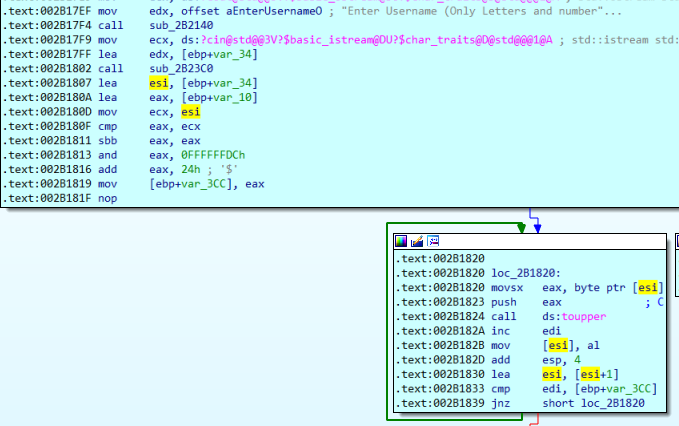
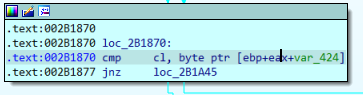
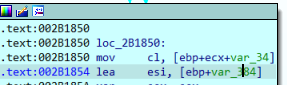
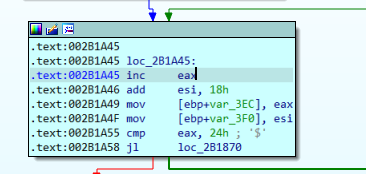
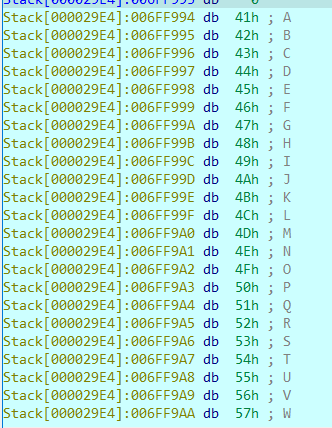
*Nguyễn Mỹ Quỳnh 19520241*

**Keygen Me Part 1**

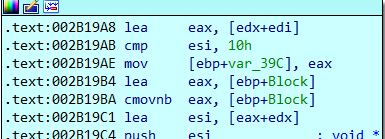
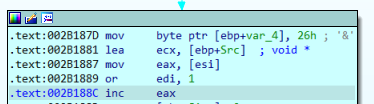
Bắt đầu đọc mã assembly cộng với việc debug, ta thấy ban đầu chương trình nhập Username vào vùng nhớ [ebp+var\_34], sau đó duyệt qua 36 lần tương ứng với từng kí tự trong chuỗi nhập vào và in hoa. Vậy chuỗi Username cần 36 kí tự.

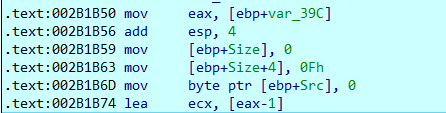


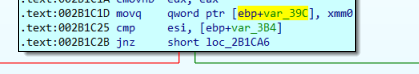
Tiếp theo chương trình thực hiện tính toán qua rất nhiều câu lệnh kết hợp debug ta có thể thấy chương trình thực hiện xét từng kí tự cl trong chuỗi nhập vào so sánh với kí tự trong mảng [ebp+eax+var\_424] và map tương ứng với sốtrong mảng [ebp+var\_384].

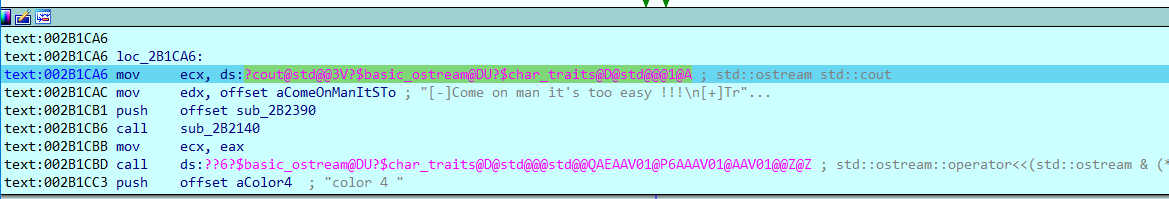


Kí tự được map như sau: A – 2; B – 4 ; C – 4; D – 3; E – 1; F – 4; G – 3; H – 4; I – 2; J – 4; K – 3; L – 4; M – 2; N – 2; O – 3; P – 4; Q – 4; R – 3; S – 3; T – 1; U – 3; V – 4; W – 3; X – 4; Y – 4; Z – 4; ‘0’ – 5; ‘1’ – 5; ‘2’ – 5; ‘3’ – 5; ‘4’ – 5; ‘5’ – 5; ‘6’ – 5; ‘7’ – 5; ‘8’ – 5; ‘9’ – 5

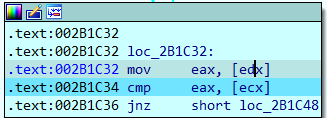
****Với mỗi lần map kí tự qua số sẽ được lưu ở [esi] sau đó tăng 1 cộng dồn và cuối cùng lưu vào vùng nhớ ebp+var\_39C

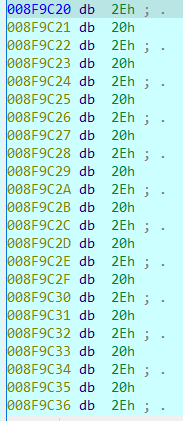
****Sau khi nhập erial, tiếp tục quan sát kết quả debug những dòng lệnh tiếp theo thì ta suy ra được (kết quả vùng nhớ ebp+var\_39C) -1 phải bằng với độ dài chuỗi serial, nếu không chương trình sẽ in ra "[-]Come on man it's too easy !!!\n[+]Tr"...

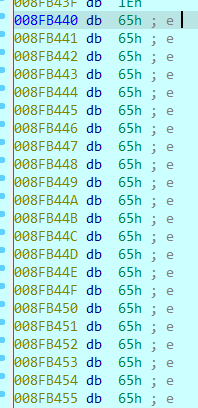


****

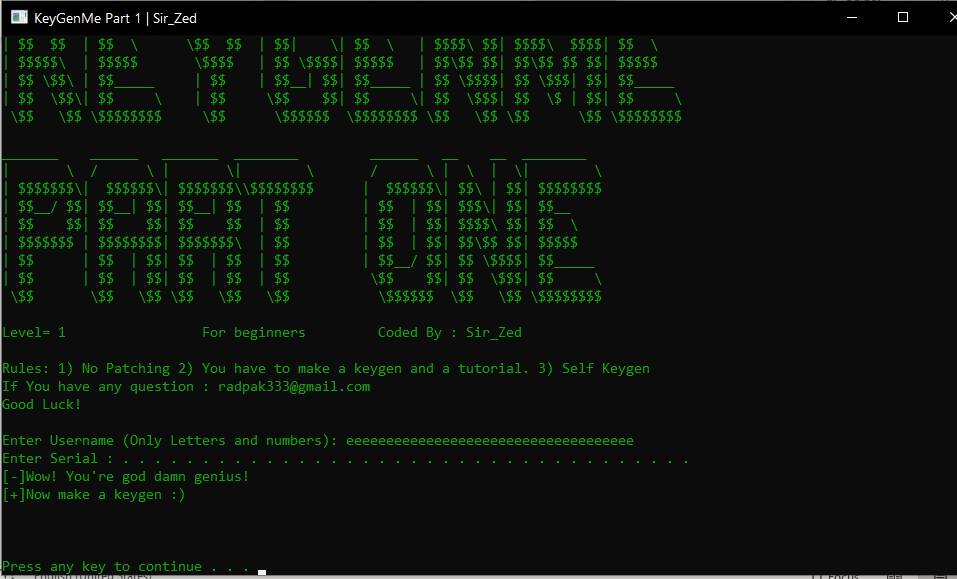
Đến đây ta thực hiện nhập lại chuỗi username và serial thỏa những điều kiện đã phân tích như trên. Để ý kí tự e map qua số ứng với 1(min) nên ta chọn user là 36 kí tự e để độ dài serial đạt cực tiểu là (1+1)\*36 -1 =71 kí tự. Tiến hành debug tiếp xem sau khi xét độ dài chuỗi serial đã thỏa mãn rồi thì chương trình làm gì. Ta thấy chương trình so sánh serial nhập vào với chuỗi 0x2E202E20….. Vậy ta suy ra 1 trường hợp thỏa bài toán là user là 36 kí tự e, serial là 71 kí tự xen kẽ “ ”(space) và “.” (little endian).

****

****

****

Kiểm tra:

****