

write up 2ourc3's Baby Keygen 3

step 1

le but de ce challenge est de construire un keygen qui vérifie les conditions donc après avoir ouvert le fichier dans IDA on c dirige dans la fonction valid_key

```
; Attributes: bp-based frame
; int __cdecl valid_key(char *key_to_valid)
public valid_key
valid_key proc near

length= dword ptr -0Ch
localkey= qword ptr -8
key_to_valid= qword ptr 10h

push    rbp
mov     rbp, rsp
sub    rsp, 30h
mov    [rbp+key_to_valid], rcx
mov    rax, [rbp+localkey]
mov    rdx, [rbp+key_to_valid] ; Source
mov    rcx, rax ; Destination
call   strcpy
mov    rax, [rbp+localkey]
mov    rcx, rax ; Str
call   strlen
mov    [rbp+length], eax
cmp    [rbp+length], 0Bh
jnz    short loc_4016C6
```

d'après la comparaison `cmp [rbp+length] , 0Bh` il faut que que la clé fasse 11 caractère

```
mov    rax, [rbp+localkey]
movzx eax, byte ptr [rax]
cmp    al, 41h ; 'A'
jnz    short loc_4016B3
```

```
mov    rax, [rbp+localkey]
add    rax, 3
movzx eax, byte ptr [rax]
cmp    al, 2Dh ; '-'
jnz    short loc_4016B3
```

```
mov    rax, [rbp+localkey]
add    rax, 7
movzx eax, byte ptr [rax]
cmp    al, 2Dh ; '-'
jnz    short loc_4016B3
```

ensuite on remarque que les 3 blocs d'instructions vérifie les certains caractère de la clé avec une comparaison , dans le premier bloc on compare le premier caractère a "A" , dans le second bloc l'instruction en assembleur `add rax, 3` nous indique que cette fois si le caractère vérifié est le 4 -ème , et dans le dernier bloc on vérifie que le caractère 8eme caractère sois un "-"

ainsi cela nous donne une clé sous la forme "Axx-xxx-xxx" vérifions cela
The key entered is valid.

```
Please enter a valid KEY: Aee-eee-eee
The key entered is valid
```

on peut voir que ça fonctionne

step 2

```
import random
import string
def keygen():
    lettres = string.ascii_letters
    key=""
    for i in range(11):
        if i ==0:
            key=key+"A"
        elif i ==3:
            key=key+"-"
        elif i ==7:
            key=key+"-"
        else :
            caractere_aleatoire = random.choice(lettres)
            key=key+caractere_aleatoire
    return key
```

ainsi si on appelle la fonction on obtient un résultat comme Alb-rfo-hha on test

```
The size of the KEY you provided is not valid.
Please enter a valid KEY: Alb-rfo-hha
The key entered is valid
```

et on peut voir que cela marche