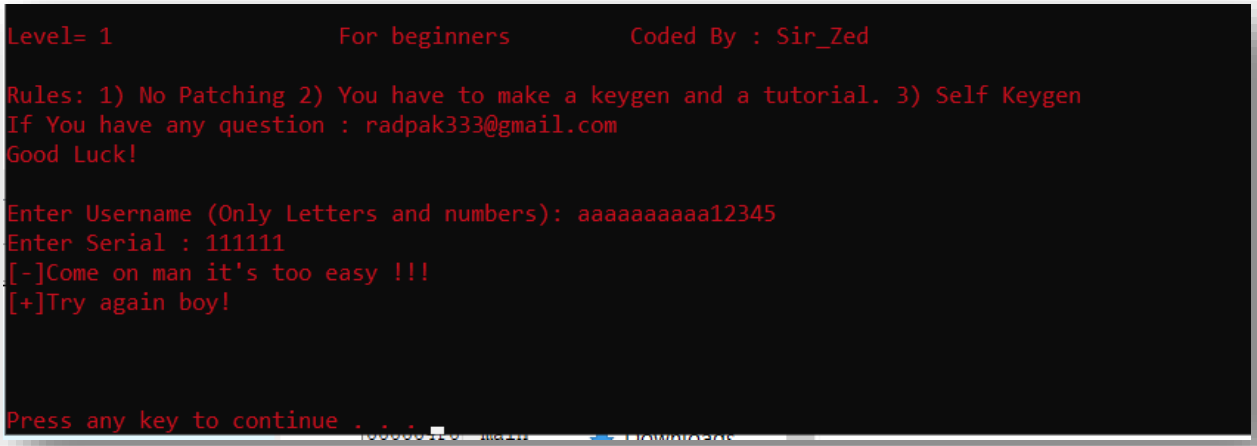
LẬP TRÌNH HỆ THỐNG– LỚP NT209.L21.ANTN

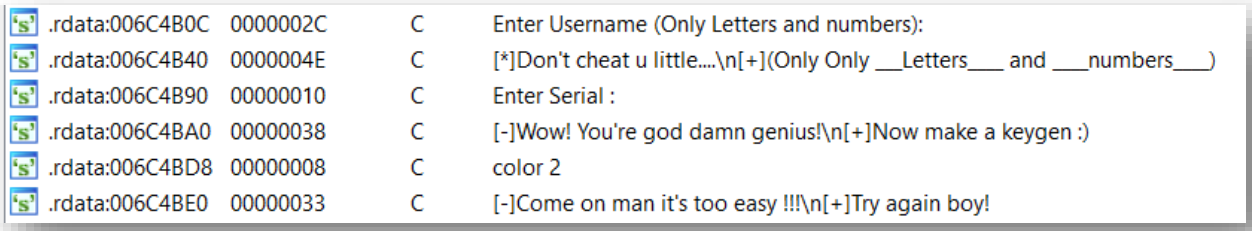
RE CHALLENGES 2: Keygen Me Part 1

Giảng viên hướng dẫn	Phạm Văn Hậu		ĐIỂM
Sinh viên thực hiện 1	Trần Đức Lương	19521815	

Chạy thử chương trình và nhập username và serial như hình dưới và nhận thông báo sai. Theo cách làm bài keygen challenge trước, có thể serial lần này cũng sẽ phụ thuộc vào username.



Phân tích file bằng IDA Pro, mở Strings thì thấy các dòng liên quan, cụ thể “Wow! You’re god damn genius! … ” là mục tiêu cần hướng đến.



Thực hiện truy xuất strings trên thì thấy chúng nằm trong hàm main. Quan sát hàm main thì thấy LABEL_64 là nơi chứa thông báo thành công. Tiếp tục truy ngược

lại thì thấy, để vượt qua challenge này cần có v41 == 0 và điều cần làm là cho chương trình đi qua LABEL_63.

```
      v41 = v37 ? -1 : 1;
      goto LABEL_64;
    }
LABEL_63:
    v41 = 0;
LABEL_64:
    if ( !v41 )
    {
      v42 = sub_402140(std::cout, "[-]Wow! You're god damn genius!\n[+]Now make a keygen :)");
      std::ostream::operator<<(v42, sub_402390);
      system("color 2");
```

Để làm được điều đó, tại LABEL_54 phải nhập username và serial sao cho v36 == 4 tức là Size[0] = 0. Tuy nhiên sẽ có thêm một cách nữa để nhảy tới LABEL_64 là với (Size[0] % 4 == 0) và sử dụng (*v34 == *v32) đủ số lần để giảm v36 -= 4.

```
    v36 = Size[0] - 4;
    if ( Size[0] < 4 )
    {
LABEL_54:
      if ( v36 == -4 )
        goto LABEL_63;
    }
    else
    {
      while ( *v34 == *v32 )
      {
        ++v34;
        ++v32;
        v37 = v36 < 4;
        v36 -= 4;
        if ( v37 )
          goto LABEL_54;
      }
    }
    v37 = *(_BYTE *)v34 < *(_BYTE *)v32;
    if ( *(_BYTE *)v34 != *(_BYTE *)v32
      || v36 != -3
      && ((v38 = *((_BYTE *)v34 + 1), v37 = v38 < *((_BYTE *)v32 + 1), v38 != *((_BYTE *)v32 + 1))
       || v36 != -2
       && ((v39 = *((_BYTE *)v34 + 2), v37 = v39 < *((_BYTE *)v32 + 2), v39 != *((_BYTE *)v32 + 2))
        || v36 != -1 && (v40 = *((_BYTE *)v34 + 3), v37 = v40 < *((_BYTE *)v32 + 3), v40 != *((_BYTE *)v32 + 3)))) )
    {
      v41 = v37 ? -1 : 1;
      goto LABEL_64;
    }
```

```
Size[0] = 0;
Size[1] = 15;
LOBYTE(Src[0]) = 0;
v29 = v68[0] - 1;
if ( v68[0] < v68[0] - 1 )
  v29 = v68[0];
v30 = Block;
if ( v68[1] >= 0x10 )
  v30 = (void **)Block[0];
sub_401E90(Src, v30, v29);
```

Hàm sub_401E90:

```
*((_DWORD *)this + 4) = Size;
```

Để ý đoạn code trên, ta thấy hàm sub_401E90 gọi với 3 tham số Src, v30, v29. Trong đó phần code hình trên của hàm sub_401E90, *(Src + 4) = v29, tức là (*Src + 4) chính là vị trí của Size[0]. Từ đó suy ra Size[0] = v29 = v68[0] – 1. Vậy mình phải nhập username sao cho (v68[0] - 1)% 4 == 0.

```
if ( Size[0] != v65 )
  goto LABEL_66;
```

Thực hiện phân tích code thì thấy một điều kiện cần nữa là Size[0] == v65 để không nhảy đế LABEL_66 . Sau khi đọc code và debug trong hàm nhập serial thì thấy v65 chính là kích thước của serial chúng ta nhập.

```
v20 = Size[0];
v21 = v68[0];
if ( Size[0] > v68[1] - v68[0] )
{
  LOBYTE(v53) = 0;
  sub_2226C0(Block, Size[0], v53, (int)v19, Size[0]);
}
else
{
  v68[0] += Size[0];
  v22 = Block;
  if ( v68[1] >= 0x10 )
    v22 = (void **)Block[0];
  v23 = (char *)v22 + v21;
  memmove((char *)v22 + v21, v19, Size[0]);
  v23[v20] = 0;
}
```

Nếu như độ dài serial < 16 (0x10) thì v65 = độ dài serial

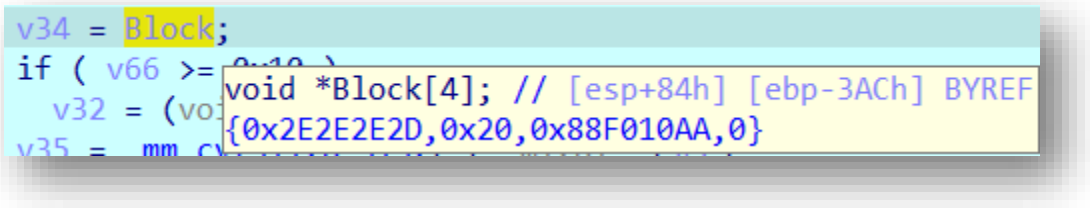
Vậy chúng ta sẽ lấy v68[0] = độ dài serial + 1 sao cho (v68[0] – 1) % 4 == 0 và số vòng while(*v34 == *v32) đủ để đưa v36 về = 0. Cụ thể sẽ lấy v68[0] = 5 => Độ dài serial cần nhập là 4.

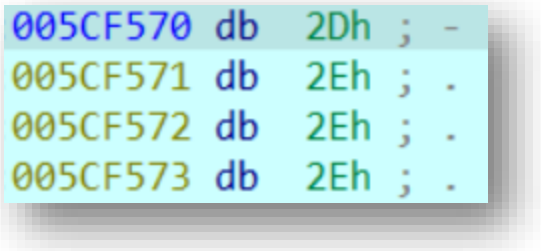
```
qmemcpy(v52, "ABCDEFGHIJKLMNOPQRSTUVWXYZ1234567890", sizeof(v52));
```

Thử debug với một vài kí tự a, b, c, d. Mình nhận ra rằng chuỗi với username bắt đầu bằng “b” sẽ tạo ra giá trị v68[0] = 5. Còn các kí tự còn lại khác với các kí tự thuộc v52 như trên nên ở đây em chọn 35 kí tự gồm “.” và “/”. Khi đó username có thể dùng là

“b/././/././/././/././/././/././/././/./”.

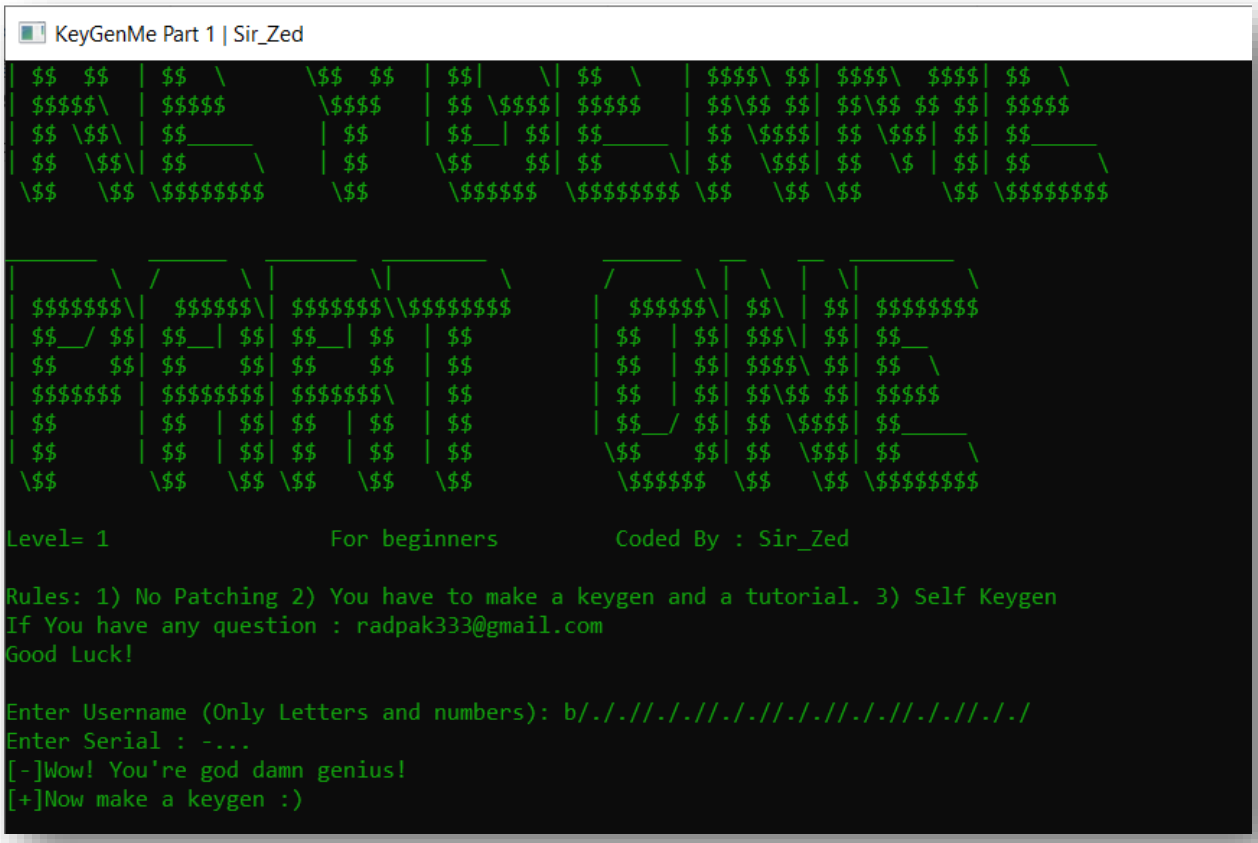
Tiếp theo mình cần nhập serial có độ dài là 4 và vòng while(*v34 == *v32) phải chạy ít nhất 1 lần. Khi đó kiểm tra *v34 thì thấy giá trị của nó bằng Block[0]:





Trong khi đó sau khi debug thì thấy *v32 chính là serial chúng ta nhập mà chương trình ở đây dùng cách lưu Little Endian nên serial chúng ta chính bằng "-..."

Thử chạy chương trình với cặp username serial trên, ta thấy thành công!!!!


KeyGenMe Part 1 | Sir_Zed
Level= 1
For beginners
Coded By : Sir_Zed
Rules: 1) No Patching 2) You have to make a keygen and a tutorial. 3) Self Keygen
If You have any question : radpak333@gmail.com
Good Luck!
Enter Username (Only Letters and numbers): b/././/././/././/././/././/././/././
Enter Serial : -...
[-]Wow! You're god damn genius!
[+]Now make a keygen :)