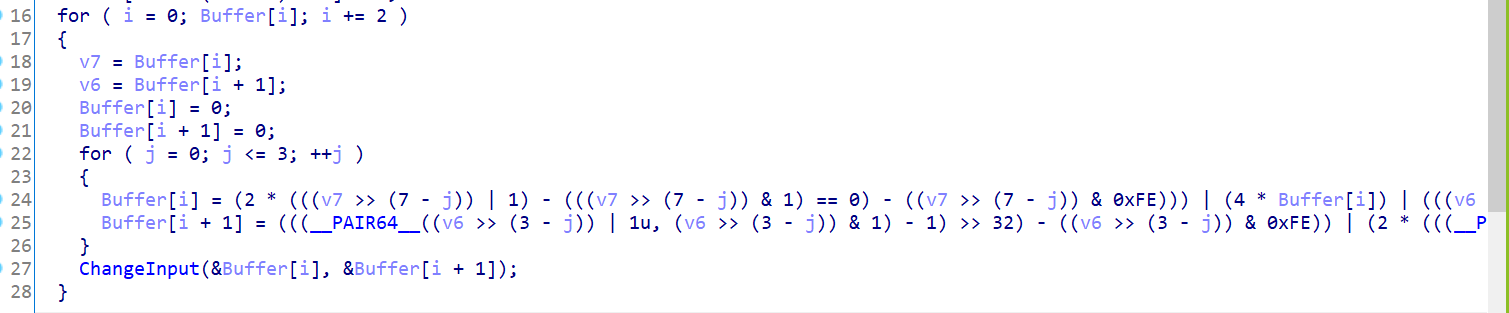


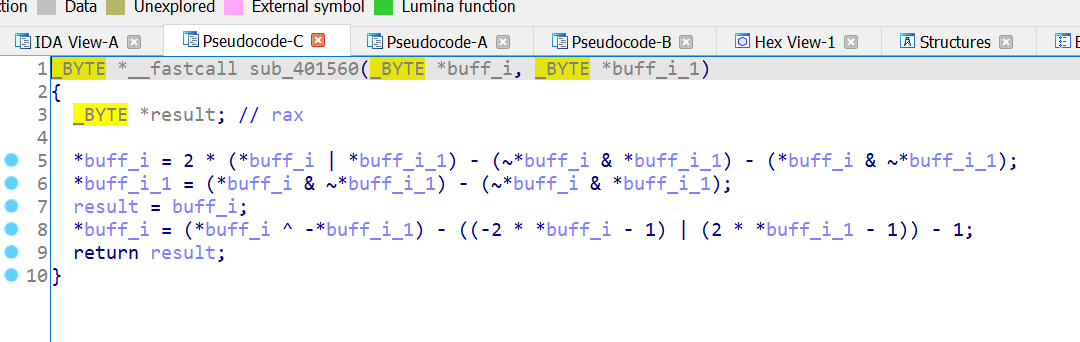
Buffer là nơi chứa dữ liệu input.

Sau khi nhận input từ bàn phím qua fgets, thì chèn byte NULL vào cuối buffer sau đó khởi tạo vòng for để xáo trộn buffer.

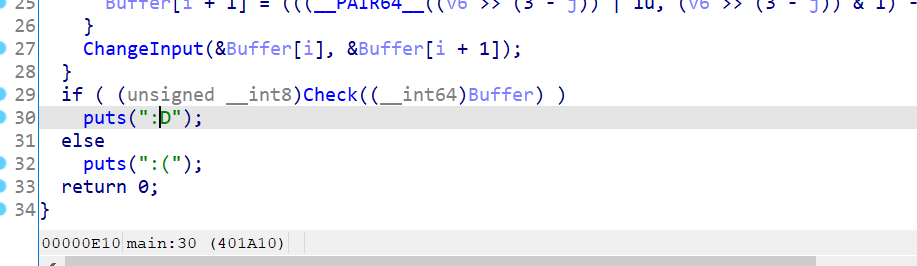


Vòng for sẽ dùng lần lượt 2 kí tự trong buffer, lưu tại v7 và v6

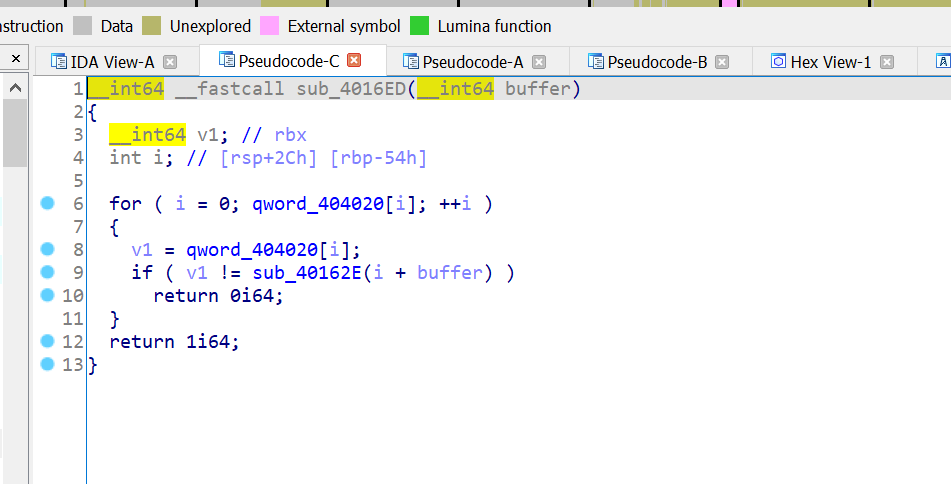
Rồi thực hiện các biểu thức and, shift, or kết hợp cùng Hàm ChangeInput với đầu vào là buff[I] và buff[I+1]



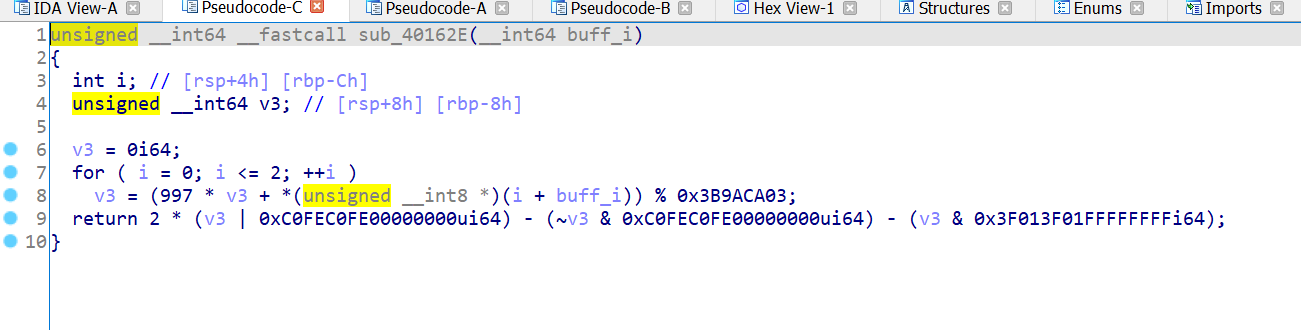
Hàm này sẽ thực hiện công việc chính đó là encrypt input



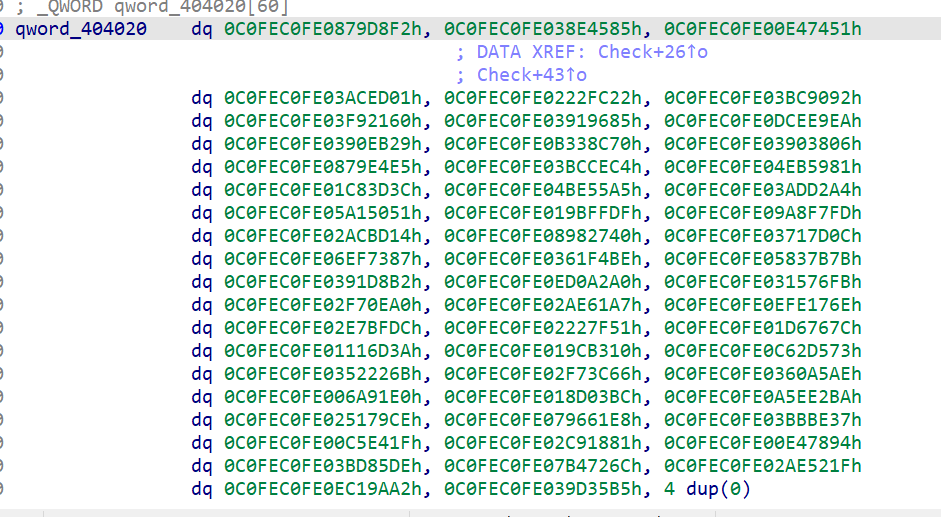
Sau đó gọi tới hàm Check



Dữ liệu sẽ tiếp tục đi qua 1 hàm biến đổi khác là sub\_40162E rồi được so sánh với từng phần tử trong mảng qword\_404020



Hàm Sub\_40162E dùng buff[I] và buff[I+1] để tạo result



Solution:

Em có đọc qua Z3 python nhưng chưa rõ cách sử dụng, bài này dùng z3 chắc dễ thở hơn nhiều

Hiện tại thì e làm như sau:

def revCMP(base):  
 for i in range(0,257):  
 for j in range(0,257):  
 arr = [i,j]  
 v3 = 0  
 for k in range(3):  
 v3 = (997 \* v3 + (arr[0]+k)) % 0x3B9ACA03  
 if ((2 \* (v3 | 0xC0FEC0FE00000000) - (~v3 & 0xC0FEC0FE00000000) - (v3 & 0x3F013F01FFFFFFFF)) == base):  
 break  
 return arr

Tạo 1 hàm vét cạn vì input là kí tự nên chỉ nằm trong 0 – 256

Truyền vào hàm này lần lượt các giá trị của mảng qword kia

Ta sẽ có 1 mảng là mảng sau lần mã hóa đầu tiên.  
  
 (x + y) == ((x ^ y) + 2(x & y))  
 (x + y) == ((x ^ y) - ((-2\*x - 1) | (-2\*y - 1)) -1)  
 (x + y) == (2(x | y) - (~x & y) - (x & ~y))  
 (x + y) == ((x | y) + (x & y))  
 (x + y) == (~(~(x + y) | ~(x + y)) | ~(~(x + y) |~(x + y)))  
 (x - y) == (x + ((y^-1) + 1))  
 (x - y) == ((x ^ (~y+1)) - ((-2\*x - 1) | (2\*y - 1)) - 1)  
 (x - y) == ((x & ~y) - (~x & y))  
 (x - y) == ~(~x + y)  
 (x ^ y) == ((x | y) - (x & y))  
 (x ^ y) == (x + y - 2\*(x & y)) (x | y) == ((x ^ y) ^ (x & y))  
 (x | y) == (x + y - (x & y))  
 (x & y) == (~(~x | ~y))  
 (x & y) == (x + y - (x | y))  
 (x & y) == ((x | y) - (~x & y) - (x & ~y))

Cái hint này em nghĩ là cách giải cho hàm mã hóa đầu tiên, e chỉ đến đây thôi ạ :<