

Esercizio Epicode 05/07/2023

0x00001141 <+8>: mov EAX,0x20 = sposta il valore convertito in decimale 32 nel registro EAX

0x00001148 <+15>: mov EDX,0x38 = sposta il valore convertito in decimale 56 nel registro EDX

0x00001155 <+28>: add EAX, EDX = somma il valore contenuto in EDX ad EAX e salva il risultato in EAX = 88 in decimale 0x58

0x00001157 <+30>: mov EBP, EAX = copia il contenuto del registro EAX nel registro EBP

0x0000115a <+33>: cmp EBP,0xa = compara il valore in EBP con il valore convertito in decimale 10, modifica i flag ZF (zero flag) e CF (carry flag). Essendo la destinazione EBP maggiore alla sorgente 0xa, ovvero 88,10 convertendo in decimale, sia ZF che CF saranno 0

0x0000115e <+37>: jge 0x1176 <main+61> = salta alla locazione 0x1176 se la destinazione EBP è maggiore o uguale della sorgente nell'istruzione cmp.

0x0000116a <+49>: mov EAX,0x0 = sposta il valore 0 nel registro EAX nel caso in cui il salto non venga eseguito

0x0000116f <+54>: call 0x1030 <printf@plt> = chiama la funzione printf in un'area specifica del processo. **printf@plt** è una piccola porzione di codice che chiama la vera funzione **printf**, modificando le cose lungo il percorso per rendere più veloci le chiamate successive.