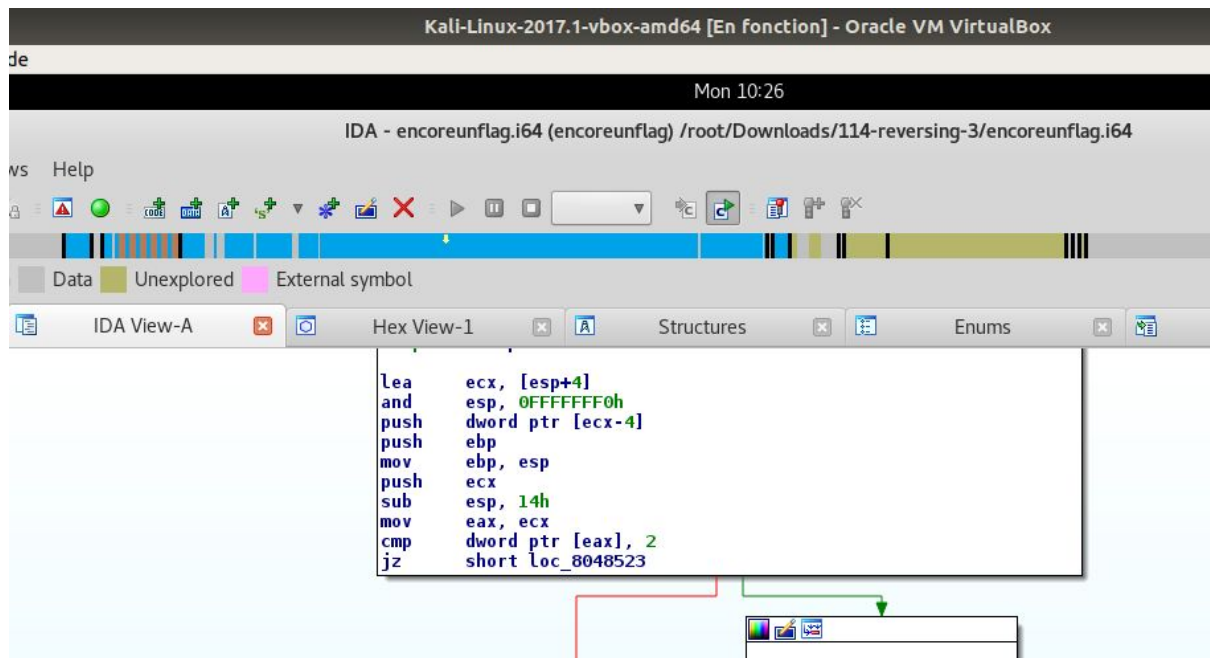
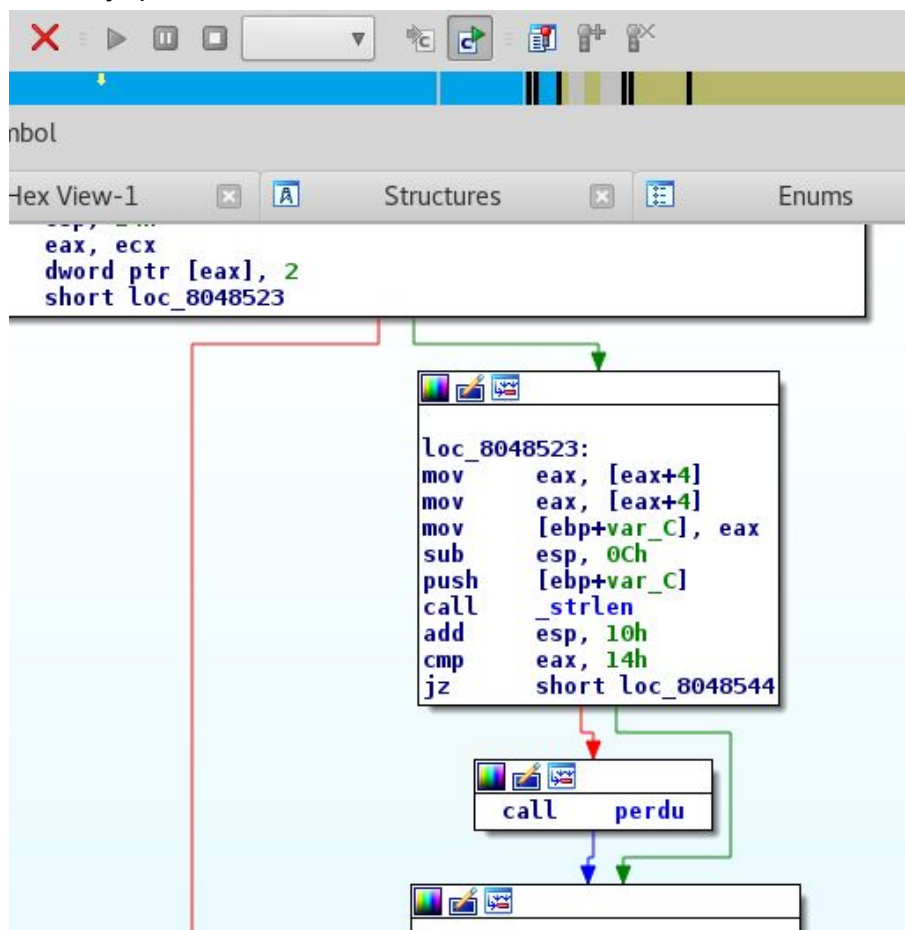


analyse du binaire avec IDA
analyse du control flow graph du main



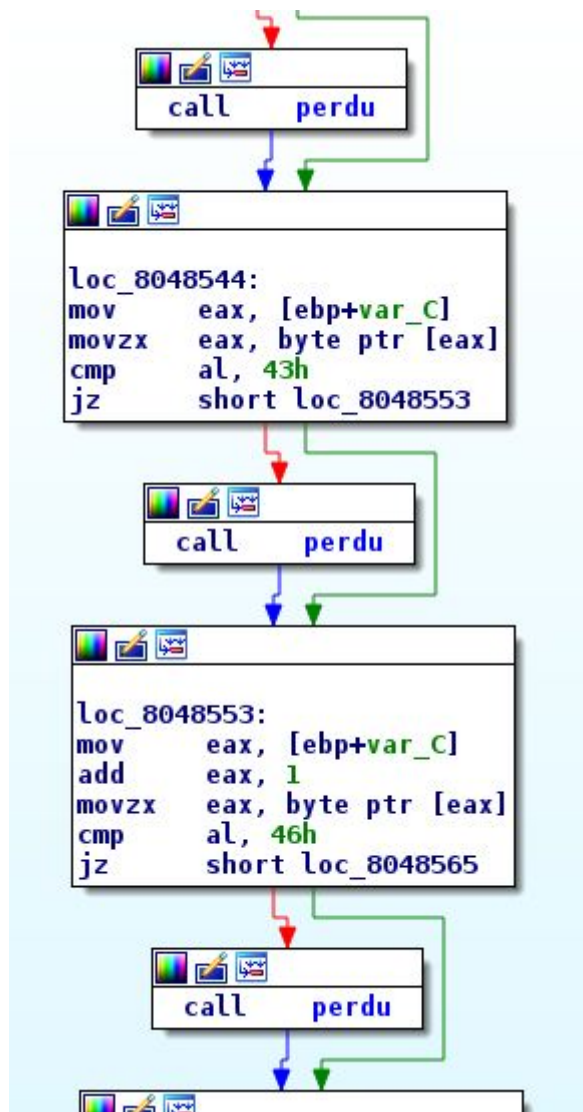
verifie nb d'arguments sinon affiche usage

ensuite jmp vers



verifie longueur de l'input (call _strlen) en comparant eax (contient le retour du call _strlen avec comme argument [ebp+var_c] qui correspond à notre input) à 14h (20 en decimal) -> le flag est composé de 20 caractères.

Si taille input != 20 -> call perdu



Chaque bloc compare le caractère de l'input a la position n avec le caractère du flag a la position n.

ligne : add eax, 1 (donne position du caractère comparer ici 1 (commence a la position 0))
cmp al, 46h (donne le caractere du flag a comparer , ici caractere du flag a la position 1 est 46 en hexa -> F

En remettant dans l'ordre on obtient le flag en hexa:

43 46 41 54 4c 59 6f 6a 74 72 48 75 4d 52 5a 49 4a 6b 71 78

CFATLYojtrHuMRZIJkqx

```
root@kali:~/Downloads/114-reversing-3# ./encoreunflag CFATLYojtrHuMRZIJkqx  
Et c'est gagné!!
```