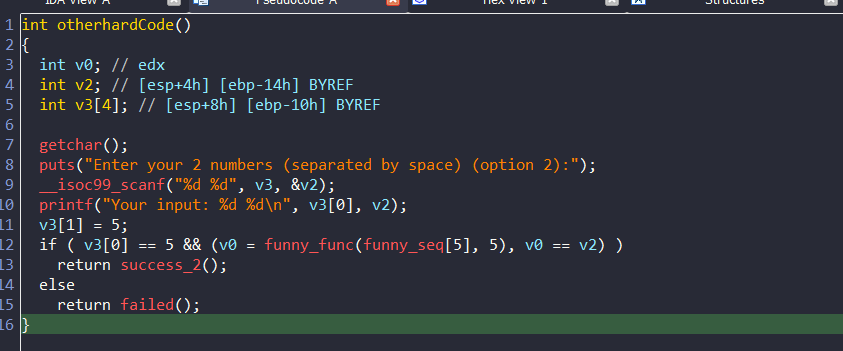
Vậy password là: When in Rome do as the Romans do

Kết quả thử nghiệm:



A close-up of a sign

Description automatically generatedXem xét hàm otherhardCode() trên mã giả:



Từ mã giả C cho thấy để có thể vào hàm success\_2() ta phải thỏa mãn điều kiện của lênh if trước đó:



Như ta đã biết thì biến v3[0] là số thứ nhất và v2 là số thứ hai nhập vào chương trình, điều kiện của lệnh if là số thứ nhất phải là 5 và số thứ 2 phải thỏa mã hàm điều kiện là phải bằng biến v0, v0 được tính bằng hàm sau:



Xem xét hàm funny\_func(): với hai tham số là funny\_seq[5], và số 5

A screenshot of a computer program

Description automatically generated

Từ đây ta thấy hàm này trả về kết quả của biểu thức trên. Vấn đề là xác định biến funny\_seq[5] có giá trị là bao nhiêu bằng cách dựa vào địa chỉ gốc của mảng funny\_seq[] với offset. Mà đây là mảng int nên địa chỉ của ô nhớ k = địa chỉ mảng + offset\*4

A screenshot of a computer

Description automatically generated

Từ hình trên ta xác định được địa chỉ của funny\_seq[5] là 08048A74 và có giá trị bằng 2. Lúc này ta đã có được 2 tham số của hàm funny\_func(): với hai tham số là funny\_seq[5] =2 và 5.Qua đó xác định được giá trị của biến v0 = 42

Vậy đáp án hai số là: 5 42

Thực nghiệm kết quả trên chương trình:

A screen shot of a computer

Description automatically generated

A close-up of a computer screen

Description automatically generated

Ta có 3 MSSV theo thứ tự tăng dần: 22520661 22502825 22521110

Từ đó xác định được username là: 661825110

Xem xét hàm userpass:

A screenshot of a computer

Description automatically generated

A screen shot of a computer program

Description automatically generated

Vào hàm userpass(): Đoạn đầu là khai báo các biến. Tiếp theo cấp vùng nhớ và giá trị “7’ZK\_” cho biến token và thực hiện tương tự công đoạn lấy dữ liệu đầu vào và hiển thị ra với 2 trường username và password: getchar() -> \_\_isoc99\_scanf -> getchar() -> puts() -> \_\_isoc99\_scanf -> printf. Xong rồi check thử xem độ dài password đầu vào của username : len(username)== 9? và xem len(password) == 9?

A computer screen shot of code

Description automatically generated

Sau đó chương trình tạo một biến v4 mới được hình thành như sau:

A computer screen shot of a program

Description automatically generated

Chạy riêng đoạn code này với 2 biến user = “661825110

” và token =”7’ZK\_” ta tìm được giá trị của v4 =”18107`ZK\_”

Tiếp đến là phần hình thành mật khẩu để đối chiếu:

A computer screen shot of a program code

Description automatically generated

Từ code trên ta thấy các kí tự của password bằng trung bình cộng các giá trị trong chuỗi của hai biến user và token:

Ta có chương trình giải mã:

A screen shot of a computer program

Description automatically generated

Password ta tìm được là: 37144JE>G

Vậy đáp án của bài này là

user: 661825110

password: 37144JE>G

Kết quả khi chạy chương trình:

A screen shot of a computer

Description automatically generated